5GCIDS: An Intrusion Detection System for 5G Core with AI and Explainability Mechanisms

Panagiotis Radoglou-Grammatikis[†‖], George Nakas[†], George Amponis[†], Sofia Giannakidou[†], Thomas Lagkas[‡], Vasileios Argyriou[§], Sotirios Goudos[¶] and Panagiotis Sarigiannidis[‖]

***Abstract*—The progression of fifth-generation (5G) networks provides multiple advantages, such as faster speed, reduced latency and increased capacity. Towards these advancements, it is clear that 5G Core (5GC) represents the heart of a 5G network, providing a variety of new network services such as Ultra-reliable low-latency communication (URLLC) and Massive machine-type communication (mMTC). However, despite the various benefits, 5GC is prone to several cyberthreats that can result in catastrophic effects. In this paper, an Intrusion Detection System (IDS) for 5GC is introduced. The proposed IDS called `5GCIDS` adopts Artificial Intelligence (AI) methods in order to detect potential cyberattacks against Packet Forwarding Control Protocol (PFCP), which is utilised for the N4 interface between Session Management Function (SMF) andUser Plane Function (UPF). For the detection process, both Transmission Control Protocol (TCP)/Internet Protocol (IP) flow statistics and application-layer PFCP flow statistics are used. In the second case, we provide a bidirectional flow statistics generator called `PFCPFlowMeter`. Finally, the detection outcomes are explained as local and global explanations with the TreeSHAP method. The evaluation results demonstrate the efficiency of the proposed IDS.**

***Index Terms*—5G, Artificial Intelligence, Intrusion Detection, PFCP**

I. Introduction

The evolution of fifth-generation (5G) networks represents a major milestone in the telecommunication domain. In particular, 5G has completely changed the game for wireless communication by providing multiple benefits, such as new levels of speed, capacity, and low latency [1]. The first steps towards 5G were taken with the deployment of 4G Long-Term Evolution (LTE) networks, which delivered improved data transfer rates and multimedia features. However, the increasing demand for greater bandwidth, the proliferation of the Internet of Things (IoT) entities and the need for ultra-reliable and low-latency communication led to the development of 5G networks. A key element of the 5G networks is the 5G Core (5GC), which serves as a centralised control and management mechanism for various Network Functions (NF). More specifically, 5GC follows a service-based architecture relying on several NF services, such as Session Management Function (SMF), Access and Mobility Management Function (AMF), and User Plane Function (UPF). Finally, it is worth mentioning that 5GC incorporates multiple technologies, such as network slicing, edge computing, Software-Defined Networking (SDN) and advanced virtualisation/containerisation mechanisms, thus allowing the development of a vast range of intelligent use cases, such as autonomous vehicles and telemedicine.

Although 5G introduces several benefits and opportunities, the complex nature of 5GC can result in various security issues [2]. First, malicious actors may attempt to exploit potential weaknesses and compromise the NFs in order to gain unauthorised access, disrupt network services and manipulate user data. On the other hand, 5GC handles a huge amount of data that can create privacy concerns. Finally, Denial of Service (DoS) attacks remain a critical threat which can raise negative effects such as service degradation or network outage. Depending on the use cases, the previous threats can lead to disastrous consequences such as financial losses or even fatal accidents in the context of Critical Infrastructures (CIs). Based on MITRE Adversarial Tactics, Techniques, and Common Knowledge (MITRE ATT&CK), FiGHT is a knowledge base of malicious tactics, techniques and sub-techniques for 5G environments. The current techniques in FiGHT are categorised into three classes: (a) theoretical, (b) Proof of Concept (PoC) and (c) observed. Therefore, based on the above-mentioned remarks, the presence of intrusion detection mechanisms for 5GC is necessary.

In this paper, an Intrusion Detection System (IDS) called `5GCIDS` is presented, focusing on the Packet Forwarding Control Protocol (PFCP), which is utilised for the communication between SMF and UPF. In particular, according to the detection techniques, an IDS can be categorised as signature/specification-based IDS or anomaly-based IDS. The first category uses pre-defined rules that can match malicious patterns, while anomaly-based IDS use statistical analysis and Artificial Intelligence (AI) methods in order

*This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 101097122 (ACROSS). Disclaimer: Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Commission. Neither the European Union nor the European Commission can be held responsible for them.

