Detecting Drones Status via Encrypted Traffic Analysis

Savio Sciancalepore, Omar Adel Ibrahim, Gabriele Oligeri, Roberto Di Pietro
Division of Information and Computing Technology
College of Science and Engineering, Hamad Bin Khalifa University
Doha, Qatar

ABSTRACT

We propose a methodology to detect the current status of a powered-on drone (flying or at rest), leveraging just the communication traffic exchanged between the drone and its Remote Controller (RC). Our solution, other than being the first of its kind, does not require either any special hardware or to transmit any signal; it is built applying standard classification algorithms to the eavesdropped traffic, analyzing features such as packets inter-arrival time and size. Moreover, it is fully passive and it resorts to cheap and general purpose hardware. To evaluate the effectiveness of our solution, we collected real communication measurements from a drone running the widespread ArduCopter open-source firmware, mounted on-board on a wide range of commercial amateur drones. The results prove that our methodology can efficiently and effectively identify the current state of a powered-on drone, i.e., if it is flying or lying on the ground. In addition, we estimate a lower bound on the time required to identify the status of a drone with the requested level of assurance. The quality and viability of our solution do prove that network traffic analysis can be successfully adopted for drone status identification, and pave the way for future research in the area.

CCS CONCEPTS

•Security and privacy → Mobile and wireless security;

ACM Reference format:
Savio Sciancalepore, Omar Adel Ibrahim, Gabriele Oligeri, Roberto Di Pietro. 2019. Detecting Drones Status via Encrypted Traffic Analysis. In *Proceedings of ACM Workshop on Wireless Security and Machine Learning , Miami, FL, USA, May 15–17, 2019 (WiseML 2019),* 6 pages.
DOI: 10.1145/3324921.3328791

1 INTRODUCTION

Unmanned Aerial Vehicles (UAVs), also known as drones, are becoming extremely popular due to their price getting cheaper and their functionalities becoming increasingly appealing. Indeed, drones are already adopted for several tasks such as inspections, perimeter control, remote assets surveillance, and emergency situations [3]. Unfortunately, drones represent the classical dual-use technology that, while providing great benefits, could also be adopted for malicious intents, such as taking video/image pictures of—or violating [6]—restricted-access areas [13], or even being used-as/carrying weapons against selected targets. The latter one is one of the major threats, not only for people [14] but also for critical infrastructures such as airports and industrial sites, to name a few. The International Air Transportation Association (IATA) warned of "an exponential increase in reports of Remotely Piloted Aircraft systems (RPA) operating dangerously close to manned aircraft and airports" [10]. While self-operated drones represent an expensive solution to deliver attacks and GPS-aided drones can be easily jammed, remotely controlled drones are cheap and can be driven several kilometers away from the operator due to the presence of a First Person View (FPV) channel. At least seven collisions have been reported between aircraft and drones [4], due to the lack of understanding on the drones operators' side. Moreover, drones can also be used to intentionally launch attacks against targets. For instance, an attack was launched in Syria on four Russian military bases by 13 crudely made drones, each equipped with GPS and powered by what appeared to be lawn mower engines, with each one carrying nearly half a kilogram of high potential explosives [4]. While drone counter-measures have already reached a significant level of reliability, drone detection can only rely on few effective techniques [5]. Among the various techniques, such as radar, visual and audio detection, an emerging solution relies on the detection of drones by eavesdropping the spectrum of the Radio Frequencies (RF). RF based techniques resort to the generation of RF fingerprints by looking at the communication channel between the drone and its remote controller [15]. RF fingerprinting is a promising technique that has been used for several purposes, but it requires specific expensive equipment such as Software Defined Radios (SDRs). Indeed, cheap SDRs available on the market, such as the RTL-SDR, cannot be considered fully reliable, especially when operating at high frequencies. Thus, we observe that the actual literature still misses a cheap solution, not requiring any dedicated hardware, while enabling the recognition not only of the presence of a drone, but also its current state in a real-time perspective. Indeed, recognizing if the drone is either flying or lying on the ground could provide additional information to strengthen current security measures and reduce false alarms, triggering countermeasures only if the drone is effectively flying.

