HAL
archives-ouvertes.fr

A Hierarchical Detection and Response System to Enhance Security Against Lethal Cyber-Attacks in UAV Networks

Hichem Sedjelmaci, Sidi Mohammed Senouci, Nirwan Ansari

▶ To cite this version:

Hichem Sedjelmaci, Sidi Mohammed Senouci, Nirwan Ansari. A Hierarchical Detection and Response System to Enhance Security Against Lethal Cyber-Attacks in UAV Networks. IEEE Transactions on Systems, Man, and Cybernetics: Systems, IEEE, In press, http://ieeexplore.ieee.org/document/7890467/ . 10.1109/TSMC.2017.2681698 . hal-01557953

HAL Id: hal-01557953
https://hal-univ-bourgogne.archives-ouvertes.fr/hal-01557953

Submitted on 19 Feb 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Hierarchical Detection and Response System to Enhance Security Against Lethal Cyber-Attacks in UAV Networks

Hichem Sedjelmaci, *Member, IEEE*, Sidi Mohammed Senouci, *Member, IEEE*, and Nirwan Ansari, *Fellow, IEEE*

***Abstract*—Unmanned aerial vehicles (UAVs) networks have not yet received considerable research attention. Specifically, security issues are a major concern because such networks, which carry vital information, are prone to various attacks. In this paper, we design and implement a novel intrusion detection and response scheme, which operates at the UAV and ground station levels, to detect malicious anomalies that threaten the network. In this scheme, a set of detection and response techniques are proposed to monitor the UAV behaviors and categorize them into the appropriate list (normal, abnormal, suspect, and malicious) according to the detected cyber-attack. We focus on the most lethal cyber-attacks that can target an UAV network, namely, false information dissemination, GPS spoofing, jamming, and black hole and gray hole attacks. Extensive simulations confirm that the proposed scheme performs well in terms of attack detection even with a large number of UAVs and attackers since it exhibits a high detection rate, a low number of false positives, and prompt detection with a low communication overhead.**

***Index Terms*—Anomaly detection and rules-based intrusion detection techniques, cyber-attacks, intrusion detection system (IDS), unmanned aerial vehicles (UAVs).**

I. Introduction

UNMANNED aerial vehicles (UAVs) have initially been utilized in military applications to engage in air-to-ground combats, surveillance, and target tracking in hostile environments. Surveillance primarily concerns collection, analysis, and management of critical information in critical sites (airport area, nuclear site, etc.). Tracking is the operation of following mobile targets (suspected persons or vehicles) and monitoring their behaviors. Nowadays, UAVs are also used in civil applications to explore inaccessible zones (e.g., disaster areas) and deliver data to and from areas with no network infrastructure (3G, 4G, etc.) [1]. An UAV network is a wireless ad-hoc network that facilitates UAV-to-UAV and/or UAV-to-ground communications in order to deliver vital information for environmental monitoring, emergency, rescue and recovery operations, and disaster assistance. Setting up an ad-hoc network consisting of UAVs is very challenging because they differ from mobile ad-hoc networks (MANETs) and vehicular ad-hoc networks in terms of mobility, connectivity, routing, services, and applications. Owing to node mobility, rapid topology changes and sparse communications, a delay tolerant network (DTN) based on a "store, carry-and-forward" mechanism [2], where a node stores and carries a message until a suitable next node is found, is adopted in UAV networks.

Security is another major challenging issue due to the wireless medium characteristics and the relevant information handled by UAVs. Cryptography and intrusion detection system (IDS) are two major security mechanisms. On one hand, cryptography is used to ensure message privacy and node authentication, and is used to prevent external intruders to penetrate the network. On the other hand, IDS uses special agents to analyze the misbehavior of a monitored node [3], [4]. IDS is effective in protecting the network against both internal and external intruders [3]–[5]. Furthermore, the IDS relies mainly on two detection techniques [5].

1) *Anomaly detection*, which builds a model of normal profiles and attempts to track deviations from normal behavior that may be subject to anomalies or possible intrusions. Though this technique may detect new attacks that have not been previously observed by the system, it is computationally costly. Anomaly detection usually uses a learning algorithm such as neural networks and support vector machines (SVMs) [6] to detect the anomaly behavior of a monitored node.
2) *Rules-based intrusion detection*, which compares the behavior of the monitored node against a set of rules related to behaviors of specific known attacks. These rules are defined by a set of attack signatures [7].

In this case, updates of the attack signatures are required on a permanent basis. Deploying IDS in a DTN, like an UAV network, is challenging because it is difficult for a set of IDS nodes to monitor the behaviors of too few nodes spread across a large geographical area. Thereby, the most appropriate approach is a distributed IDS solution where all nodes throughout the network activate their IDS agents by performing promiscuous monitoring [8] to observe its neighborhood activities. Furthermore, determining the malice of a node in a sparse network-based only on few (one or two) recommendations

H. Sedjelmaci and S. M. Senouci are with the DRIVE Laboratory, University of Burgundy, 58000 Nevers, France (e-mail: sid-ahmed-hichem.sedjelmaci@u-bourgogne.fr).

N. Ansari is with the New Jersey Institute of Technology, Newark, NJ 07102 USA (e-mail: nirwan.ansari@njit.edu).

is difficult. In fact, this decision should be made, based on the history of the node's activities, by a centralized trusted node. Therefore, a hierarchical intrusion monitoring and decision process is the most appropriate solution in a DTN, where different detection and response techniques run at two layers, UAV and ground station.

Existing security mechanisms [9]–[11] applied to UAV networks are based on cryptography to ensure message privacy and node authentication. Specifically, Strohmeier *et al.* [9] and Wesson *et al.* [10] proposed an authentication-based solution to authenticate and ensure the privacy of messages broadcasted by the automatic dependent surveillance-broadcast (ADS-B) component, which is an on-board component part of the UAV system; ADS-B broadcasts critical information, such as position, heading, speed, and collision avoidance. Detecting attacks in UAV networks has not been well addressed in the literature. To the best of our knowledge, the intrusion detection framework designed by Mitchell and Chen [12] is the only publicly available work that relies on detection techniques to protect such networks. In this paper, the normal behavior of an UAV is modeled with a set of rules, where the IDS agent should wade all the rules and exchange them with its neighbors to detect malicious anomalies. This IDS solution incurs a high communication overhead. In addition, according to their simulation results, the system incurs high false positives. In this schema, the intrusion detection techniques proposed in MANET is applied directly in the UAV network without taking into account of the UAV network's requirements such as mobility of nodes and energy constraints. Thereby, in this paper, we propose a novel intrusion detection and response scheme that aims to detect the most lethal cyber-attacks that can target an UAV network, including: false information dissemination, GPS spoofing, jamming, and gray hole and black hole attacks. Our scheme alleviates the drawbacks of the scheme proposed in [12] because it is fast in terms of attack detection, lightweight in terms of communications overhead, scalable, and achieves a high accurate detection rate. In addition and unlike [12], it takes into account of the UAV network's requirements.

This paper work focuses on UAV-based civilian applications where UAVs explore an isolated zone to collect and transmit critical information to a remote ground station for analysis and decision processes. The proposed hierarchical detection and response scheme is running at the UAV and ground station levels to detect any malicious anomalies that threaten the network. To achieve high accuracy, the hierarchical scheme combines rules-based detection and anomaly detection techniques. With the help of these detection techniques, we also develop a new response scheme that categorizes the monitored UAVs into appropriate lists (normal, suspect, abnormal, and malicious) according to their behaviors. Our IDS-based solution achieves the following characteristics.

1) Smart activation of the intrusion monitoring process: in fact, when a large number of nodes launch their monitoring processes, the incurred overhead can be substantial; therefore, a tradeoff between the intrusion detection rate and overhead is considered in this paper.

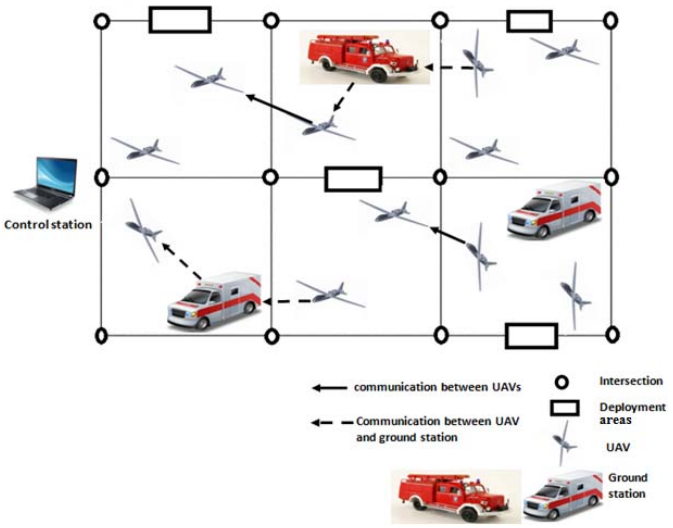


Fig. 1. UAVs deployment and exploration with ground stations deployment.

2) The embedded IDS agents react against specific UAV threats and are incorporated into the protocols used in this network.
3) UAV and IDS are fully trusted.

The rest of this paper is organized as follows. In Section II, we present the network model that we attempt to secure along with an overview of the most important attacks observed in UAV networks. Section III describes our proposed intrusion detection and response scheme, and Section IV presents NS3 simulation results and analysis. In Section V, we highlight the related works. Finally, the conclusion and discussion on future works are presented in Section VI.

II. Network Model and Security Objectives

This section is organized into two sections: the network model and the different attacks we aim to identify, respectively.