[†]P. Radoglou-Grammatikis, G. Nakas, G. Amponis and S. Giannakidou are with K3Y Ltd, William Gladstone 31, 1000 Sofia, Bulgaria - E-Mail: {pradoglou,gnakas gamponis}@uowm.gr

[‡] T. Lagkas is with the Department of Computer Science, International Hellenic University, Kavala Campus, Kavala, 65404, Greece - E-Mail: tlagkas@cs.ihu.gr

[§] V. Argyriou is with the Department of Networks and Digital Media, Kingston University London, Penrhyn Road, Kingston upon Thames, Surrey KT1 2EE, UK - E-Mail: vasileios.argyriou@kingston.ac.uk

[¶] S. Goudos is with School of Physics, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece - E-Mail: sgoudo@physics.auth.gr

[‖] P. Radoglou-Grammatikis and P. Sarigiannidis are with the Department of Electrical and Computer Engineering, University of Western Macedonia, Campus ZEP Kozani, Kozani, 50100, Greece - E-Mail: {pradoglou,psarigiannidis@uowm}.gr

to discriminate potential abnormalities. On the one hand, signature/specification-based IDS are characterised by high detection accuracy; however, they cannot recognise unknown anomalies and zero-day attacks. On the other hand, although anomaly-based IDS can recognise unknown attacks, false alarms may occur. Moreover, AI-based IDS suffer from explainability issues, given that the security administrators cannot comprehend and therefore trust the decision-making process of the AI models.

The rest of this paper is organised as follows. Section II discusses similar works in this field. Section III summarises the PFCP attacks examined in this study. In section IV, the architecture of `5GCIDS` is described. Next, section V focuses on the evaluation results. Finally, section VI concludes this paper.

II. Related Work

Many works have already investigated the 5G security issues. Based on them, subsequently, we briefly discuss a set of works that focus on 5GC-related attacks and relevant detection mechanisms.

In [3], G. Amponis et al. investigate and provide a set of PFCP DoS attacks against 5GC. In particular, five attacks are studied, namely (a) Unauthorised PFCP Session Deletion Request, (b) Unauthorised PFCP Session Modification Request, (c) Unauthorised PFCP Session Establishment, (d) Unauthorised UPF Forwarding rules Misconfiguration and (e) Eavesdropping User Traffic. The first attack targets the N4 interface of 5GC, while its impact is observed in the N3 interface. The attack is executed through a PFCP Session Deletion Control Message, which is sent by a compromised SMF to UPF. In the second attack, the goal of the attacker is to enforce UPF to drop packet-handling settings. For this purpose, PFCP Session Modification Requests with a DROP fag is sent by a compromised SMF to UPF. The third attack aims to exhaust the resources of UPF. To this end, a flood of Session Establishment Requests are sent by a compromised SMF to UPF. Next, in the context of Unauthorised UPF Forwarding Rules Misconfiguration, given the assumption that the attacker has already compromised UPF, then various settings of this NF can be violated, such as the `/proc/sys/net/ipv4` directory. Finally, the last attack is an extension of the Unauthorised PFCP Session Modification Request, where the attacker sends a Session Modification Request to redirect the UE traffic from the UPF to a compromised entity. Based on the first three attacker, the authors published in `IEEE Dataport` and `Zenodo`, the `5GC PFCP Intrusion Detection Dataset`.

In [4], Y. Fan et al. present `IoTDefender`, an intrusion detection framework for 5G-based IoT, taking full advantage of federated transfer learning. The architecture of the proposed IDS focuses mainly on the training procedure rather than the inference process (i.e., the use of the trained AI models in production environments). Hence, the architectural design follows a typical federated learning schema where first, the federated server distributes the pre-trained AI models to the federated clients. Next, the local training process is carried out by the federated clients. This process takes place in Mobile Edge Computing (MEC) nodes, taking full advantage of edge computing. Next, the parameters of the pre-trained AI models (after the local training procedure) are sent to the federated server for the fusion process. Then, the production of the final AI model follows by the federated server. It is noteworthy that the federated server is located in a cloud environment. Finally, the global AI model is sent to each federated client. Then, this client can be used in the production environment of each IoT network. Based on the experimental results with various publicly available datasets (CICIDS2017, NSL-KDD and three private IoT datasets), the detection accuracy of `IoTDefender` reaches 91.3%.