Our contribution. The proposed methodology significantly improves the current state of the art in drone detection via the following two contributions: (i) our solution is able to discriminate the current state of a powered-on drone, i.e., if the drone is either flying or lying on the ground—to the best of our knowledge, this is the first solution to achieve this result, and (ii), we provide a statistical analysis of the detection delay for the aforementioned classification scenario, as a function of the requested level of assurance. Results

ACM, 2019. This is personal of copy of the authors. It is posted here by permission of ACM for your personal use. Not for redistribution.
Please cite as: S. Sciancalepore, O. Ibrahim ,G. Oligeri, and R. Di Pietro, Detecting Drones via Encrypted Traffic Analysis, Proceedings of the ACM Workshop on Wireless Security and Machine Learning, Miami FL, USA, 15-17 May 2019. DOI: 10.1145/3324921.3328791.
The definitive version of the paper will be published soon through the ACM Digital Library on https://dl.acm.org.
WiseML 2019, Miami, FL, USA
© 2019 ACM. 978-1-4503-6769-1/19/05…$0.00
DOI: 10.1145/3324921.3328791

show that such a delay is pretty low (less than 4 sec when the 93% of accuracy is required). To the best of our knowledge, the proposed methodology represents the first solution able to detect the current state of a drone in real-time, analyzing the wireless traffic only. The results included in this paper have been obtained by using the popular drones' firmware ArduCopter, within the Ardupilot operating system. Thus, beyond the 3DR SOLO drone used in this paper, our results are fully applicable also to over 20 products, including DJI and HobbyKing vehicles, to name a few.

Paper organization. The paper is organized as follows: Sec. 2 reviews related work, Sec. 3 introduces the system and the adversary models assumed in this work, while Sec. 4 details the measurement scenario and draw some preliminary considerations about the measurements. Sec. 5 provides a characterization of the network traffic generated by the drone, while Sec. 6 introduces the methodology we used for the classification of the network traffic generated by the drone and the remote controller. Section 7 show the performance of our proposal for detecting the state of the drone. Finally, Sec. 8 reports some concluding remarks.

2 RELATED WORK

In the last years, the widespread diffusion of commercial drones has paved the way for several research work discussing the potential identification of UAVs in a certain area of interest.

Authors in [13] built a proof-of-concept system for counter-surveillance against spy drones by determining whether a certain person or object is under aerial surveillance. They show methods that leverage physical stimuli to detect whether the dronefis camera is directed towards a target in real time. They demonstrate how an interceptor can perform a side-channel attack to detect whether a target is being streamed by analyzing the encrypted First-Person View (FPV) channel that is transmitted from a real drone (DJI Mavic) in two use cases: when the target is a private house and when the target is a subject. Although being a significant step towards drone identification, this solution is specifically designed to identify drones that are employed to target a specific asset, while not being suitable for drone's detection at large or for drones that do not feature FPV.

Authors in [19] show that the radio control signal sent to an UAV using a typical transmitter can be captured and analyzed to identify the controlling pilot using machine learning techniques. Authors collected messages exchanged between the drone and the remote controller and used them to train multiple classifiers. They observed that the best performance is reached by a random forest classifier achieving an accuracy of around 90% using simple time-domain features. Extensive tests have shown that the classification accuracy depends on the flight trajectory and that the pitch, roll, yaw, and thrust control signals show different levels of significance for pilot identification. The work focuses on a scenario where civil UAVs are remotely controlled by different pilots, there is no (or weak) authentication on the ground-to-aircraft command channel, and little to null difference in the low-level timing or power of the control signals exist. In addition, the paper assumes pilots carry out identical maneuvers, as well as the existence and availability of trustworthy recordings of each pilotfis behavior. While exploiting the same principle, i.e., classification of the traffic, this paper focuses on the pilot and not on the drone status identification.

Authors in [15] explored the feasibility of RF-based detection of drones by looking at radio physical characteristics of the communication channel when the drones's body is affected by vibration and body shifting. The analysis considered whether the received drone signals are uniquely differentiated from other mobile wireless phenomena such as cars equipped with Wi-Fi or humans carrying a mobile phone. The sensitivity of detection at distances of hundreds of meters as well as the accuracy of the overall detection system are evaluated using a Software Defined Radio (SDR) implementation. Being based on both Received Signal Strength Indicator (RSSI) and phase of the signals, the precision of the approach varies with the distance of the receiver from the transmitter. In addition, the solution resorts to physical layer information and special hardware (SDR), while our current contribution only exploits network layer information that can be collected by any WiFi device.