A. Network Model

This paper focuses on civilian applications, where UAVs are envisioned to carry out explorations in isolated zones (e.g., disaster areas) to collect and transmit critical information about detected events to a remote control station for further actions as shown in Fig. 1. We suppose that each UAV follows a *deterministic mobility model* [13] to explore a zone as detailed in Section IV-A. The communication mode is either between UAVs (UAV-to-UAV) or between UAV and the ground station (UAV-to-ground). In this paper, we target UAV applications in case of a disaster (e.g., tsunamis, volcanic eruptions, etc.), which are time-sensitive applications. To best ensure this time constraint in such zero-infrastructure environment, we propose to form an opportunistic network composed of drones and assisted by a set of ground stations. Within this network, it is impossible to have persistent connections between UAVs or between UAV and ground station unless we put drones and ground stations everywhere across the field, which will, however, incur insurmountable cost as well as safety concerns (the ground stations cannot be deployed everywhere

in a disaster area, e.g., the middle of a fire, tsunamis). In time-sensitive applications, a DTN based on "store-carry-and-forward" mechanism [2] is used to decrease the packets loss in a hop-by-hop manner. When neither UAVs nor ground stations are available, data are buffered in the UAV until a neighbor becomes available. The "store-carry-and-forward" mechanism is also called opportunistic data forwarding [14], where nodes store, carry, and forward packets to target nodes.

A number of DTN mechanisms have been proposed to address communications disconnections and improve the network delay [14], [15]. In this paper, inspired by the DTN routing protocol proposed by Lu *et al.* [14], we propose to use a ground station as a relay node when the next-hop UAV is not available. Note that the ground station could be for instance an emergency vehicle (ambulance, police car, etc.) and is part of the network. These kinds of vehicles are assumed to be relatively static since they are not mobile in case of disasters, i.e., they are quickly deployed and remain stationary for a relatively long period of time in, e.g., hours or days. In addition, they are assumed to be trusted nodes, and possess higher computational capabilities as compared to UAVs. They are able to communicate between each other using secure digital professional mobile radios like TETRAPOL [16]. As illustrated in Fig. 1, UAV fleets are launched from some deployment areas to explore an assigned region. They return back to these departure points to recharge their batteries when their remaining energy is not sufficient to continue the exploration. As shown in Fig. 1, ground stations are deployed at sites, where there are not enough UAVs to guarantee full coverage, to help data forwarding. In the following, we discuss the ground stations deployment and explain how data are forwarded in this DTN.

1) Ground Stations Deployment: The optimal deployment of the ground stations is achieved with the help of a graph-based model [17]. Here, our aim is to determine a low degree of intersection vertices (see Fig. 1) where the ground stations are deployed to help forwarding data in case the next-hop UAV is unavailable. The low degree of intersection vertices refers to a region with a low distribution of UAVs. Denote $U = \{u_1, u_2, u_3, \ldots\}$ as the set of UAVs that will explore the monitored zone, and $\phi = \{V, \delta\}$ as a directed random graph where $V = \{v_1, v_2, v_3, \ldots\}$ is the set of intersection vertices and δ is the set of directed edges between any two directly connected vertices. For any edge $e_{i,j} \in \delta$ from v_i to v_j, where $i \neq j$, the flow rate of UAVs traversing the edge is denoted as $G(e_{i,j}) = \lambda_{i,j}$ if v_i and v_j are directly connected, and $G(e_{i,j}) = 0$ otherwise [14]. We assume the arrivals of UAVs traversing the edge/link ($v_i \longrightarrow v_j$) is Poisson, from which its parameter, $\lambda_{i,j}$, can realistically be captured by counting the average number of UAVs passing from v_i to v_j per unit of time.

As in [14] and because all UAVs $U = \{u_1, u_2, u_3, \ldots\}$ follow the shortest path, the degree of vertex $v_i \in V$ is defined as:

$$D_i = \frac{G(v_i)}{\sum_{u_j \in U} \xi_j} = \frac{\sum_{v_j \in V} \lambda_{j,i}}{\sum_{u_j \in U} \xi_j} \tag{1}$$

Algorithm 1 Packet Forwarding in UAV Network

1: **Begin**
2: UAV u_i stores, carries a message M for a period of time (Ts), and tries to forward M to the next-hop UAV using a greedy forwarding mechanism within Ts.
3: ***if*** (u_i detects a nearby UAV u_j) ***then***
4: u_i forwards M to u_j.
5: ***else if*** (u_i detects a nearby ground station v_{i}) ***then***
6: u_i forwards M to v_i.
7: ***else if*** (Ts is elapsed and no next-hop is available) ***then***
8: M is removed from $u_i{}'$s buffer.
9: ***end***

where ξ_j is the number of shortest paths that UAV $u_j \in U$ could explore (depending on the remaining energy) and $G(\mathcal{V}_i)$ is the flow rate of intersection vertex v_i, which is equal to $\sum_{v_j \in V} G(e_{j,i}) = \sum_{v_j \in V} \lambda_{j,i}$.

As a result, the *set of "low degree" intersection vertices*, where the ground stations are deployed, is determined according to (2). Here, TR denotes the threshold, which is adjusted to allow the ground stations to have a high capability to assist UAVs to store-and-forward packets in the network. Note that, for safety purposes, the ground stations cannot be deployed everywhere in a disaster area, e.g., in the middle of a fire, tsunamis, or volcano even if (2) holds

$$\mathrm{LIV} = \left\{ v_i \in V \middle| D_i < \mathrm{TR} \right\}. \tag{2}$$

2) Data Forwarding: When the UAV detects an event (e.g., critical disaster), it forwards a message that contains the event type to the final destination (i.e., control station). This forwarding is done hop by hop using a greedy forwarding mechanism [18]–[20]. In the latter, the next-hop (UAV or ground station) is selected as the farthest one from the sender [20]. In case neither other UAVs nor ground stations are available, the UAV stores and carries the message for a period of Ts. In case when Ts has elapsed and no next-hop is available, the message is removed from the UAV's buffer. The packet forwarding in UAV network is illustrated in Algorithm 1.

B. Common UAVs Cyber-Attacks

Owing to the nature of wireless medium and relevant data handled by UAVs, securing UAV networks becomes vital to protect them against lethal cyber-attacks. In this paper, we focus on mitigating two types of cyber-attacks: 1) integrity and 2) denial of service (DoS) attacks.

1) Integrity Attacks: Such attacks aim to fabricate false information, i.e., altering GPS coordinates or disseminating false information.

a) GPS spoofing attacks: The GPS receiver of a UAV can be spoofed by an attacker, thus leading to a false estimate of the drone position. For instance, the spoofed signal can lead to the situation where the receiver estimates the UAV's position, which is different from the real position. It has been claimed that the capture of a military drone aircraft by the Iranian military in December 2011 was the result of such attack [11]. Furthermore, the Iranian hacker used a jamming attack to jam signals between the controller and UAV that caused the UAV

to switch to the autopilot mode. The latter relied on GPS coordinates to guide itself back to its home base. Afterward, the hacker launched a GPS spoofing attack to falsify GPS coordinates and led the UAV to think that it was close to the home base. Recently, Wesson [21] described the procedure to stage a GPS spoofing attack on a civilian UAV: 1) precisely align a spoofed (counterfeit) GPS signal with the authentic signal generated by the satellite at the target node; 2) gradually increase the counterfeit signal strength to get control of the target node; and 3) move the counterfeit signal slowly away from the authentic one. At the end, the attacker has a complete control of the legitimate UAV. Such attack is simple to implement and inexpensive [22]. Furthermore, according to several research works [21], [23], it is difficult to detect such attack since the spoofer triggers no alarms on the ship's navigation component.

b) False information dissemination attacks: A malicious UAV could broadcast a different physical phenomenon such as environmental conditions or forest fires to its neighbors. Such attack is defined as a *false data injection attack*. The ADS-B attack [10] is another kind of attacks that aims to disseminate false information. ADS-B is an on-board component of the UAV system that broadcasts information such as position and collision avoidance [24]. According to [10] and [24], an *ADS-B attack* either aims to broadcast a false position or spoof the GPS coordinates (i.e., GPS spoofing) of a target UAV. Therefore, the survivability of legitimate drones is affected. A malicious intrusion detection agent, referred to as UDA[1], could also provide false detection information to degrade the network performances. In Section III, we give more details about UDA and on how this agent can carry out a monitoring process. Such misbehavior is categorized into two kinds of threats [12]: *bad-mouthing*, i.e., UDA claims that a well-behaved node is malicious or *good-mouthing*, i.e., UDA provides good recommendations regarding a malicious node.

2) DoS Attacks: The malicious node that executes a DoS attack attempts to exhaust energy resources of UAVs or disturb the network and routing protocol [25]–[27]. Jamming and gray hole and black hole attacks are among the major lethal DoS attacks.

a) Jamming attacks: Jamming aims to jam the communications between the controller and UAV, and to take control of a target UAV by launching another kind of attack such as GPS spoofing. We will consider two kinds of jamming attacks: 1) deceptive and 2) random. The former constantly broadcasts packets without any gap between subsequent packets [28]. However, a *random* jammer alternates between sleeping and jamming [28]. During the jamming, it can play the role of a *deceptive* jammer.

b) Gray hole and black hole attacks: In DTN, the malicious node first lures packets by claiming that it can help to forward them to the destination [14]; afterward, it carries out a black hole or gray hole attack by dropping all or certain received packets, respectively.

[1]UDA, which stands for UAV detection agent, runs at the UAV node level.

III. Hierarchical Intrusion Detection and Response Scheme for UAV Networks

We propose and conceive in this paper an efficient and lightweight detection and response scheme to protect UAV networks. This system is efficient since it detects the attacks promptly and it is lightweight because it requires a low overhead to achieve a high level of security (i.e., high detection and low false positive rates). The hierarchical scheme aims to prevent the occurrence of the most lethal cyber-attacks that could target an UAV network, such as GPS spoofing, jamming, false information dissemination, and gray hole and black hole attacks. The network model that we attempt to secure is the one described in Section II-A. Our hierarchical scheme relies on two mechanisms: 1) an intrusion detection mechanism running at the UAV node level and 2) an intrusion response mechanism running at the ground station level. In this section, we start describing the set of detection rules applied by the detection mechanism to detect the attacks cited above. Afterward, we explain the process of intrusion verification, node assessment, and UAVs' categorization into well or bad-behaved nodes performed by the response mechanism.