In [5], K. Sood et al. present an anomaly detection system which is compatible with the European Telecommunications Standards Institute (ETSI)-NFV standard for 5G environments. The proposed solution's architecture consists of two main modules: (a) Dimensionality Reduction (DR) and (b) Anomaly Detection (AD). The first module is responsible for reducing the dimensionality of the features used for the detection process, while AD has been trained to detect potential anomalies. For their experiments, the authors use the `UNSW-NB15 dataset`, `OMNET++` and `ETSI-NFV OSM`. From the architectural viewpoint, the aforementioned modules are handled as two complementary Virtual Network Functions (VNFs) that are integrated into the 5G Authentication Server Function (AUSF). Within 5GC, AUSF is responsible for handling the authentication and key management processes for the UEs. For the first module, a deep autoencoder is used, while for the second one, a Deep Neural Network (DNN) is adopted. Based on the experimental results, the accuracy of the proposed anomaly detection system reaches 98%.

Similarly to the previous work, in [6], Y-E. Kim et al. propose a feature selection method and an ML-based intrusion detection mechanism in order to recognise potential cyberattacks in 5GC, focusing on the General Packet Radio Service (GPRS) Tunneling Protocol (GTP). For their experiments, the authors use `UERANSIM` and `Open5GS` in order to emulate the Radio Access Network (RAN) and 5GC services, respectively. On the other hand, regarding the training of the ML-based detectors, the `Kitsune Network Attack Dataset` is utilised, including nine attack types: (a) scanning, (b) fuzzing, (c) video injection, (c) Address Resolution Protocol (ARP)-based Man-In-The-Middle (MITM), (d) active witercap, (e) Simple Service Discovery Protocol (SSDP) flood, (f) Synchronisation (SYN) DoS, (g) Secure Sockets Layer (SSL) renegotiation and (h) Mirai. However, Mirai and video injection are excluded from this work. Based on a specific packet sampling process of the `Kitsune Network Attack Dataset`, the Internet Protocol (IP) addresses of this dataset were used to create and establish the UEs within `UERANSIM`. Next, the device-level packets were replayed from each UE to 5GC. Then, the GTP-U packets are collected in the N3 interface between gNB and UPF. Finally, for the detection process, four AI methods are investigated: (a) Decision Tree, (b) Random

Forest, (c) K-Nearest Neighbour (KNN) and (d) Stacking Autoencoder. Based on the experimental results, the best accuracy is achieved by Random Forest.

Undoubtedly, all the previous works provide useful solutions and mechanisms. However, it is worth mentioning that none of them focuses on the N4 interface of 5GC. To the best of our knowledge, this is the first work which provides an AI-powered IDS with explainability mechanisms for the N4 interface, considering a set of cyberattacks related to the PFCP payload.

III. Threat Modelling

Our threat model focuses mainly on the N4 interface between SMF and UPF, where PFCP is used. Therefore, based on our previous work in [3], we investigate four PFCP-related cyberattacks, as briefly summarised below.

PFCP Session Establishment DoS Attack: This attack floods the UPF with valid Session Establishment and Heartbeat Requests, aiming to drain its resources. The attack has the potential to disrupt the 5GC network's ability to establish new Protocol Data Unit (PDU) sessions between clients and the Data Network (DN). This exploit specifically targets the N4 interface and impacts intermediate interfaces as well. To add complexity, each session establishment request generates a randomized Session ID (SEID).

PFCP Session Deletion DoS Attack: This attack aims to disconnect a UE from the DN. The attack script disrupts client-DN PDU sessions without disconnecting the UE from the 5G RAN or 5GC network. This exploit affects both the N4 and N6 interfaces. Connectivity can be restored by restarting the UE or by entering another gNb's coverage range. These actions will associate a new SEID with the UE's PDU session, effectively neutralising the attack.

PFCP Session Modification DoS Attack (DROP): This attack aims to invalidate session-specific packet handling rules in order to disconnect a targeted UE from the DN. During rules update, the UPF removes Forwarding Action Rule (FAR) entries associated with the Tunnel Endpoint Identifier (TEID) and base station IP address. As a result, the subscriber's GTP tunnel for downlink data transmission is severed, blocking DN access. However, sending data to the UPF can restore the GTP-U tunnel. It's important to note that this exploit specifically targets client-DN PDU sessions, without disconnecting the UE from the 5G RAN or 5GC. Its impact is limited to the DN, and the attack occurs via the N4 interface, affecting the N6 interface.

PFCP Session Modification DoS Attack (DUPL): This attack leverages the DUPL flag in the Apply Action field to compel the UPF to duplicate session rules, resulting in the creation of multiple paths for data originating from a single source. As a consequence, N6 interface instability may occur, along with the duplication of DN traffic. Moreover, this technique can be employed to initiate a DDoS attack targeting the entire DN. By forwarding packets to hosts outside the 5GC network, it exhausts the UPF's resources. A malicious actor can exploit this method by increasing the volume of transmitted packets per active user, progressively depleting the UPF's packet-handling resources.