Fingerprinting of wireless radio traffic at the network layer is emerging as a promising technique to uniquely identify devices in the wild. A fingerprinting approach for drone identification is proposed in [12]. Authors analyzed the WiFi communication protocol used by drones and developed three unique methods to identify a specific drone model: (i) examining the time intervals between probe request frames; (ii) utilizing the signal strength carried in the frame header; and, finally (iii) exploiting some frame header fields with specific values. However, fingerprint approaches require specific equipment to be used, such as the Software Defined Radios (SDRs).

A network-based traffic classification is proposed in [7]. Authors describe a WiFi-based approach aimed at detecting nearby aerial or terrestrial devices by performing statistical fingerprint analysis on wireless traffic. They proved network layer classification as viable means to classify class of drones, i.e., aerial, terrestrial, and hybrid scenarios. However, their solution is only able to identify the presence of a drone and the associated video streaming. In addition, it takes into account a very specific drone based on a proprietary architecture (DJI Parrot), and it is able to detect only the presence of the drone without inferring its current status.

Another passive detection technique is proposed by [8]. Authors presented a technique specifically designed for two reference scenarios: (i) a drone communicating with the ground controller, where the cyclo-stationarity signature of the drone signal and pseudo Doppler principle are employed; and, (ii) a drone that is not sending any signal, where a micro Doppler signature generated by the RF signal is exploited for detection and identification. Also in this case, authors resort to both SDRs and physical layer fingerprinting, thus making their solution very hardware-invasive.

Machine learning techniques have been successfully used for other purposes in this research field. To provide an example, in [16], authors demonstrated that machine learning can successfully predict the transmission patterns in drone network. The packet transmission rates of a communication network with twenty drones were simulated, of which results were used to train the linear regression and Support Vector Machine with Quadratic Kernel (SVM-QK).

Standard anti-drone active detection techniques resort to radar [9]. Nevertheless, those techniques involve the transmission of signals and specific devices for the detection of the echo fingerprint.

Finally, authors in [11] analyze the basic architecture of a drone and propose a generic drone forensic model that would improve the digital investigation process. They also provide recommendations on how one should perform forensics on the various components of a drone such as camera and Wi-Fi.

To sum up, none of the previous contributions can detect the current status of a drone by only exploiting the wireless traffic. Moreover, differently from the current literature, our contribution provides a thorough measurement campaign adopting a widely accepted open-source operating system and firmware for drones, as well as a an analysis of the detection delay in relation to a given performance target.

3 SYSTEM AND ADVERSARY MODEL

Adversarial model. We envisage a scenario characterized by a drone located in a no-fly-zone, where the GPS system is not available. Indeed, we recall that drones leveraging GPS navigation can be easily defeated by adopting GPS-spoofing and jamming techniques. We assume the drone is remotely controlled by a malicious operator that intentionally wants to fly the drone across the border of a restricted-access area such as an airport, industrial plant, or critical infrastructure. We also assume that the adversary is deploying additional countermeasures preventing the drone identification, such as dynamically changing the MAC addresses of the network interfaces of both the drone and the remote controller. In addition, we assume the link between the RC and the drone is encrypted at the layer-2 of the communication link, and therefore packet content cannot be inspected. We assume also that the link between the controller and the drone cannot be jammed, as the same frequencies are used for legitimate communications by other devices.

Finally, we assume the adversary does not apply any evasion techniques, e.g., it does not modify the transmission rate and the size of the packets in order to mitigate the detection. While implementing such countermeasures might improve the probability to escape detection, its overall efficacy is not guaranteed, as a new training of our model would suffice to identify again the status of the drone. However, the application of evasion strategies requires further feasibility studies based on the specific requirements of the application [22]. In addition, none of the actual commercial products implements such features, making our proposed solution effective.

System model. Our main goal is the passive detection of the drone status without resorting to: (i) active radar technology [9]; (ii) visual detection [18]; or, (iii) physical stimuli to FPV [13]. Our solution does not require any intervention in the already existing ICT infrastructure and it does not conflict with any already deployed RF system. Indeed, the proposed methodology exploits only the messages transmitted between the remote controller and the drone, and therefore it only requires a fully passive eavesdropper to be deployed in the region to be controlled.