A. Intrusion Detection Mechanism

IDS is the most reliable technique to detect the cyber-attacks. Furthermore, it is noted that, due to sparse communications in DTN, an UAV node or a set of UAVs could not activate their IDSs and monitor all the behaviors occurred within the network. Thereby, the most appropriate solution in DTN is to deploy a distributed intrusion monitoring and detection approach [29], [30]. In this approach, each node can activate an IDS agent (UDA) by using a *promiscuous monitoring mode*, i.e., UDA can hear all packets within its radio range and can observe the behavior of UAVs that traverse its neighboring area. In addition, *mutual monitoring* is applied where each UAV can play the UDA role and monitor its neighbors, and vice versa. Furthermore, in case when the UDA is detected as an abnormal or malicious node, it loses the ability to monitor anymore as explained in Section III-B. As a result, a secure community of UDAs is achieved. This final decision-making, i.e., the monitored UAV is an attacker or a malicious decision is provided by UDA, should be taken by a centralized trust entity (e.g., ground station) to decrease false positives and negatives.

In this paper, we propose to use a rules-based intrusion detection approach to identify the most lethal attacks that target UAV networks, where we define the following detection rules against GPS spoofing, jamming, false information dissemination, and gray hole and black hole attacks.

1) GPS Spoofing Attack's Detection Rule: A set of rules is proposed to model the normal behavior of the nodes based on GPS spoofing's characteristics[2] [24], [31]: 1) the GPS spoofing attack generates a high signal strength intensity (SSI) to get control of the drone, and this SSI is higher than that from satellites as demonstrated by Shepard *et al.* [11] and Kim *et al.* [24] and 2) the spoofer transmits several signals

[2]This is unlike anti-GPS spoofing detection approach that requires additional hardware components (e.g., antenna) to detect this attack [33].

from a single antenna; consequently, they have almost the same signal strength [24]. The detection process is carried out as follows: the UDA collects SSIs that come from the transmitters (satellites and attackers), and then evaluates their SSIs' distribution using the *normal distribution concept*. In the latter, SSIs' mean and standard deviations (σ) are computed according to (3) and (4), respectively. Furthermore, the SSIs are correctly distributed (i.e., have almost the same value) if they lie within mean $\pm 3{*}\sigma$ [32]

$$\text{Mean(SSI)} = \sum_{i=1}^{n} \frac{\text{SSI}_i}{n} \tag{3}$$

$$\sigma(\text{SSI}_i) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\text{SSI}_i - \text{Mean (SSI)})^2}. \tag{4}$$

Here, n is the number of *signals* generated by transmitters. The UDA checks whether there are some SSIs that lie within (mean $-3{*}\sigma$) and (mean $+3{*}\sigma$). However, SSIs that lie within this range are identified as generated by the same transmitter. Therefore, the monitored transmitter is suspected to carry out a GPS spoofer attack. Furthermore, a reasonable maximum SSI can be set to limit the spoof signal power since according to Wen *et al.* [31], the spoofer node will increase the signal power in space by at least 3 dB. As a result, in case when the monitored transmitter exceeds the SSI's threshold TH_{ssi}, it will be detected as a GPS spoofer attacker. We note that the antenna type and environmental effects like multipath may change the received signal power [33], [34]. In addition, the signal strength is very easy to manipulate by an attacker; all it needs to do is to set the transmission amplifier gain, which will render this rules-based detection ineffective. Thereby, in order to overcome these issues, TH_{ssi} is updated over time with the help of an SVM learning algorithm (embedded at the ground station level) as explained in Section III-B. The rule for detecting GPS spoofing is illustrated in Algorithm 2. Such attack could be launched by a GPS spoofer equipment [11]. Thereby, when an attack is detected, the UDA stores and forward to the ground station an *intrusion report*, which includes the location where the malicious equipment is detected and the attacker's SSI.

2) Jamming Attack's Detection Rule: The most lethal jamming attacks are *deceptive* and *random* jammers that aim to jam communications, and then instigate GPS spoofing to alter the GPS coordinates as performed by Iranian military in December 2011. As described in Section II-B2, the characteristics of these attacks according to [28] are defined as follows: 1) when the node carries out a jamming attack, it sends a considerable amount of packets and as a consequence the number of packets sent (NPS) significantly differs from its neighbors and 2) JITTER is very low (high) when *deceptive* (*random)* jammers are instigated. According to these characteristics, a rule-based detection is defined to detect such attacks as follows. The UDA collects the packets that come from the transmitters (e.g., UAV, satellites, and attacker) located within its radio range, and then evaluates the distributions of NPS and JITTER using the *normal distribution concept*, in which the NPS (and JITTER) are said to be correctly distributed if they lie within (mean $\pm 3{*}\sigma$), [32]

$$\text{Mean(NPS)} = \sum_{j=1}^{k} \frac{\text{NPS}_j}{k}$$

$$\sigma\left(\text{NPS}_j\right) = \sqrt{\frac{1}{k}\sum_{J=1}^{k}\left(\text{NPS}_j - \text{Mean (NPS)}\right)^2} \tag{5}$$

$$\text{Mean(JITTER)} = \sum_{j=1}^{s} \frac{\text{JITTER}_j}{s}$$

$$\sigma\left(\text{JITTER}_j\right) = \sqrt{\frac{1}{s}\sum_{J=1}^{s}\left(\text{JITTER}_j - \text{Mean(JITTER)}\right)^2}. \tag{6}$$

Here, k is the number of the suspected transmitter's neighbors observed by UDA and s is the number of messages sent by a suspected transmitter. In this case, the agent checks whether NPS (and JITTER) follows a normal distribution, i.e., within (mean $\pm 3{*}\sigma$). However, when the NPS (and JITTER) of a monitored node does not lie within this range, this node is suspected to carry out a jamming attack. To increase the accuracy detection, we fix a threshold TH_{NPS}, and when the NPS of a monitored node is greater than TH_{NPS}, the node is suspected to be carrying out the jamming attack. This threshold is updated over time with the help of an SVM learning algorithm as explained in Section III-B. The rule for detecting jamming attack is illustrated in Algorithm 3.

When the UDA detects a jamming attack, it stores and forward to the ground station an *intrusion report*, which includes the identity of the suspected transmitter (e.g., UAV), the kind of detected jamming (deceptive or random jammers), and the attacker's NPS (and JITTER).

3) False Information Dissemination Attack's Detection Rule: As described in Section II-B, our aim is to protect the UAVs network against three types of false information dissemination attacks, where the detection rules related to each one of them are illustrated in Algorithm 4 and summarized as follows.

a) False data injection's detection: The UAV task is to monitor, sense, and broadcast the observed physical phenomena, e.g., forest fires, injured persons, and traffic accidents. Furthermore, UAVs that are located within the same neighborhood should report the same phenomena [12]. However, the malicious UAV could compromise the sensors' readings and inject a wrong physical phenomenon. The UDA relies on the *promiscuous mode* (which overhears all the packets that pass within its radio range) to detect such attack, where the observed phenomena are compared with those broadcasted by UAV neighbors. Furthermore, when the monitored UAV delivers false information, it will be suspected as an attacker. Note that when there are not enough UAVs within the same neighborhood, for instance, only one UAV and one UDA, it is hard to decide whether if the suspected UAV, detected by the UDA, is really malicious or not. Therefore, to address this issue and decrease the false positives, the ground station performs a node assessment process (as explained in Section III-B) by collecting the historical data of UDA detection activities. Afterward,

Algorithm 2 Detection Rules of GPS Spoofing Attack

1: UDA monitors the distribution of SSIs that come from the neighboring transmitters
2: ***if*** ($SSI_i \in [(mean_{ssi} - 3*\sigma_{ssi}), (mean_{ssi} + 3*\sigma_{ssi})]$ && ($\text{transmitter}_\text{i}$ claims that it generates signals from different locations))
3: //the $\text{transmitter}_\text{i}$ is suspected to carry out a *GPS spoofing attack*
4: ***if*** ($SSI_i > TH_{ssi}$)
5: //the $\text{transmitter}_\text{i}$ is a *GPS spoofer*
6: Forward an *Intrusion Report (malicious equipment's location, attacker's SSI, GPS spoofer)*
7: Updates TH_{ssi} based on SVM
8: ***end if***
9: ***end if***

Algorithm 3 Detection Rules of Jamming Attack

1: UDA monitors the distribution of NPS and JITTER that come from the neighboring UAVs
2: ***if*** ($NPS_i \in [(mean_{nps} - 3*\sigma_{nps}), (mean_{nps} + 3*\sigma_{nps})]$ && $JITTER_i \in [(mean_{jitter} - 3*\sigma_{nps}), (mean_{jitter} + 3*\sigma_{nps})]$)
3: //the $\text{transmitter}_\text{i}$ is suspected to carry out a *jamming attack*
4: ***if*** ($NPS_i > \text{TH}_\text{NPS}$)
5: //the $\text{transmitter}_\text{i}$ is a *jammer*
6: Forward an *Intrusion Report (identity of a malicious UAV, attacker's NPS (and JITTER), jamming)*
7: Updates TH_NPS based on SVM
8: ***end if***
9: ***end if***

the ground station makes the final decision, i.e., the suspected UAV is classified as an attacker or not.