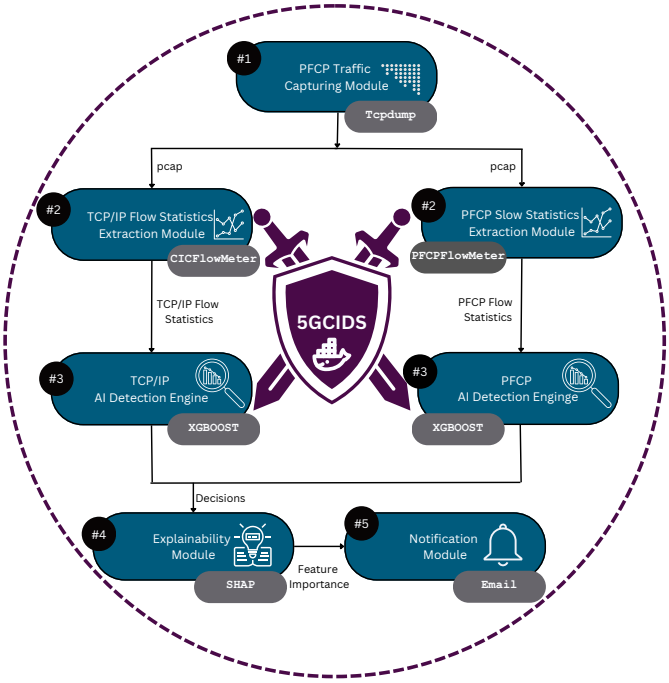

Fig. 1. Visual Representation of 5GCIDS Architecture

IV. 5GCIDS Architecture

The architecture of the proposed IDS consists of seven modules, namely: (a) PFCP Traffic Capturing Module, (b) TCP/IP Flow Statistics Extraction Module, (c) PFCP Flow Statistics Extraction Module, (d) TCP/IP AI Detection Engine, (e) PFCP AI Detection Engine, (f) Explainability Module and (g) Notification Module. The first module is responsible for capturing the PFCP network traffic data (i.e., PFCP pcap file). For this purpose, the overall IDS is deployed as a microservice within UPF, which is the target of the aforementioned PFCP-related cyberattacks. The second module receives the PFCP pcap files and generates bidirectional TCP/IP flow statistics/features. To this end, `CICFlowMeter` is used. On the other hand, the third module generates bidirectional flow statistics related to the application-layer attributes of PFCP. For this purpose, a new flow statistics generator was created, namely `PFCPFlowMeter`. The flow statistics/features generated by `PFCPFlowMeter` are given in Table I. Next, the TCP/IP AI Detection Engine and the PFCP AI Detection Engine receive the features from the previous module (i.e., `CICFlowMeter` and `PFCPFlowMeter`) and generate the detection results, respectively. Then, the Explainability Module receives (a) the AI models of the TCP/IP AI Detection Engine and the PFCP AI Detection Engine and (b) the flow statistics/features for each case and provides the local explanations (in terms of feature importance) for their detection results in the previous step, respectively. Finally, the Notification Module generates

the final security events, including the detection outcomes and the local explanations. More details about the detection process and the explainability of the AI models are provided in the following subsections.