Figure 1 wraps up on the system and adversarial models: the adversary (A) is determined to remotely fly a drone (D) into a no-fly zone (CI). Our solution is able to detect the drone's status by simply deploying a WiFi probe (P). We observe that the WiFi probe is able to eavesdrop both the traffic from the controller to the drone (A-D) and the one from the drone to the controller (D-A).

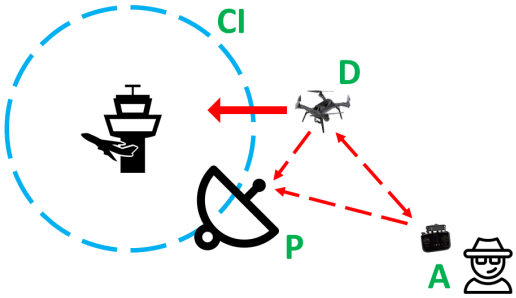

Figure 1: System model: a no-fly zone (e.g. critical infrastructure — CI) featuring a WiFi probe (P) to detect an approaching drone (D) remotely controlled by an adversary (A).

.

4 MEASUREMENT SCENARIO AND PRELIMINARY CONSIDERATIONS

Our measurement scenario is constituted by a 3DR SOLO drone[1] and a wireless probe capable of eavesdropping the radio traffic. The 3DR Solo drone is an open-source architecture featuring the Pixhawk 2.0 flight controller and the ArduCopter 3.3 firmware. The drone has been configured for the *manual mode*, i.e., GPS switched off, and therefore, being able to fly in indoor environments. As a wireless probe, we adopted Wireshark 2.4.5, running in a Lenovo Ideapad 320 featuring Linux Kali 4.15.0. We configured the WiFi card of our laptop to work in monitor mode, being able to eavesdrop and log all the transmitted packets by either the remote controller or the drone, by periodically scanning all the WiFi spectrum. Figure 2 shows our measurement set-up.

Figure 2: Our measurement set-up: the drone, the remote controller, and the laptop we used to eavesdrop the radio spectrum.

.

Subsequently, we collected several packets from the controller-drone link, while the drone was performing two different types of actions, as depicted below:

- *Steady* (S). The drone is associated to the remote controller but it lays on the ground.

- *Movement* (M). The drone is flying around, performing several movements in all directions.

Table 1 wraps up on the two states of the drone and provides ground for some preliminary considerations on the collected measurements. Firstly, we break down the communication link into 4 different flows: (i) the packets sent by the RC to the Drone; (ii) the packets sent by the Drone to the RC; (iii) the packets broadcast by the RC; and, finally (iv) the packets broadcast by the drone. Secondly, we considered a measurement lasting for about 10 minutes for the Steady state, where the drone is associated to the remote controller but it is lying on the ground; then, we unscrewed the propellers from the drone and we "flew" the drone for about 10 minutes for the Movement state. During the flight, we performed different maneuvers by continuously moving the control sticks. The percentage of exchanged packets is similar in both the drone's states, i.e., about 35% of the traffic is transmitted by the RC, while about 58% of the traffic is received by the RC. Finally, we observe the presence of broadcast traffic transmitted by the RC (about 6%) while we did not detect any broadcast communication transmitted by the drone after the pairing process.

Table 1: Overview of the Drone's states and flows.

State	Source	Dest.	Pkts No.	Flow/Link (%)
Steady (S)	RC	Drone	32706	35.8
	Drone	RC	52856	57.8
	RC	Broadcast	5789	6.3
	Drone	Broadcast	0	0
Movement (M)	RC	Drone	32868	34.6
	Drone	RC	56248	59.2
	RC	Broadcast	5837	6.1
	Drone	Broadcast	0	0

5 LINK FLOW STATISTICS

In the following analysis, we focus on two packet attributes: *packet size* and *packet inter-arrival time*. For both the two attributes, we took the previously identified flows from Sec. 4, and we analyzed the packet sizes for the 4 different flows combining the two states, i.e., Steady and Movement, and the packets travelling from the RC to drone, and from the drone back to the RC.