B. ADS-B Attack's Detection

As described in Section II-B, such attack aims to either broadcast a false position or spoof GPS coordinates in order to, for instance, incur an UAV's crash. Several detection policies have been proposed to check the position claimed by a mobile node [35], [36]. To perform this task, we use the detection strategy proposed by Sedjelmaci *et al.* [35] to determine the position of a target. In this strategy, two algorithms, relying on SSI and packet's round trip time (RTT) are used to determine the location of a target node. We refer the readers to [35] for more details about these algorithms. Furthermore, for the GPS spoofing coordinates attack, the detection rules explained above (see GPS spoofing attack's detection) are used.

C. Bad- and Good-Mouthing's Detection

In this attack, the UDA brings a false detection, i.e., claiming a normal UAV as an attacker or vice versa, thus leading to an increase on the false positive and negative rates. Mármol and Perez [37] proposed a node assessment approach to mitigate such attack. The assessment approach (see Section III-B), which is embedded at a ground station, evaluates the behavior of a monitored UAV and the decision provided by the UDA. Furthermore, the UDA that provides a wrong detection will see its trust level (TL) decreasing, and will eventually be considered malicious as it continues to provide false detection.

When the UDA detects an attack, it stores and forward to the ground station an *intrusion report*, which includes the identity of the suspected UAV and the kind of the detected attack (e.g., ADS-B or false data injection attacks).

1) Gray Hole and Black Hole Attacks' Detection Rule: As described in Section II-B, when these attacks occur, they drop all or a certain number of received packets. Furthermore, it is difficult for the UDA to monitor and detect such attacks in this network owing to UAV mobility and frequent disconnections. As mentioned in Section II, the ground stations are assumed to be trusted nodes and static, thus allowing them to monitor the packets that circulate within the network [14]. The use of ground stations for intrusion monitoring could generate an overhead. In fact, there is a tradeoff between the security requirement and overhead. Thereby, centralized intrusion monitoring is launched only for black hole and gray hole attacks. The detection policy of these attacks is explained below.

Algorithm 4 Detection Rules of False Information Dissemination Attack

1: UDA compares the observed phenomena with those broadcasted by UAVs' neighbors
2: ***if*** (UAV u_i delivers false information)
3: //u_i is carrying out a *false data injection attack*
4: Forward an *Intrusion Report (identity of malicious u_i, false data injection)*
5: ***end if***
6: UDA monitors the distribution of SSIs and RTTs
7: ***if*** ($SSI_i \in [(mean_{ssi} - 3*\sigma_{ssi}), (mean_{ssi} + 3*\sigma_{ssi})]$)
8: //the UAV u_i is suspected to broadcast a false position
9: ***if*** ($RTT_i > TH_{RTT}$)
10: //u_i is carrying out an *ADS-B attack*
11: Forward an *Intrusion Report (identity of malicious u_i, ADS-B attack)*
12: ***end if***
13: ***end if***
14: Ground station evaluates the intrusion decision provided by UDAs
15: Compute the trust level (*TL*)
16: ***if*** (*TL* of UDA $u_i < TH_{TL}$)
17: //u_i is carrying out a *bad- or good-mouthing attack*
18: ***end if***

Each UAV, in the neighborhood of a ground station, sends a *neighboring packet* to the ground station. This packet contains the node's type (source or relay), next hop node (i.e., the neighboring node to which the UAV forward the packet), and previous hop node (i.e., the neighboring node from which the UAV receives the packet). To minimize the false positive, the ground station collects *neighboring packets* from normal and suspected UAVs, and ignores the ones received from abnormal and malicious UAVs (refer to Section III-B on categories of UAVs). Afterward, the ground station checks whether the relay node forwards a packet and computes the number of packets dropped (NPD). It is important to recall that the NPD could also be caused by the collisions and signal attenuation due to obstacles. Note that these packet-dropping events are less than those caused by the black hole and gray hole attacks. Therefore, in order to distinguish between them, we set new thresholds TH_BH and TH_GH for black hole and gray hole attacks, respectively. These thresholds will be updated by using the SVM algorithm as will be explained in Section III-B. The detection rules of gray hole and black hole attacks are illustrated in Algorithm 5.

Algorithm 5 Detection Rules of Gray Hole and Black Hole Attacks

1: Ground station collects *neighboring packets* from normal and suspected UAVs
2: Monitors *NPD* of UAV u_i
3: ***if*** ($NPD_i > TH_{GH}$)
4: //u_i is carrying out a *gray hole attack*
5: ***end if***
6: ***else if*** ($NPD_i > TH_{BH}$)
7: // u_i is carrying out a *black hole attack*
8: Updates TH_{GH} and TH_{BH} with a help of SVM
9: ***end if***

D. Intrusion Response Mechanism

The response mechanism is embedded in the ground station to evaluate the UAV's behavior and categorize each UAV according to its perceived threat into the appropriate list. UAV categorization into a well or bad behaved node and permanent exclusion of a malicious UAV can only be done by a trust entity (i.e., ground station) to decrease the false positives and negatives. Thereby, as mentioned in Section III-A, each UDA broadcasts to the ground station located within its radio range an *Intrusion Report* message, which includes the suspected UAV's information. The ground station stores into its database the id of this suspected UAV, the *ids* of UDAs that detect the suspected UAV as an intruder and also the *ids* of UDAs (neighbor of suspected UAV) that do not detect it as an intruder. Afterward, the ground station executes the following three processes.

1) Verification Process: The ground station uses an anomaly detection technique to check the attack detected by the UDAs and make the final decision whether the suspected UAV is an attacker. The anomaly detection technique can model the anomaly behavior of a target node with high accuracy [5], [38]. The response mechanism uses an SVM learning algorithm to carry out data training and classify the incoming data as normal or anomaly. In the *training phase*, the ground station collects the *Intrusion Report* delivered by the UDAs, which contains the features of the suspected attack and its identity. For instance, the feature related to a GPS spoofing is the SSI. These features are used as input vectors for the SVM training. The latter computes a set of vectors called *support vectors* to obtain a separating hyperplane, and hence procures binary classes (normal and anomaly). Readers are referred to [39] for more details about a hyperplane separation. To obtain an accurate binary categorization, the training process is carried out periodically since the *support vectors* will vary over time due to noise and unreliable wireless communications [40]. In the discrimination phase, the ground station classifies the new features according to the anomaly and the normal patterns, determined during the training phase. Once the ground station confirms the attack detected by the UDA, it informs the UDAs within its radio range to update the threshold related to the detected attack with the value of the current feature used by the SVM training algorithm. Otherwise, the ground station ignores the *Intrusion Report* delivered by this agent.

2) Node Assessment: Based on the *Intrusion Report* delivered by the UDA, the ground station evaluates the UAVs' behavior and assigns to each monitored UAV (and UDA) u_i, a TL computed as follows:

$$\mathrm{GTL}_i = \alpha_1 \mathrm{NR}_i + \beta_1 \mathrm{TDR}_i, \mathrm{BTL}_i = \alpha_2 \mathrm{DR}_i + \beta_2 \mathrm{FDR}_i$$
$$\mathrm{TL}_i = (\mathrm{GTL}_i - \mathrm{BTL}_i) \tag{7}$$

where GTL is the good TL, BTL is the bad TL, FDR represents the number of UDAs that do not agree on the detection provided by the UDA u_i, TDR denotes the number of UDAs that agree on the detection claimed by the UDA u_i, DR is the number of UDAs that detect the UAV u_i as an attacker, NR is the number of UDAs that detect the UAV u_i as a normal node, and finally α_1, β_1, α_2, $\beta_2 \in [0, 1]$ are weighting factors. The ground station stores the TL of each UAV that passes within its range in the Trust_database as follows: (UAV u_i, TL_i); afterward, this information is exchanged among all ground stations within the network. At the end, each ground station computes a Total TL as shown in (8). As explained in Section II-A, the ground stations are connected among them through a secure digital professional mobile radio standard like TETRAPOL [16]

$$\mathrm{Total\_TL}_i = \frac{\sum_{k=1}^{d} \mathrm{TL}_{ik}}{d} \tag{8}$$

where d is the number of ground stations that compute the TL of UAV u_i. Note that when $\mathrm{Total\_TL}_i < 0$, the monitored UAV is considered as a malicious node.

The frequent computation of the node's TL is not desirable since it is affected by noise and packets collisions [12]. Thereby, to get an accurate TL value, we adopt the beta distribution [41]. We model Total_TL by a variable x with $P(\cdot) = \mathrm{Beta}(\alpha, \beta) \in [0, 1]$ [41], which is given by

$$P(x) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}. \tag{9}$$

$P(x)$ is characterized by two parameters (α, β), where $0 \leq x \leq 1$, $\alpha > 0$ and $\beta > 0$. As in [12], in order to determine an accurate Total_TL value, we set α and β values to 1 and $[1/\log[1/(1 - \mathrm{Total\_}TL)])]$, respectively. The output of $P(x)$ varies between 0 and 1, with the value 0 indicating that the UAV u_i persists to misbehave, i.e., does not switch to a well-behaved node and 1 indicating that the UAV u_i is a normal node.

3) Monitored UAVs' Categorization Process: It is not wise to eject the UAV directly when it exhibits a malicious anomaly since it could switch to a normal behavior and acts as an ordinary node during its passage through the network [41]. Therefore, as illustrated in Table I, the ground station categorizes the monitored UAV into four categories according to its observed behavior.

UAVs that are categorized as abnormal nodes are limited from their participation in the network. For instance, they do not have the right to activate their UDA agents. Furthermore, when they persist to monitor their neighbors and forward an Intrusion Report to the ground station, their $P(x)$ will be decreased. Moreover, the UAVs that are categorized as malicious (attacker), their id_s will be broadcasted in order to prevent legitimate UAVs from communicating with them.