TABLE I
APPLICATION LAYER PFCP FLOW STATISTICS – FEATURES

Feature	Description
flow ID	Flow identifier
source IP	Source IP address
destination IP	Destination IP address
source port	Source port number
destination port	Destination port number
protocol	Network layer protocol
duration	Length of time flow was active
fwd_packets	Number of forward packets
bwd_packets	Number of backward packets
PFCPHeartbeatRequest_counter	Number of PFCP Heartbeat Request messages
PFCPHeartbeatResponse_counter	Number of PFCP Heartbeat Response messages
PFCPPFDManagementRequest_counter	Number of PFCP PFD Management Request messages
PFCPPFDManagementResponse_counter	Number of PFCP PFD Management Response messages
PFCPAssociationSetupRequest_counter	Number of PFCP Association Setup Request messages
PFCPAssociationSetupResponse_counter	Number of PFCP Association Setup Response messages
PFCPAssociationUpdateRequest_counter	Number of PFCP Association Update Request messages
PFCPAssociationUpdateResponse_counter	Number of PFCP Association Update Response messages
PFCPAssociationReleaseRequest_counter	Number of PFCP Association Release Request messages
PFCPAssociationReleaseResponse_counter	Number of PFCP Association Release Response messages
PFCPVersionNotSupportedResponse_counter	Number of PFCP Version Not Supported Response messages
PFCPNodeReportRequest_counter	Number of PFCP Node Report Request messages
PFCPNodeReportResponse_counter	Number of PFCP Node Report Response messages
PFCPSessionSetDeletionRequest_counter	Number of PFCP Session Set Deletion Request messages
PFCPSessionSetDeletionResponse_counter	Number of PFCP Session Set Deletion Response messages
PFCPSessionEstablishmentRequest_counter	Number of PFCP Session Establishment Request messages
PFCPSessionEstablishmentResponse_counter	Number of PFCP Session Establishment Response messages
PFCPSessionModificationRequest_counter	Number of PFCP Session Modification Request messages
PFCPSessionModificationResponse_counter	Number of PFCP Session Modification Response messages
PFCPSessionDeletionRequest_counter	Number of PFCP Session Deletion Request messages
PFCPSessionDeletionResponse_counter	Number of PFCP Session Deletion Response messages
PFCPSessionReportRequest_counter	Number of PFCP Session Report Request messages
PFCPSessionReportResponse_counter	Number of PFCP Session Report Response messages
Downlink_counter	Number of downlink packets
Uplink_counter	Number of uplink packets
Bidirectional_traffic_counter	Number of bidirectional traffic packets
Label	Flow label (e.g. benign or malicious)

A. Detecting PFCP Cyberattacks

For the detection process two modules are used, namely (a) PFCP AI Detection Engine and (b) TCP/IP AI Detection Engine. For the PFCP AI Detection Engine, a decision tree is used for the detection process. A decision tree can be represented as a set of `if-else` statements, categorising the various instances into particular classes based on the various features. As indicated by their name, a decision tree consists of internal nodes and leaves. On the one hand, the internal nodes represent the *if-else* statements responsible for the decision about the classification problem in terms of splitting the overall data space into smaller data spaces given the available features. This decision can rely on various criteria, such as Entropy defined by $E(S) = -p_{(+)}logp_{(+)} - p_{(-)}logp_{(-)}$ where p_+ denotes the probability for the positive class, while p_i indicates the probability for the negative class. Finally, S implies a subset of the training dataset. The Entropy indicates the degree of uncertainty related to a node. The lower the Entropy, the higher the purity of this node. Although the Entropy can identify the uncertainty of a node, it cannot provide the Entropy of the parent node. In particular, the Entropy cannot identify whether the Entropy of the parent nodes has been decreased or not. For this purpose, Information Gain: $IG = E(Y) - E(Y|X)$ is used. IG can measure the reduction of uncertainty based on the various features and play an important role as a deciding factor regarding which nodes will act as internal ones or leaves. Based on the aforementioned remarks, many algorithms can generate decision trees based on a labelled dataset, such as the Classification and Regression Tree (CART), Iterative Dichotomiser 3 (ID3), J48, Chi-square Automatic Interaction Detector (CHAID), C4.5 and Quick, Unbiased, Efficient, Statistical Tree (QUEST). In this paper, CART i used. In particular, CART separates the dataset based on a single feature x and a relevant threshold t. To this end, CART searches for the best pair between x and t, which will provide the purest subsets. Next, for each subset, the same method is used according to a hyperparameter called `max depth` which defines when the splitting process will stop, thus avoiding overfitting issues.

$$\text{Log Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij}\log(p_{ij}) \tag{1}$$

On the other hand, eXtreme Gradient Boosting (XGBoost) is used by the TCP/IP AI Detection Engine. XGBoost is an ensemble ML method that is able to produce an ensemble based on weak learners. In this case, short decision trees are used. In particular, the ensemble is sequentially constructed by training each weak learner with the data samples that are previously classified incorrectly. For this purpose, the multi-class log loss (Equation 1) is used as a loss function. In each iteration, a new decision tree is added with the goal of minimizing the error. The final model prediction is the aggregation of the individual models' outputs.
where N is the total number of instances (samples) in the dataset, M is the total number of classes. y_{ij} is a binary indicator (0 or 1) if class j is the true class for instance i. Finally, p_{ij} is the predicted probability of instance i belonging to class j