Packet size analysis. Figure 3 shows the frequency distribution function associated with each packet size belonging to the 4 different flows. Firstly, the vast majority of packets transmitted in the configuration *S, RC to drone* (red circles) has the size equal to 156 Bytes. A similar phenomenon can be observed for *M, RC to drone* (black crosses). Conversely, the flows coming from the drone are characterized by very different packet sizes spanning from 130 to 1144 Bytes, while more packets are transmitted by the drone when it is in the Steady state, i.e., blue crosses have on average higher values than the green stars.

Inter-arrival time analysis. We extracted the time associated with all the events belonging to the same flow and we differentiated them obtaining the inter-arrival times. Figure 4 shows the number of packets as a function of their inter-arrival time, while Fig. 5 is a magnification of the time spanning between $1\mu s$ seconds and

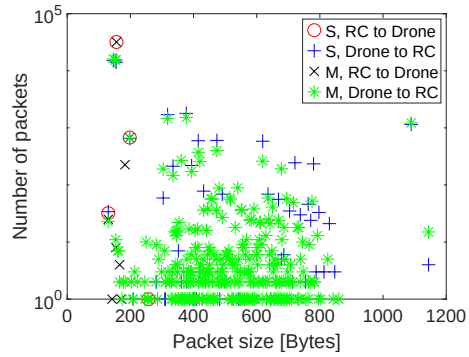

Figure 3: Packet size analysis: Number of packets as a function of their size considering the four communication flows.

$10ms$, using a logarithmic scale. Firstly, we observe periodic packets at $20ms$ and $40ms$ transmitted by the RC to the drone in both the drone's states. Secondly, we also observe periodic phenomena from the drone to the RC every $32ms$ and $50ms$. A similar phenomenon can be observed for the Movement state. Finally, in Fig. 5 we consider an in-depth analysis for the inter-arrival time values less than $10ms$. We observe how the two drone's states, i.e., Steady and Movement, are characterized by almost the same profile: there are only minor differences at about $31\mu s$ and between $50\mu s$ and $90\mu s$. Indeed, the correlation coefficients computed over the frequency distribution functions are 0.71 and 0.75, for the flows transmitted and received by the remote controller, respectively. Finally, for both the considered drone's state, we observe that the drone is transmitting more data then the RC, i.e., blue and green curves are higher than the black and red ones.

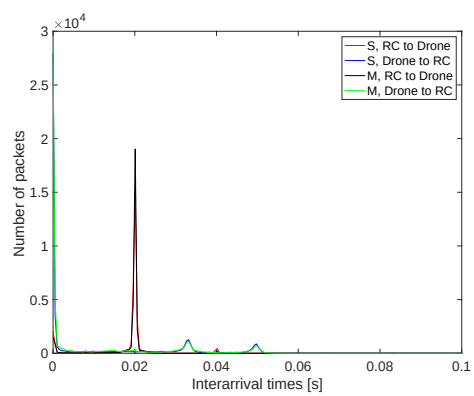

Figure 4: Packet interarrival time analysis: Number of packets as a function of the interarrival time considering the 4 communication flows.

Broadcast traffic. Packets transmitted to the broadcast address by the remote controller have the same size (289 bytes) and an inter-arrival time equal to $100ms$ in both the drone's state. Given their strictly periodic nature, we do not consider them for our analysis.

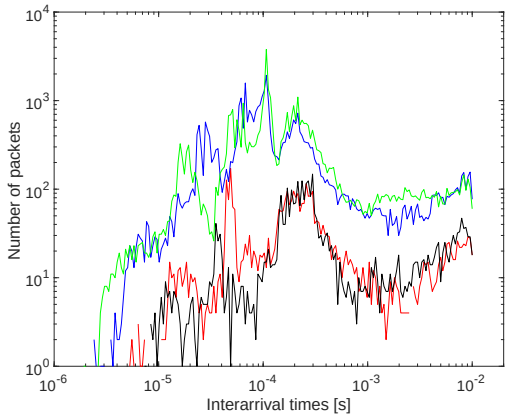

Figure 5: Packet interarrival time analysis for the time frame spanning between 10^{-6} and 10^{-2} seconds.