TABLE I
UAV'S CATEGORIZATION

Category	Description
Normal	The UAV u_i is a normal node. Here, P(x)= 1
Suspect	The UAV u_i oscillates between a well and badly-behaved node. However, the switching rate to a badly behaved node is less than that to a well-behaved node. Here, P(x)∈ [0.7, 0.9]
Abnormal	The UAV u_i oscillates between a well and badly behaved node. However, the switching rate to a badly-behaved node is higher than that to a well-behaved node. Here, P(x)∈ [0.3, 0.6]
Malicious (Attacker)	The UAV u_i behaves persistently bad, i.e., does not switch to a well-behaved node. Here, P(x)∈ [0, 0.2]

IV. SIMULATION RESULTS

In this section, we evaluate the performance of our hierarchical detection and response scheme using NS3 simulator [43]. We compare it with a distributed scheme where the rules-based detection, anomaly detection, node assessment, and UAVs' categorization are carried out at the UAV level (i.e., ground stations are not involved in the detection and response). In addition, we compare our hierarchical scheme with the current intrusion detection scheme proposed for UAVs, namely, BRUIDS [12]. Here, we compute the detection rate, false positive rate, efficiency, and communications overhead, as defined below.

1) *Detection rate* is the ratio of correctly identified attackers over the total number of attackers.
2) *False positive rate* is the number of normal UAVs that are incorrectly classified as attackers over the total number of normal UAVs.
3) *Efficiency* is the time required for UDAs to detect attackers, computed as follows:

$$\text{Efficiency} = \sum_{i=1}^{N} \frac{\text{DT}_i - \text{MT}_i}{\text{Sampling frequency} * N} \qquad (10)$$

where MT_i is the moment when the attack starts, DT_i is the detection time of the cyber-attack, and N is the number of cyber-attacks [40].
4) *Communications overhead* is the amount of bytes generated by our hierarchical detection and response scheme to achieve high detection and low false positive rates.

A. Mobility Model and Simulation Setup

The mobility model used for simulations has an impact on the simulation results. To simulate a realistic UAV network, a *deterministic mobility model* [13] is used. In this paper, we use a *paparazzi deterministic mobility model* [44] where the UAV follows a well-defined trajectory. The map explored by the UAVs is mapped onto a grid with homogeneous squares as illustrated in Fig. 1. Each UAV explores a monitored zone and collects critical disaster information, and then transmits them to a suitable next hop node, i.e., UAV or ground station, by using a greedy forwarding mechanism [18]–[20]. As mentioned in Section II-A, the farthest next-hop is selected to relay data. As explained in Section III-B, the threshold related to each attack's detection is updated over time via SVM learning.

TABLE II
SIMULATION PARAMETERS

Simulation area	$5000 \times 5000\text{m}^2$
Simulation time	1800 seconds
Mobility model	Paparazzi deterministic
UAV number	From 50 to 250
UAV speed	60 to 150 km/h, step size15
Link layer	802.11b-11 Mbps
Transport layer	UDP
UAV's transmission range	27 meters
Routing algorithm	DTN routing
TH_{ssi}	(at t=0) -32 dBm
TH_{NPS}	(at t=0) more than 15% of packets are sent as compared to normal UAV nodes
TH_{RTT}	(at t=0) 180 millisecond
TH_{GH}	(at t=0) 34% of packets are dropped
TH_{BH}	(at t=0) 84% of packets are dropped

This update is mandatory due to the following reasons: 1) environmental effects like multipath or noise could affect the monitored features (e.g., SSI, NPS, JITTER, and NPD) and 2) the attacker could have knowledge of the threshold and set an appropriate one, which will render the rules-based detection ineffective. At the beginning of the simulation (i.e., $t = 0$), each threshold has been set with an initial value. The main simulation parameters are summarized in Table II.

B. Analysis of the Results

We implement, using NS3, our hierarchical mechanism, BRUIDS [12] and the distributed scheme. We inject the same number of attacks, namely, *false data injection, ADS-B, bad- and good-mouthing, GPS spoofing, jamming*, and *gray hole and black hole attacks*. Note that BRUIDS does not have the capability to detect *GPS spoofing*, *gray hole, and black hole*. Thereby, we add to this scheme our detection rule (see Section III-A) to enable it to identify these attacks. Afterward, we compare the performances of these schemes in terms of the metrics cited above. Here, the number of attackers vary from 10% to 30% of the overall nodes. The most important results are summarized below.

1) Detection Rate: As illustrated in Fig. 2, when the number of UAVs increases, the detection rates of our hierarchical, BRUIDS and distributed schemes decrease, specifically when the number of attackers is high. This reduction is much greater for BRUIDS as compared to other mechanisms. The hierarchical scheme can detect all the attacks cited above with a detection rate that is above 93%. This result is achieved when the number of UAVs and attackers is equal to 250% and 30% of overall nodes, respectively (i.e., worst case). This result is achieved due to the following reasons.

1) *Intrusion Detection Techniques:* Our scheme combines the advantages of rules-based and anomaly detection techniques to achieve high detection accuracy, unlike the BRUIDS framework, which relies only on rules-based detection to identify the attackers.
2) *No Drone Is 100% Trustable:* The attackers are more interested to target attractive nodes that manage relevant information, for instance, intrusion detection agents that

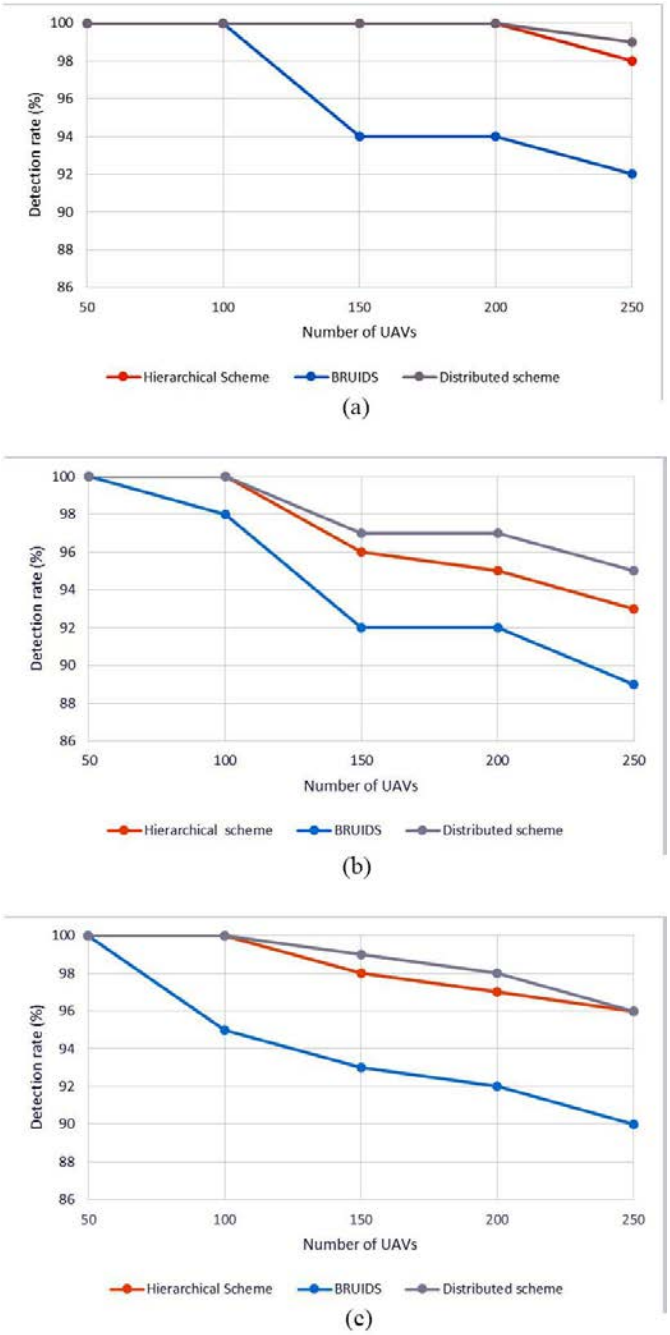


Fig. 2. Detection rate: number of attackers equal to (a) 10% of overall nodes, (b) 20% of overall nodes, and (c) 30% of overall nodes.

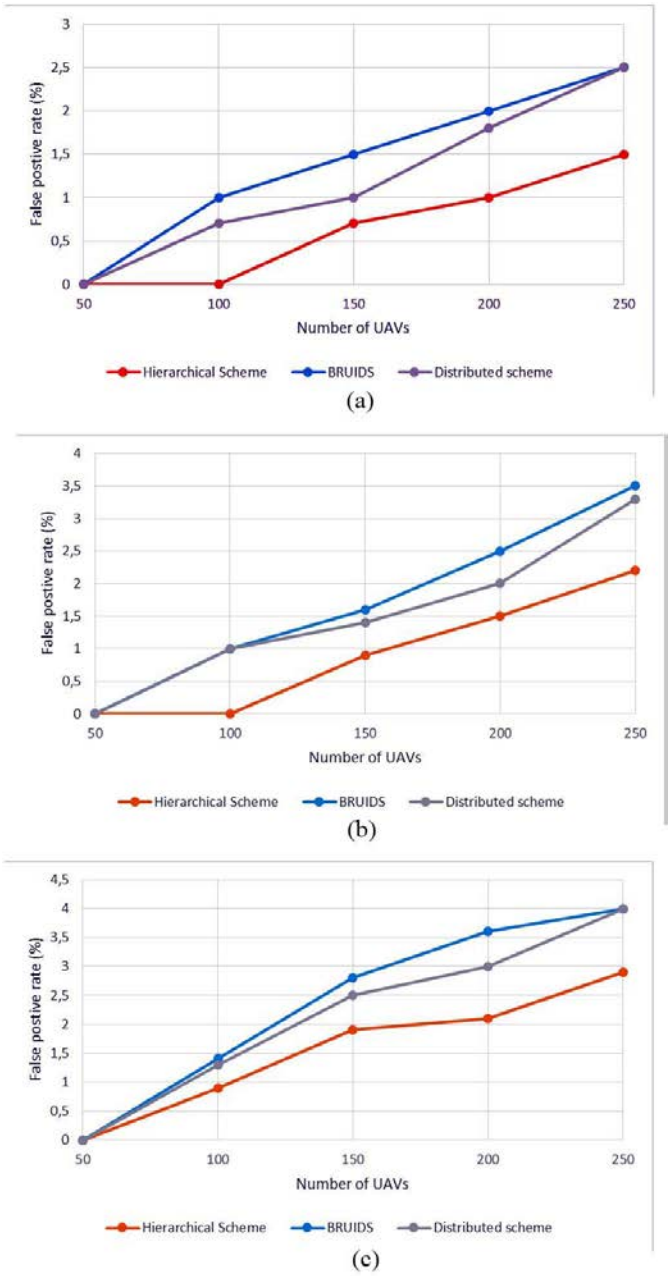
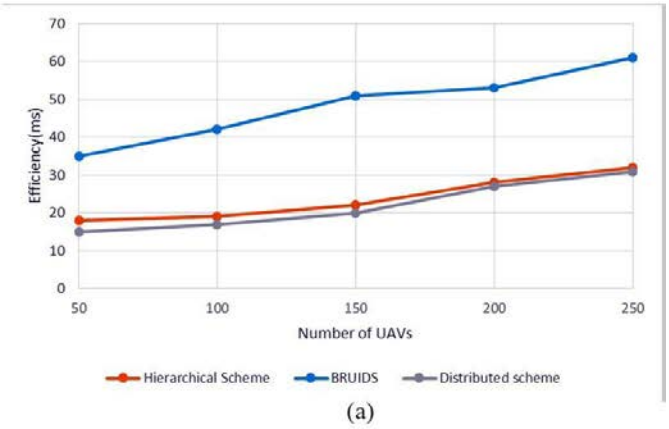