B. Explainability of the Detection Outcomes

The Explainability Module is responsible for providing consistent and reliable explanations for the predictions of the previous detection engines. Based on the tree-based methods used for the detection process, the TreeSHAP method is chosen in order to explain their decision in terms of feature attribution. TreeSHAP is a model-specific explanation method designed for tree-based models, such as decision trees, random forests, and gradient boosting machines. It is based on Shapley values, which are a way of allocating credit to features in a model's prediction. As illustrated in Algorithm 1, TreeSHAP works by recursively traversing the decision tree, attributing contributions to each feature as it moves down the tree.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$TPR = \frac{TP}{TP + FN} \tag{3}$$

$$FPR = \frac{FP}{FP + TN} \tag{4}$$

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{5}$$

Algorithm 1: TreeSHAP Algorithm

Input: Decision Tree Model (e.g., XGBoost, LightGBM);
Instance to explain;
Dataset to compute average Shapley values
Output: Shapley values for each feature

Algorithm:

Initialize Shapley values for each feature to zero:
shapley_values = [0] * num_features;

Traverse the decision tree to the leaf node corresponding to the instance:
while *not at leaf node* **do**
  Follow the appropriate branch based on the feature value
end

Update the Shapley values at each node:
Calculate the contribution of the feature at the current node:
contribution = (phi_left - phi_right) / num_instances
where phi_left = prediction from the left branch
phi_right = prediction from the right branch
num_instances = number of instances that reached the current node
Update the Shapley values for each feature:
shapley_values[feature_index] += contribution;

Recursive backtracking:
while *not at root node* **do**
  Move to the parent node
  Repeat the update of Shapley values
end

Average the Shapley values across all instances:
shapley_values /= num_instances_in_dataset;

Return the computed Shapley values for each feature.

V. Evaluation Analysis

Before analysing the evaluation results, we have to define first the necessary evaluation metrics. In particular, for the detection process, we use four metrics: (a) Accuracy (ACC) (Equation 2), True Positive Rate (TPR) (Equation 3), False Positive Rate (FPR) (Equation 4) and the F1 score (Equation 5), where TP: True Positive, TN: True Negatives, FP: False Positives and FN: False Negatives. On the other side, for the explanations generated by TreeSHAP, appropriate diagrams show the importance of the various features. Regarding the detection process, a set of ML/DL methods are compared with each other, including: Logistic Regression, Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis, K-Nearest Neighbour (KNN), Decision Tree, Naive Bayes, Support Vector Machine (SVM), AdaBoost, Gradient Boosting, Random Forest, Extra Trees, XGBosst, LightGBM and Multi-Layer Perceptron (MLP). For this comparative study, the `5GC PFCP Intrusion Detection Dataset` [7] is used. As depicted in Table II, using the application-layer PFCP flow statistics/features generated by `PFCPFlowMeter`, the best detection efficiency is achieved by the decision tree. Fig. 2 and Fig. 4 shows the confusion matrix and the global explanations of this model. On the other hand, as illustrated in Table III, using the TCP/IP flow statistics/features provided by `CICFlowMeter`, the best performance is accomplished by XGBoost. Similarly, Fig. 3 and Fig. 5 depict the confusion matrix and the global explanations of this model.

TABLE II
Comparative Study of ML/DL Models trained with Application Layer PFCP Flow Statistics/Features

ML/DL Method	ACC	TPR	FPR	F1
Logistic Regression	0.536	0.533	0.116	0.486
LDA	0.536	0.533	0.116	0.492
QDA	0.54	0.538	0.115	0.463
KNN	0.612	0.61	0.097	0.614
Decision Tree	**0.641**	**0.639**	**0.09**	**0.642**
Naive Bayes	0.54	0.538	0.115	0.492
SVM	0.532	0.529	0.117	0.482
AdaBoost	0.62	0.618	0.095	0.627
Gradient Boosting	0.62	0.618	0.095	0.629
Random Forest	0.629	0.627	0.093	0.633
Extra Trees	0.633	0.631	0.092	0.641
XGBoost	0.603	0.601	0.099	0.612
LightGBM	0.565	0.563	0.108	0.579
MLP	0.536	0.533	0.116	0.486