6 CLASSIFICATION METHODOLOGY

In this section, we introduce our solution to drone's status detection, and we test it against the previously introduced drone's states, i.e., Steady and Movement. As our main classification tool, we adopted WEKA [21]. WEKA is a collection of machine learning algorithms for data mining tasks. It contains tools for data preparation, classification, regression, clustering, association rules mining, and visualization. For our analysis, we consider the following configuration:

- *Classifiers.* We run 3 different algorithms: Trees-J48 (J48), Random Forest (RF), and Neural Networks (NN).
- *Flows.* We consider 3 different flows, i.e., RC to drone, drone to RC, and the overall link.
- *Features.* We look at 6 different features, i.e., interarrival time, packet size, mean and standard deviation computed over a certain number of samples of interarrival time and packet size.

Tree-J48 (J48) is an implementation of the C4.5 decision tree algorithm, representing a predictive model mapped to a tree structure to achieve perfect classification with minimal number of decisions [20]. The Random forest (RF) classifier is a term for an ensemble classifier consisting of many decision trees giving as outputs the mode of the output by each individual trees [17]. Finally, the Neural Network (NN) algorithm is based on a MultilayerPerceptron [2]. For all the algorithms we used the 10-folds cross-validation method, i.e., a technique to evaluate the classifier performance by partitioning the original sample into a training set (9 randomly chosen folds) to train the model, and a test set to evaluate it (the remaining fold from the 9 previously chosen). Figure 6 wraps up the details of our methodology to detect the status of the drone. We assume a communication link between the drone (D) and the adversary (A) constituted by two flows: remote controller to drone and backwards. Both the flows are eavesdropped by a WiFi probe collecting only two attributes: interarrival time between subsequent packets and packet size. Subsequently, a set of features is generated from the attributes. Indeed, for each instance of the attributes, i.e., each pair

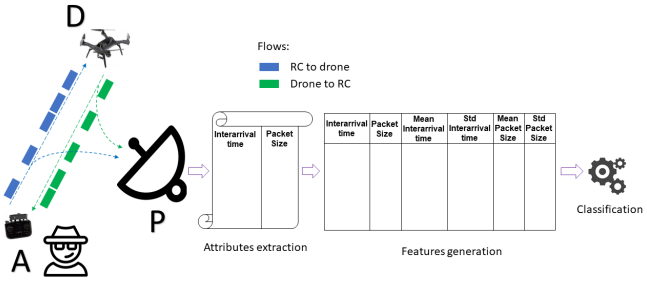

Figure 6: Classification methodology: WiFi radio eavesdropping, attribute extraction, feature generation, and classification.

of [interarrival time - packet size], we compute the mean and standard deviation on a predetermined sequence of subsequent values for both the interarrival time and the packet size. The new data set of features is then provided to the classifier.

7 DRONE SCENARIO IDENTIFICATION: IS IT FLYING?

In this section, we introduce the methodology used to identify the current state of the drone, i.e., Steady or Movement. Therefore, we assume the defense system is eavesdropping the WiFi spectrum, collecting packets and generating the associated features as already discussed in Sec. 6. For each instance in the features set, we challenge the classifier to identify the state of the drone. In particular, we make the hypothesis that *the current instance belongs to the Movement state* and we consider 5 different metrics: True Positive (TP), False Positive (FP), False Negative (FN), True Negative (TN), and the overall Success Rate (SR) as the number of correct classifications (TP+TN) divided by the total number of classifications (TP+TN+FP+FN). Table 2 depicts the results of the 10-fold cross-validation, assuming the three aforementioned classifiers and metrics with the features computed over 200 consecutive samples. Firstly, we observe that both J48 and RF behave better than NN: TP and TN exceed 89% for both the flows. For all the cases, link fingerprinting achieves the worst performance, while the flows, i.e., RC to Drone and Drone to RC, behave almost in the same way. Considering the best performers (J48 and RF), and the representative link drone to RC, we observe that both FP and FN are less than 8%. Finally, for each configuration we report the overall SR being equal to the total number of correctly classified instances.