Fig. 3. False positive rate: number of attackers equal to (a) 10% of overall nodes, (b) 20% of overall nodes, and (c) 30% of overall nodes.

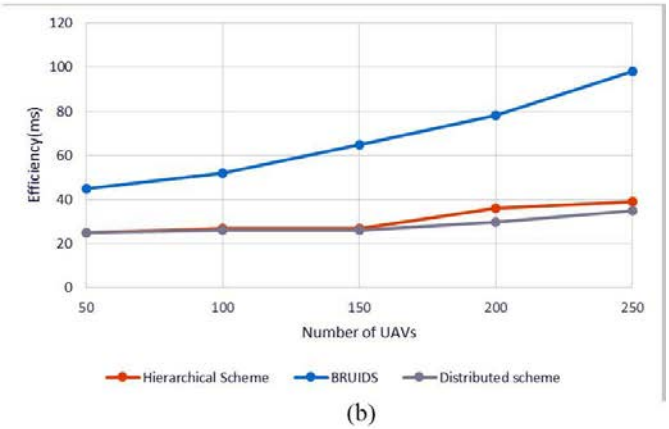
have the authority to categorize a monitored node as normal or attacker. Thereby, our scheme takes into account of this fact and also analyzes the decision provided by UDAs (with the help from ground stations), unlike BRUIDS, which assumes that an intrusion detection agent is a trusted node.

2) False Positive Rate: As shown in Fig. 3, the false positive rate generated by the hierarchical scheme increases slowly as compared to other schemes when the number of both UAVs and attackers increase. The false positive rate yielded by the proposed scheme is less than 3% when the number of UAVs and attackers are equal to 250% and 30% of overall nodes, respectively. This result is attributed to the following reasons.

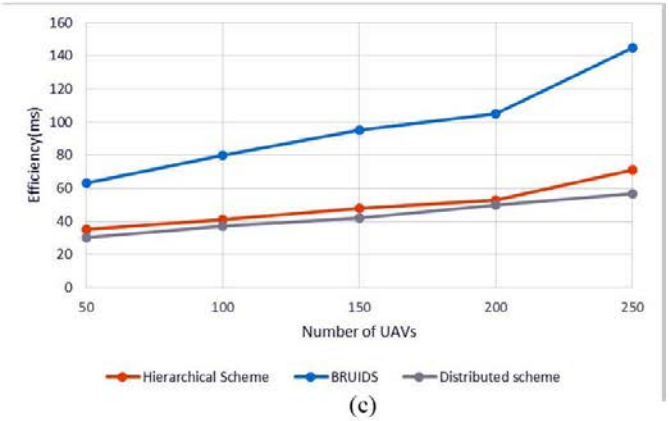
1) *Node Assessment:* The ground station categorizes the monitored UAVs into *normal*, *suspect, abnormal,* and *malicious (attacker)* by computing their TL $P(x)$. This classification is done via the node assessment process. Therefore, this process helps decrease the false positive rate.
2) *Cooperative Detection and Decision:* To distinguish between a normal UAV and a malicious one, and hence decrease the false positive, the intrusion detection agents that are running at UAV and ground station levels cooperate between each other to detect an attacker and make



(a)



(b)



(c)

Fig. 4. Efficiency: the number of attackers equal to (a) 10% of overall nodes, (b) 20% of overall nodes, and (c) 30% of overall nodes.

the final decision (i.e., the monitored UAV is normal or malicious).

3) *Intrusion Detection Techniques:* As mentioned above, the combination of rules-based and anomaly detection techniques enables detection of the malicious UAV with a high accuracy.

3) Efficiency: Fig. 4 illustrates the required time for intrusion detection agents (UDAs for the hierarchical scheme) to detect the attacks cited above. It is observed that the efficiency of the hierarchical and distributed schemes is almost the same, specifically when the number of UAVs is large. This is unlike BRUIDS, which incurs a high intrusion detection delay,

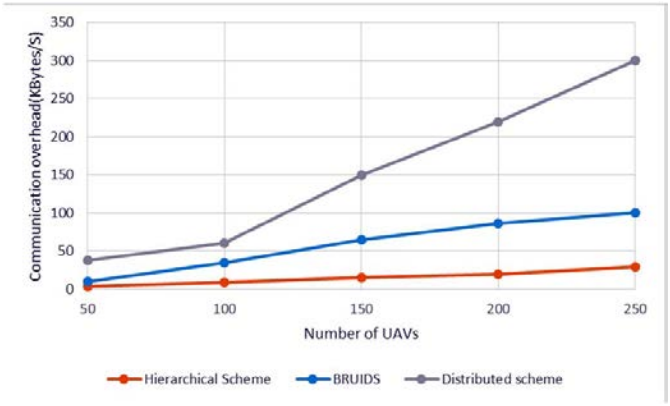


Fig. 5. Communications overhead: number of attackers equal to 30% of overall nodes.

due to the fact that the IDS agents should wade all the rules to detect a malicious anomaly. According to the simulation results, the hierarchical scheme detects the attacks promptly in the order of milliseconds. This result is achieved even when the numbers of UAVs and attackers equal to 250% and 30% of overall nodes, respectively. Furthermore, when the numbers of UAVs and attackers are high, the efficiency of the hierarchical intrusion detection and response scheme is about 70 ms as illustrated in Fig. 4(c), which satisfies the requirement of delay-sensitive applications. Such efficiency is attributed to *community of trusted UDAs:* all the UAVs, in the hierarchical scheme, have the ability to run an intrusion detection agent (i.e., UDA) and monitor their neighbors. However, when the UAV is suspected to be malicious, it cannot play the role of UDA. Therefore, only a community of trusted UDAs carry out the intrusion monitoring and detection, thus reducing the time of detecting malicious UAVs.

4) Communications Overhead: We evaluate the communications overhead generated by the hierarchical scheme, BRUIDS and distributed scheme by fixing the number of attackers to 30% of overall nodes. As illustrated in Fig. 5, the hierarchical scheme requires a low communications overhead to detect the cyber-attacks as compared to the other schemes. This is achieved even when the number of UAVs increases.

5) Theoretical Analysis: In this section, we analyze the security of the proposed hierarchical detection and response schema by using the communications overhead, false positive rate and detection rate as main metrics.

Theorem 1: $V\ (t) >> V'(t)$, where $V\ (t)$ and $V'(t)$ are the communications overhead generated by BRUIDS [12] (or distributed) and hierarchical schemas, respectively.

Proof: $F(t) = \sum_{i=1}^{|C|} \sum_{j=1}^{|D_i|} f(c_i, d_j, t)$ is the number of messages sent and received by the IDSs agents in BRUIDS (or distributed) schema, and $F'(t) = \sum_{i=1}^{|C|} \sum_{j=1}^{|D'_i|} f(c_i, d'_j, t)$ is the number of messages sent and received by UDA agents in the hierarchical schema, where $C = \{c_1, \ldots, c_n\}$ is the set of all UAVs in a network, $D = \{d_{i1}, \ldots, d_{im}\}$ is the set of neighbors of the suspected UAV u_i, and $D' = \{d'_{i1}, \ldots, d'_{im}\}$ is the set of trusted neighboring UDAs of the suspected UAV u_i

$$f(c_i, d_j, t) = \begin{cases} 1 \text{ IDS agent sent or received message} \\ 0 \text{ Otherwise} \end{cases}$$

and

$$f(c_i, d'_j, t) = \begin{cases} 1 \text{ UDA agent sent or received message} \\ 0 \text{ Otherwise.} \end{cases}$$

The condition $F(t) >> F'(t)$ is held for the following reasons: 1) in our hierarchical mechanism, only a trusted number of UDA nodes are activated to analyze the behavior of their neighbors, unlike BRUIDS and distributed schemes, in which all UAVs activate simultaneously their IDSs and 2) our proposed mechanism minimizes the information exchanged between UAVs and the ground station. BRUIDS and distributed schemes incur a huge amount of monitoring messages to be exchanged between UAVs when a suspected node is detected. Due to the fact that the communications overhead depends on the number of exchanged messages and size of message (m) as shown in (11) and (12) [45], hence $V(t) >> V'(t)$, which is also demonstrated in simulation analysis, as shown in Fig. 5

$$V(t) = F(t)m(t) \tag{11}$$

$$V'(t) = F'(t)m'(t) \tag{12}$$

where m and m' are the average sizes of messages that UAVs sent and received in BRUIDS (or distributed) and hierarchical schema, respectively. ■

Theorem 2: The false positive rate of our hierarchical scheme depends on the collision rate and number of UDA agents.