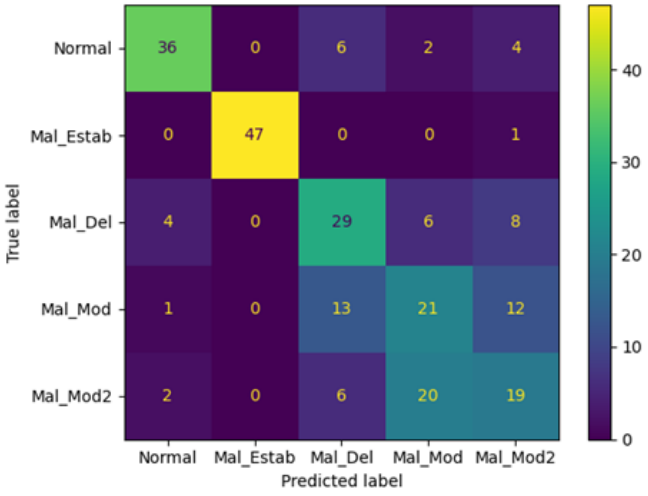

Fig. 2. Confusion Matrix Explanations for the Decision Tree Model trained with Application Layer PFCP Flow Statistics/Features

VI. Conclusions

In this paper, we introduce an AI-powered IDS for 5GC with an explainability service, focusing on the N4 interface of 5GC between SMF and UPF. In particular, the proposed IDS called `5GCIDS` emphasises on four cyberattacks against PFCP, which is used for the N4 interface. `5GCIDS` take into account both TCP/IP flow statistics (generated by `CICFlowMeter`) and application-layer PFCP flow statistics (generated by `PFCPFlowMeter`). `PFCPFlowMeter` was

TABLE III
COMPARATIVE STUDY OF ML/DL MODELS TRAINED WITH TCP/IP FLOW STATISTICS/FEATURES

ML/DL Method	ACC	TPR	FPR	F1
Logistic Regression	0.559	0.559	0.11	0.535
LDA	0.566	0.566	0.108	0.565
QDA	0.547	0.546	0.113	0.508
KNN	0.769	0.769	0.058	0.769
Decision Tree	0.817	0.817	0.046	0.817
Naive Bayes	0.487	0.487	0.128	0.425
SVM	0.489	0.489	0.128	0.432
AdaBoost	0.681	0.68	0.08	0.667
Gradient Boosting	0.844	0.844	0.039	0.844
Random Forest	0.83	0.83	0.043	0.83
Extra Trees	0.832	0.832	0.042	0.831
XGBoost	**0.855**	**0.855**	**0.036**	**0.854**
LightGBM	0.841	0.84	0.04	0.84
MLP	0.559	0.559	0.11	0.536

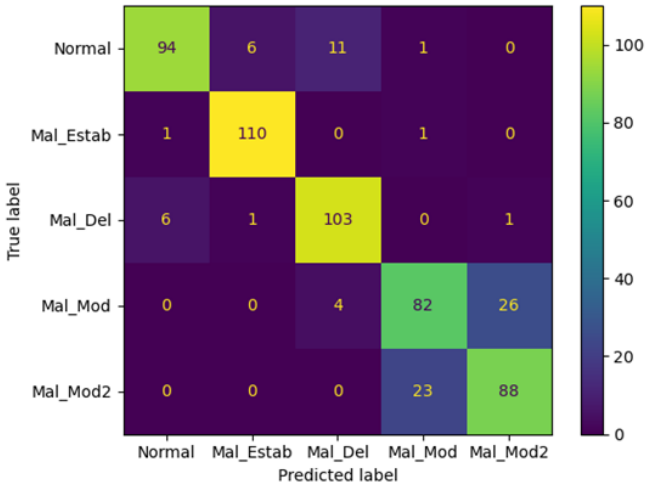

Fig. 3. Confusion Matrix for the XGBoost Model trained with TCP/IP Flow Statistics/Features

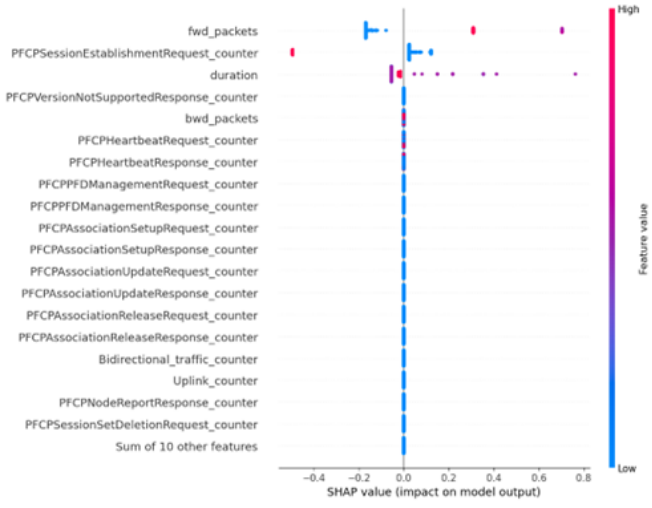

Fig. 4. Global Shap Explanations for the Decision Tree Model trained with Application Layer PFCP Flow Statistics/Features