Detection delay. We now consider the time to detect the state of the drone. Given the results of the previous section, we assume the Random Forest classifier (RF) using the 10-folds cross-validation method and the Drone to RC link. Figure 7 shows the Receiver Operating Characteristic (ROC) curve associated with the aforementioned configuration while varying the number of samples used for the generation of the features. Indeed, as in the previous case, we considered the inter-arrival time, the packet size, and their mean and standard deviation computed over different set sizes spanning between 50 and 500 samples.

Increasing the number of samples significantly improves the performance of the detection process. Errorbars in the inset figure

Table 2: Drone's Status Detection performance overview. Reported values are all in percentages.

Classif.	Flow	TP	FP	FN	TN	SR
J48	RC to Drone	93.72	10.44	6.28	89.56	91.54
	Drone to RC	93.16	7.88	6.84	92.12	92.65
	Link	85.63	11.95	14.37	88.05	86.77
RF	RC to Drone	92.37	8.65	7.63	91.35	91.86
	Drone to RC	92.98	6.89	7.02	93.11	93.05
	Link	88.15	8.57	11.85	91.43	89.69
NN	RC to Drone	80.82	24.52	19.18	75.48	77.91
	Drone to RC	80.21	30.14	19.79	69.86	74.29
	Link	69.93	30.77	30.08	69.23	69.59

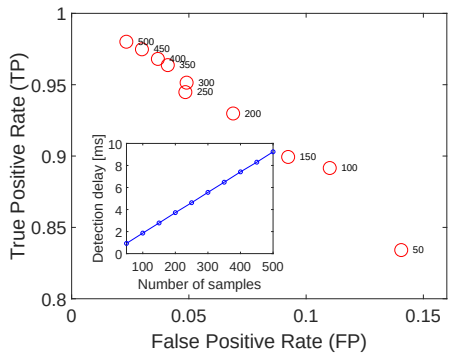

Figure 7: Relationship between the number of samples and the detection delay.

show the quantile 5, 50, and 95 associated with the detection delay as a function of the number of samples. Given the linear relationship between the number of samples and the detection delay, a good trade-off between detection performance and delay can be estimated in 200 samples (FP=0.07, TP=0.93) being equivalent to about 3.71 seconds of eavesdropping time. Better performance can be achieved using 400 samples (FP=0.04, TP=0.97), while incurring in a detection delay of about 7.42s.

8 CONCLUSIONS AND FUTURE WORK

In this paper, we have introduced a novel methodology to detect whether a drone is either flying or lying on the ground with a high degree of assurance and a very short delay. These results are achieved without resorting to any active techniques, but just eavesdropping the radio traffic and processing it with standard ML techniques. In particular, we proved that network traffic classification can be effectively used to detect the status of drones employing the popular operating system ArduCopter (such as some DJI and Hobbyking vehicles). Moreover, we also provide a lower bound on the detection delay when using the aforementioned methodology: our solution is able to discriminate the state of the drone, i.e., steady or movement, in about 3.71 seconds with a SR of about 0.93.

We believe that our methodology can have more general applications and, given that our study has been performed on a popular open-source operating system for drones, it can be used to detect and identify different brands and models of drones.

ACKNOWLEDGEMENTS

This publication was partially supported by awards NPRP-S-11-0109-180242, UREP23-065-1-014, and NPRP X-063-1-014 from the QNRF-Qatar National Research Fund, a member of The Qatar Foundation. The information and views set out in this publication are those of the authors and do not necessarily reflect the official opinion of the QNRF.