Proof: According to Hai *et al.* [46], in the network where the topology changes frequently due to fading and node failures (such as the case of UAV-DTN), the probability of detection P_D of IDS agent u_j against an attacker u_i depends mainly on the probability of collision P_C occurring in a monitored transmission link as shown in

$$P_D(t) = P_C(t)(1 - P_C(t))^2 \tag{13}$$

$$\begin{aligned} \Rightarrow P_{F_j}(t) &= (1 - P_C(t))^2 P_C(t) + {P_C}^2(t)(1 - P_C(t)) \\ &= P_C(t)(1 - P_C(t)) \end{aligned} \tag{14}$$

where $P_{F_j}(t)$ is the probability of false positive generated by UDA agent u_j.

So, the probability of false positive generated by our hierarchical scheme when the attacks occur in the network is

$$P_F(t) = (1 - P_C(t))^S P_C(t)^{K-S} + \cdots + (1 - P_C(t))^K P_C(t)^{K-S}. \tag{15}$$

Here, K is the number of UDA agents and S is the number of UDAs that detect the cyber-attacks.

When K UDA agents collaborate to decide the status of a monitored UAV, i.e., the monitored UAV is normal or attacker, the probability of false positive can be defined as follows:

$$P_F(t) = \sum_{i=s}^{k} (1 - P_C(t))^i P_C(t)^{K-S}. \tag{16}$$

As a result, we can claim that to decrease the false positive rate, the number of collisions and UDA agents should be decreased and increased, respectively. ■

Theorem 3: The attack detection rate of the hierarchical scheme depends on the radio range of UAV (R) and network density (D).

Proof: Let y be the distance between UAVs u_1 and u_2, and the radio range (R) of UAVs are assumed to be the same. For any distance y, the area, where IDSs (in our case UDA agents) are located, is calculated as follows [47]:

$$S(y) = 2R^2 \cos^{-1}\left(\frac{y}{2R}\right) - 2y\sqrt{R^2 - \frac{y^2}{4}}. \tag{17}$$

According to Hai *et al.* [46], the number of monitor nodes (i.e., UDAs) for each link $u_i - u_j$ where $i \neq j$ is given by

$$\begin{cases} \lfloor 0.362R^2D \rfloor, & \text{if } y = R \\ \lfloor E[S(y)]D \rfloor, & \text{if } y \neq R. \end{cases} \tag{18}$$

$E[S(y)]$ is the expected value of the area $S(y)$ and is computed as follows:

$$E[S(y)] = \int_0^R S(y).f(y)dy.$$

The probability distribution of y is computed by: $F(y) = (y^2/R^2)$, and the probability density function $f(y) = (dF(y)/y) = (2y/R^2)$.

As a result

$$\begin{aligned} E[S(y)] &= \int_0^R \frac{2y}{R^2}\left(2r^2\cos^{-1}\left(\frac{y}{2R}\right) - 2y\sqrt{R^2 - \frac{y^2}{4}}\right)dy \\ &\approx 0.30\pi R^2. \end{aligned} \tag{19}$$

Therefore, we claim that to increase the attack detection rate, both UAV's radio range and network density should be increased. ■

V. Related Works

Intrusion detection is essential to protect the network against attackers since it can potentially detect the cyber-attacks with a high detection and low false positive rates [5], [12], [48]–[51], while authentication only aims to prevent the external attacker from entering the network. UAVs are attractive and easy targets for the attackers due to the relevant information handled by UAVs. However, the protection of such network has not been well investigated to the best of our knowledge; the intrusion detection scheme called BRUIDS proposed by Mitchell and Chen [12] is the only publicly available work that relies on detection techniques to protect the UAV network against cyber-attacks. BRUIDS aims to detect the attacks that target the integrity such as false data injection and attacks that target the availability such as malicious UAV that directs its weapon against a friendly resource, jamming, bad- and good-mouthing. In this detection framework, the authors proposed a set of rules related to the attacks to model a normal UAV behavior. According to their simulation results, their detection system exhibits a low false negative. However, the false positive is higher (equal to 7%). In addition, this framework incurs a high overhead when the number of UAVs is large, and is thus not scalable. Kim *et al.* [24] evaluated the behavior of attackers that target UAV nodes such as

intruders that access to the autopilot components and intruders that inject false data into UAV on-board sensors. According to these malicious anomalies, they proposed certain detection rules to model a normal UAV behavior. They did not evaluate the performances of their approach in terms of detection accuracy and overhead.

Strohmeier *et al.* [9] summarized the attacks that could target ADS-B component such as eavesdropping, jamming, and data injection, and proposed a set of countermeasures to identify them. However, they did not conduct any simulation or experimental analysis to evaluate the performances of their proposed security countermeasures. Strohmeier *et al.* [9] and Wesson *et al.* [10] claimed that ADS-B is a vulnerable component to a wide range of attacks since it has no built-in security mechanism. They proposed a confidentiality-based solution to ensure the privacy of messages broadcasted by the ADS-B component. However, detecting the attacks that can target this component is not addressed. According to Shepard *et al.* [11], GPS spoofing is the most lethal cyber-attack that could target UAVs as explained in Section II-B, in which the Iranian hacker captured a military UAV by using such attack [11]. Several research works have proposed detection policies to identify the GPS spoofing attack. Sedjelmaci *et al.* [52] aimed to protect the UAV-aided vehicular network against the cyber-attacks. In their work, they focus only to address the issue of IDS activation in UAV-assisted network. They did not provide detection techniques to detect the cyber-attacks that target such network. Zhang *et al.* [33] aimed to protect the power grid system against this attack by proposing the quickest spoofing detection algorithm, which is based on evaluating the distribution of a GPS signal issued from the transmitter nodes. Furthermore, they proved that the spoofer transmits the same signal from a single antenna several times to deceive the legitimate nodes that it is located in different places. According to their simulations and experimental results, the GPS spoofing can be detected with a high accuracy. However, they required an additional hardware (e.g., antenna) to identify this cyber-attack. Sedjelmaci *et al.* [53] addressed the false positive and false negative issues that are generated by the IDS agents to secure the UAVs network. In this paper, a threat estimation model based on belief concept was developed. They did not provide details on how to detect the attacks in UAVs network.

VI. Conclusion

In this paper, we have taken the challenge of securing an UAV network by proposing a hierarchical intrusion detection and response scheme, which orchestrates the intrusion detection, decision, and categorization mechanisms cooperatively between UAVs and ground stations to detect and eliminate security threats that may disrupt the network. To model a normal UAV behavior, a set of detection rules related to each cyber-attack is proposed. Furthermore, at the ground station level, SVM-based anomaly detection is used to verify the attack detected by UAV agents; node assessment and UAV's categorization (*normal, abnormal, suspect, and malicious*) are developed. We have analyzed the performance of our scheme using NS-3, and showed that it exhibits a high-level of security with a high detection rate (more than 93%) and low false positive rate (less than 3%), and facilitates prompt detection with a low communications overhead, as compared to current state of the art. Our future direction is to embed our scheme in a fleet of a dozen of Parrot drones [54].

References

[1] E. Yanmaz, R. Kuschnig, and C. Bettstetter, "Channel measurements over 802.11a-based UAV-to-ground links," in *Proc. IEEE Globecom Wi-UAV Workshop*, Houston, TX, USA, 2011, pp. 1280–1284.

[2] M. O. Cherif, S.-M. Senouci, and B. Ducourthial, "Efficient data dissemination in cooperative vehicular networks," *Wireless Commun. Mobile Comput.*, vol. 13, no. 12, pp. 1150–1160, 2013.

[3] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J.-P. Hubaux, "Eviction of misbehaving and faulty nodes in vehicular networks," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 8, pp. 1557–1568, Oct. 2007.

[4] S. Ruj, M. A. Cavenaghi, Z. Huang, A. Nayak, and I. Stojmenovic, "On data-centric misbehavior detection in VANETs," in *Proc. IEEE Veh. Technol. Conf. (VTC Fall)*, San Francisco, CA, USA, 2011, pp. 1–5.

[5] H. Sedjelmaci, S. M. Senouci, and M. Feham, "An efficient intrusion detection framework in cluster-based wireless sensor networks," *Security Commun. Netw.*, vol. 6, no. 10, pp. 1211–1224, 2013.

[6] X. Haijun, P. Fang, W. Ling, and L. Hongwei, "Ad hoc-based feature selection and support vector machine classifier for intrusion detection," in *Proc. IEEE Int. Conf. Grey Syst. Intell. Services*, Nanjing, China, 2007, pp. 1117–1121.

[7] C. Callegari, S. Vaton, and M. Pagano, "A new statistical method for detecting network anomalies in TCP traffic," *Eur. Trans. Telecommun.*, vol. 21, no. 7, pp. 575–588, 2010.

[8] A. Mitrokotsa and A. Karygiannis, "Intrusion detection techniques in sensor networks," in *Wireless Sensor Network Security* (Cryptology and Information Security Series). Amsterdam, The Netherlands: IOS Press, 2008, pp. 251–272.

[9] M. Strohmeier, V. Lenders, and I. Martinovic, "On the security of the automatic dependent surveillance-broadcast protocol," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 2, pp. 1066–1087, 2nd Quart., 2014.