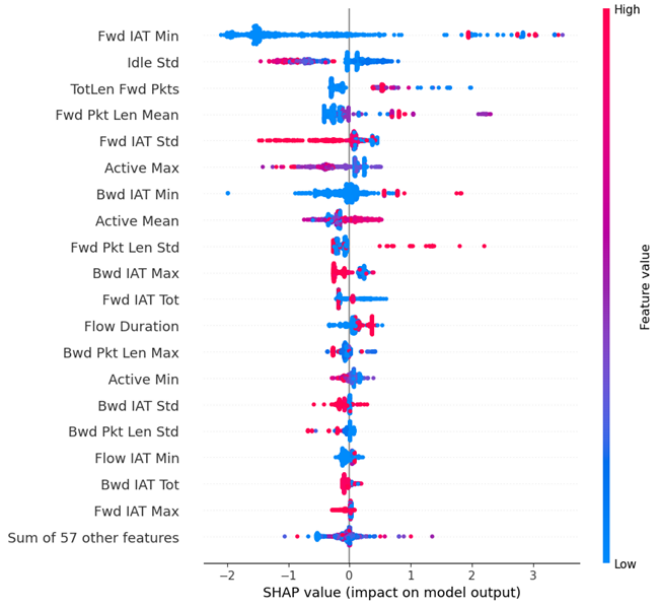

Fig. 5. Global Explanations for the XGBoost Model trained with TCP/IP Flow Statistics/Features

implemented in the context of this work. For the detection process, XGBoost is used with the TCP/IP flow statistics, while the decision tree is used with the application-layer PFCP flow statistics. Finally, the explainability module relies on the TreeSHAP methods. The efficiency of the proposed IDS is demonstrated by the evaluation results.

REFERENCES

[1] S. Giannakidou, P. Radoglou-Grammatikis, S. Koussouris, M. Pertselakis, N. Kanakaris, A. Lekidis, K. Kaltakis, M. P. Koidou, C. Metallidou, K. E. Psannis *et al.*, "5G-Enabled NetApp for Predictive Maintenance in Critical Infrastructures," in *2022 5th World Symposium on Communication Engineering (WSCE)*. Nagoya, Japan: IEEE, 2022, pp. 129–132.

[2] G. Amponis, P. Radoglou-Grammatikis, T. Lagkas, S. Ouzounidis, M. Zevgara, I. Moscholios, S. Goudos, and P. Sarigiannidis, "Towards Securing Next-Generation Networks: Attacking 5G Core/RAN Testbed," in *2022 Panhellenic Conference on Electronics & Telecommunications (PACET)*. Tripolis, Greece: IEEE, 2022, pp. 1–4.

[3] G. Amponis, P. Radoglou-Grammatikis, T. Lagkas, W. Mallouli, A. Cavalli, D. Klonidis, E. Markakis, and P. Sarigiannidis, "Threatening the 5G core via PFCP DoS attacks: the case of blocking UAV communications," *EURASIP Journal on Wireless Communications and Networking*, vol. 2022, no. 1, pp. 1–27, 2022.

[4] Y. Fan, Y. Li, M. Zhan, H. Cui, and Y. Zhang, "Iotdefender: A federated transfer learning intrusion detection framework for 5g iot," in *2020 IEEE 14th international conference on big data science and engineering (BigDataSE)*. Guangzhou, China: IEEE, 2020, pp. 88–95.

[5] K. Sood, M. R. Nosouhi, D. D. N. Nguyen, F. Jiang, M. Chowdhury, and R. Doss, "Intrusion detection scheme with dimensionality reduction in next generation networks," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 965–979, 2023.

[6] Y.-E. Kim, Y.-S. Kim, and H. Kim, "Effective feature selection methods to detect IoT DDoS attack in 5G core network," *Sensors*, vol. 22, no. 10, p. 3819, 2022.

[7] G. Amponis, P. Radoglou-Grammatikis, G. Nakas, S. Goudos, V. Argyriou, T. Lagkas, and P. Sarigiannidis, "5G Core PFCP Intrusion Detection Dataset," in *2023 12th International Conference on Modern Circuits and Systems Technologies (MOCAST)*. Athens, Greece: IEEE, 2023, pp. 1–4.