REFERENCES

[1] 3DR SOLO Website. 2019. https://3dr.com/solo-drone. (2019). Accessed: 15-04-2019.
[2] A. A. Ali and R. Tervo. 2001. Traffic identification using artificial neural network [Internet traffic]. In *Canadian Conference on Electrical and Computer Engineering 2001. Conference Proceedings (Cat. No.01TH8555)*, Vol. 1. 667–672 vol.1.
[3] Riham Altawy and Amr M. Youssef. 2016. Security, Privacy, and Safety Aspects of Civilian Drones: A Survey. *ACM Transactions on Cyber-Physical Systems* 1, 2 (Nov. 2016), 7:1–7:25.
[4] ASI. 2018. https://www.asi-mag.com/drones-protecting-airports-and-aircraft/, last accessed on 15-04-2019. (2018).
[5] M. M. Azari, H. Sallouha, A. Chiumento, S. Rajendran, E. Vinogradov, and S. Pollin. 2018. Key Technologies and System Trade-offs for Detection and Localization of Amateur Drones. *IEEE Communications Magazine* 56, 1 (Jan 2018), 51–57.
[6] BBC. 2019. http://www.bbc.co.uk/news/uk-england-sussex-46623754, last accessed on 15-04-2019. (2019).
[7] I. Bisio, C. Garibotto, F. Lavagetto, A. Sciarrone, and S. Zappatore. 2018. Blind Detection: Advanced Techniques for WiFi-based Drone Surveillance. *IEEE Trans. on Vehicular Technology* (2018), 1–1.
[8] H. Fu, S. Abeywickrama, L. Zhang, and C. Yuen. 2018. Low-Complexity Portable Passive Drone Surveillance via SDR-Based Signal Processing. *IEEE Communications Magazine* 56, 4 (Apr. 2018), 112–118.
[9] F. Hoffmann, M. Ritchie, F. Fioranelli, A. Charlish, and H. Griffiths. 2016. Micro-Doppler based detection and tracking of UAVs with multistatic radar. In *IEEE Radar Conf. (RadarConf)*. 1–6.
[10] IATA. 2019. https://www.iata.org/whatwedo/ops-infra/air-traffic-management/Pages/remotely-piloted-aircraft-systems.aspx, last accessed on 15-04-2019. (2019).
[11] U. Jain, M. Rogers, and E. T. Matson. 2017. Drone forensic framework: Sensor and data identification and verification. In *IEEE Sensors Applications Symposium (SAS)*. 1–6. DOI : https://doi.org/10.1109/SAS.2017.7894059
[12] H. Li, G. Johnson, M. Jennings, and Y. Dong. 2017. Drone profiling through wireless fingerprinting. In *IEEE International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*. 858–863.
[13] B. Nassi, R. Ben-Netanel, A. Shamir, and Y. Elovici. 2019. Drones' Cryptanalysis - Smashing Cryptography with a Flicker. In *IEEE Symposium on Security and Privacy (SP)*, Vol. 00. 833–850.
[14] New York Times. 2018. https://www.nytimes.com/2018/08/10/world/americas/venezuela-video-analysis.html, last accessed on 15-04-2019. (2018).
[15] P. Nguyen, H. Truong, M. Ravindranathan, A. Nguyen, R. Han, and T. Vu. 2017. Matthan: Drone Presence Detection by Identifying Physical Signatures in the Drone's RF Communication. In *Proc. of the Annual Int. Conf. on Mobile Systems, Applications, and Services (MobiSys '17)*. 211–224.
[16] J. Park, Y. Kim, and J. Seok. 2016. Prediction of information propagation in a drone network by using machine learning. In *Int. Conf. on Informat. and Communicat. Technol. Converg. (ICTC)*. 147–149.
[17] A. Paul, D. P. Mukherjee, P. Das, A. Gangopadhyay, A. R. Chintha, and S. Kundu. 2018. Improved Random Forest for Classification. *IEEE Trans. on Image Processing* 27, 8 (Aug. 2018), 4012–4024.
[18] M. Saqib, S. Daud Khan, N. Sharma, and M. Blumenstein. 2017. A study on detecting drones using deep convolutional neural networks. In *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. 1–5.
[19] A. Shoufan, H. M. Al-Angari, M. F. A. Sheikh, and E. Damiani. 2018. Drone Pilot Identification by Classifying Radio-Control Signals. *IEEE Transactions on Information Forensics and Security* 13, 10 (Oct 2018), 2439–2447.
[20] P. H. Swain and H. Hauska. 1977. The decision tree classifier: Design and potential. *IEEE Trans. on Geoscience Electron.* 15, 3 (Jul. 1977), 142–147.
[21] Ian H. Witten, Eibe Frank, and Mark A. Hall. 2011. *Data Mining: Practical Machine Learning Tools and Techniques* (3rd ed.). M. Kaufmann Pub. Inc.
[22] Fei Zhang, Patrick PK Chan, Battista Biggio, Daniel S Yeung, and Fabio Roli. 2016. Adversarial feature selection against evasion attacks. *IEEE Transactions on Cybernetics* 46, 3 (2016), 766–777.