[10] K. D. Wesson, T. E. Humphreys, and B. L. Evans. *Can Cryptography Secure Next Generation Air Traffic Surveillance?* [Online]. Available https://radionavlab.ae.utexas.edu/images/stories/files/papers/adsb_for_submission.pdf

[11] D. P. Shepard, J. A. Bhatti, T. E. Humphreys, and A. A. Fansler, "Evaluation of smart grid and civilian UAV vulnerability to GPS spoofing attacks," in *Proc. ION GNSS Meeting*, Nashville, TN, USA, 2012, pp. 1–15.

[12] R. Mitchell and I.-R. Chen, "Adaptive intrusion detection of malicious unmanned air vehicles using behavior rule specifications," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 5, pp. 593–604, May 2014.

[13] E. Yanmaz, R. Kuschnig, M. Quaritsch, C. Bettstetter, and B. Rinner, "On path planning strategies for networked unmanned aerial vehicles," in *Proc. IEEE INFOCOM M2MCN Workshop*, Shanghai, China, Apr. 2011, pp. 212–216.

[14] R. Lu, X. Lin, and X. S. Shen, "SPRING: A social-based privacy-preserving packet forwarding protocol for vehicular delay tolerant networks," in *Proc. IEEE INFOCOM*, San Diego, CA, USA, 2010, pp. 1–9.

[15] Y. Ren, M. C. Chuah, J. Yang, and Y. Chen, "Detecting wormhole attacks in delay-tolerant networks," *IEEE Wireless Commun. Mag.*, vol. 17, no. 5, pp. 36–42, Oct. 2010.

[16] D. Kuypers and M. Schinnenburg, "Traffic performance evaluation of data links in TETRA and TETRAPOL," in *Proc. 11th Eur. Wireless Conf. Next Gener. Wireless Mobile Commun. Services*, Nicosia, Cyprus, 2005, pp. 1–7.

[17] Z. Han, A. L. Swindlehurst, and K. J. R. Liu, "Optimization of MANET connectivity via smart deployment/movement of unmanned air vehicles," *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, pp. 3533–3546, Sep. 2009.

[18] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. ACM/IEEE 6th Annu. Int. Conf. Mobile Comput. Netw.*, Boston, MA, USA, 2000, pp. 243–254.

[19] H. Füssler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, "Contention-based forwarding for mobile ad hoc networks," *Ad Hoc Netw.*, vol. 1, no. 4, pp. 351–369, 2003.

[20] M. Rondinone and J. Gozalvez, "Contention-based forwarding with multi-hop connectivity awareness in vehicular ad-hoc networks," *Comput. Netw.*, vol. 57, no. 8, pp. 1821–1837, 2013.

[21] D. P. Shepard and T. E. Humphreys, "Characterization of receiver response to spoofing attacks," in *Proc. ION GNSS*, Portland, Oregon, 2011, pp. 1–11.

[22] T. Nighswander, B. Ledvina, J. Diamond, R. Brumley, and D. Brumley, "GPS software attacks," in *Proc. ACM Conf. Comput. Commun. Security*, Raleigh, NC, USA, 2012, pp. 450–461.

[23] J. A. Larcom and H. Liu, "Modeling and characterization of GPS spoofing," in *Proc. IEEE Int. Conf. Technol. Homeland Security*, Waltham, MA, USA, 2013, pp. 729–734.

[24] A. Kim, B. Wampler, J. Goppert, and I. Hwang, "Cyber attack vulnerabilities analysis for unmanned aerial vehicles," in *Proc. Infotech@Aerospace Conf.*, Garden Grove, CA, USA, 2012, pp. 1–30.

[25] A. Shevtekar and N. Ansari, "A router-based technique to mitigate reduction of quality (RoQ) attacks," *Comput. Netw.*, vol. 52, no. 5, pp. 957–970, Apr. 2008.

[26] P. Sakarindr and N. Ansari, "Security services in group communications over wireless infrastructure, mobile ad hoc, and wireless sensor networks," *IEEE Wireless Commun.*, vol. 14, no. 5, pp. 8–20, Oct. 2007.

[27] Z. Gao and N. Ansari, "Tracing cyber attacks from the practical perspective," *IEEE Commun. Mag.*, vol. 43, no. 5, pp. 123–131, May 2005.

[28] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming sensor networks: Attack and defense strategies," *IEEE Netw. Mag.*, vol. 20, no. 3, pp. 41–47, May/Jun. 2006.

[29] Y. Ren, M. C. Chuah, J. Yang, and Y. Chen, "Detecting wormhole attacks in delay-tolerant networks," *IEEE Wireless Commun. Mag.*, vol. 17, no. 5, pp. 36–42, Oct. 2010.

[30] Y. Guo, S. Schildt, T. Pogel, and L. Wolf, "Detecting malicious behavior in a vehicular DTN for public transportation," in *Proc. IEEE Glob. Inf. Infrastruct. Symp.*, Trento, Italy, 2013, pp. 1–8.

[31] H. Wen, P. Y.-R. Huang, J. Dyer, A. Archinal, and J. Fagan, "Countermeasures for GPS signal spoofing," in *Proc. 18th Int. Tech. Meeting Satellite Div. Inst. Navig.*, Long Beach, CA, USA, 2005, pp. 1285–1290.

[32] N. Wisitpongphan, F. Bai, P. Mudalige, V. Sadekar, and O. Tonguz, "Routing in sparse vehicular ad hoc wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 8, pp. 1538–1556, Oct. 2007.

[33] Z. Zhang, M. Trinkle, L. Qian, and H. Li, "Quickest detection of GPS spoofing attack," in *Proc. IEEE Milcom*, Orlando, FL, USA, 2012, pp. 1–6.

[34] B. W. O'Hanlon, M. L. Psiaki, J. A. Bhatti, D. P. Shepard, and T. E. Humphreys, "Real-time GPS spoofing detection via correlation of encrypted signals," *Navigation*, vol. 60, no. 4, pp. 267–278, 2013.

[35] H. Sedjelmaci, S. M. Senouci, and M. A. Abu-Rgheff, "An efficient and lightweight intrusion detection mechanism for service-oriented vehicular networks," *IEEE Internet Things J.*, vol. 1, no. 6, pp. 570–577, Dec. 2014.

[36] B. Yu, C.-Z. Xua, and B. Xiao, "Detecting Sybil attacks in VANETs," *J. Parallel Distrib. Comput.*, vol. 73, no. 6, pp. 746–756, 2013.

[37] F. Mármol and G. Perez, "TRIP, a trust and reputation infrastructure-based proposal for vehicular ad hoc networks," *J. Netw. Comput. Appl.*, vol. 35, no. 3, pp. 934–941, 2012.

[38] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Network anomaly detection: Methods, systems and tools," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 303–336, 1st Quart., 2014.

[39] B. Scholkopf and A. J. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, U.K.: MIT Press, 2006.

[40] H. Sedjelmaci and S. M. Senouci, "A new intrusion detection framework for vehicular networks," in *Proc. IEEE ICC*, Sydney, NSW, Australia, Jun. 2014, pp. 538–543.

[41] S. Ganeriwal and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," in *Proc. 2nd ACM Workshop Security Ad Hoc Sensor Netw.*, Washington, DC, USA, 2004, pp. 66–77.

[42] H. Sedjelmaci, T. Bouali, and S. M. Senouci, "Detection and prevention from misbehaving intruders in vehicular networks," in *Proc. IEEE Globecom*, Austin, TX, USA, Dec. 2014, pp. 39–44.

[43] *Network Simulator (NS-3)*. (2016). [Online]. Available: http://www.nsnam.org

[44] O. Bouachir, A. Abrassart, F. Garcia, and N. Larrieu, "A mobility model for UAV ad hoc network," in *Proc. IEEE Int. Conf. Unmanned Aircraft Syst.*, Orlando, FL, USA, 2014, pp. 383–388.

[45] J. Xu and M. J. Chung, "Predicting the performance of synchronous discrete event simulation," *IEEE Trans. Parallel Distrib. Syst.*, vol. 15, no. 12, pp. 1130–1137, Dec. 2004.

[46] T. H. Hai, E.-N. Huh, and M. Jo, "A lightweight intrusion detection framework for wireless sensor networks," *Wireless Commun. Mobile Comput.*, vol. 10, no. 4, pp. 559–572, 2010.

[47] I. Khalil, S. Bagchi, and N. B. Shroff, "LITEWORP: A lightweight countermeasure for the wormhole attack in multihop wireless networks," in *Proc. IEEE Int. Conf. Depend. Syst. Netw.*, Yokohama, Japan, 2005, pp. 612–621.

[48] R. Mitchell and I.-R. Chen, "Effect of intrusion detection and response on reliability of cyber physical systems," *IEEE Trans. Rel.*, vol. 62, no. 1, pp. 199–210, Mar. 2013.

[49] Q. Zhang *et al.*, "Multimodel-based incident prediction and risk assessment in dynamic cybersecurity protection for industrial control systems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 10, pp. 1429–1444, Oct. 2016.

[50] O. Y. Al-Jarrah *et al.*, "Data randomization and cluster-based partitioning for botnet intrusion detection," *IEEE Trans. Cybern.*, vol. 46, no. 8, pp. 1796–1806, Aug. 2016.

[51] C. Zhou *et al.*, "Design and analysis of multimodel-based anomaly intrusion detection systems in industrial process automation," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 10, pp. 1345–1360, Oct. 2015.

[52] H. Sedjelmaci, S. M. Senouci, and N. Ansari, "Intrusion detection and ejection framework against lethal attacks in UAV-aided networks: A Bayesian game-theoretic methodology," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–11, Aug. 2016.

[53] H. Sedjelmaci, S. M. Senouci, and M.-A. Messous, "How to detect cyber-attacks in unmanned aerial vehicles network?" in *Proc. IEEE Globecom*, Washington, DC, USA, Dec. 2016, pp. 1–6.

[54] *Drones Civils—Parrot*. (2016). [Online]. Available: http://www.parrot.com/fr/drones/