Secure communication between UAVs using a method based on smart agents in unmanned aerial vehicles

Maryam Faraji-Biregani[1]

Abstract Unmanned aerial systems (UASs) create an extensive fighting capability of the developed military forces. Particularly, these systems carrying confidential data are exposed to security attacks. By the wireless's nature within these networks, they become susceptible to different kinds of attacks, hence, it seems essential to design the appropriate safety mechanism in such networks. The sinkhole attack is one of the most dangerous and threatening attacks amongst types of attack in UAS. A malicious UAV exists in such a threat attacking as a black hole for absorbing all traffic in the network. Mainly, in a Flow-based protocol, the attacker considers the requests on the route, then, it replies to the target UAV such as high quality or the best route towards Gard station. The malicious UAV is able to only insert itself on one occasion between the nodes relating to each other (such as sink node and sensor node), and act for passing packets among them. In this study, the malicious attacks are detected and purged using two stages were. In the first stage, some principles and rules are used to detect black hole, gray hole, and sinkhole attacks. In the second stage, using a smart agent-based strategy negotiation procedure for three steps, a defense mechanism is designed to prevent these attacks. The smart agent is used by reliable neighbors via the negotiation procedure for three steps, hence, the traffic formed by the malicious UAV is not considered. The suggested protocol is called SAUAS. Here, the technique is assessed through extensive simulations performed in the NS-3 environment. Based on the simulation outcomes, it is indicated that the UAS network performance metrics are enhanced based on the packet delivery rate, detection rate, false-negative rate and false-positive rate.

Keywords Unmanned Aerial Systems (UASs) . UAV . Sinkhole attack. IDS . Routing security

1 Introduction

✉ Maryam Faraji-Biregani
m.faraji@ashrafi.ac.ir

[1] Department of Computer Engineering, Shahid Ashrafi Esfahani non-profit University, Sepahanshahr, Esfahan, Iran

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

Unmanned Aerial Systems (UASs) are explained as any specific aerial vehicle communicating with other vehicles like an Aircraft with another Aircraft (A2A), a UAV with another UAV (U2U), and a UAV with an Aircraft (U2A), or communicating with stationary infrastructures like a UAV with Ground station (U2G) and an Aircraft with Traffic control tower (A2T) [1-3]. Nevertheless, in spite of the benefits of UAVs across different applications where the activities are not monitored by any pilot, they are potentially vulnerable to cyber risks. It strengthens the necessity of designing reliable and secure UASs and overcoming the difficulties to prevent destruction and damage to other systems and human lives. Some attacks like Blackhole (BH), Sinkhole (SH), illegally enter the system. When an attack affects an unmanned system, it is difficult to remove the threat and bring the system back online. It is worth to mention that the usual approaches for securing the data like intrusion detection or encryption are not adequate for coping with such risks. For elaborating, within the stated schemes, the actuator and sensor measurements compatibility factor are not considered with the control mechanism and physical procedure of the UAV that are considerable in the protection outline. Many invented unmanned aerial systems work on the imaging of the target's position and sending it to ground or aerial stations with a focus on a type of attack. Besides, problem of previous unmanned aerial systems was the mere attempt on eliminating a single attack type and were only resistant to it. If the system was subject to combined attack, it would be practically inoperative, and the intrusion operation would fail the system quickly [4-8].

In the suggested scheme, the malevolent UAV is strong against 4 lethal attacks (SH, and BH), hence, intrusive operations are rapidly recognized and eliminated from the top-secret data surveillance or spying missions. The suggested design, the critical standards of service quality are improved such as detection rates, packet delivery rate, false-negative rates and false-positive rates. In Figure 1, the typical UAV communication scenario is provided.

This paper presents contributions as follows:

- Analyze the UAS to find unknown attacks launched by malicious UAVs
- We formalized the SAUAS by checking its performance in terms of QoS after proving its loop freedom property.
- Analyze the sensitivity and robustness of the UAV against lethal security attacks.
- We have done a series of simulations to study the realistic effects of UAS environments on SAUAS.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

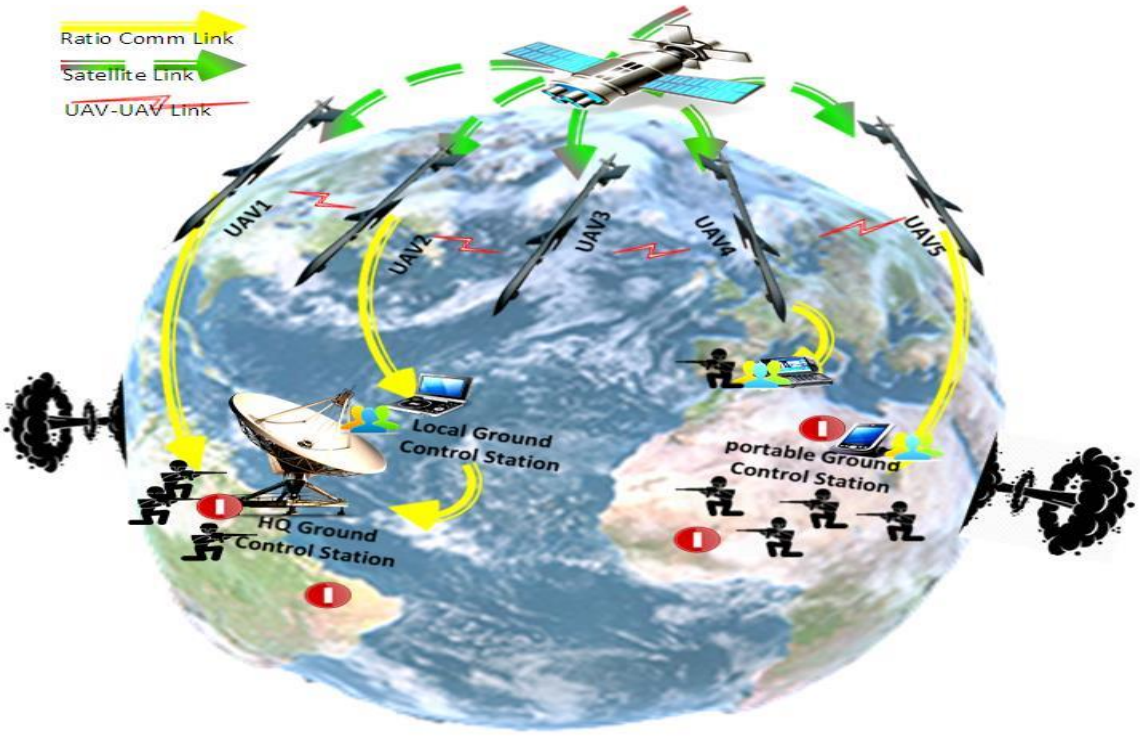


Fig. 1 Typical UAV communication scenario [1].

The presented work is structured as the following. Section 2 converses lethal security threats and detection schemes for UAS. Section 3 presents the Hash Function Algorithm (HFA). In Sect. 4 brings the proposed SAUAS strategy. The parameters used for assessing the performance are studied and simulation outcomes are deliberated in Section 5. Finally, conclusion of this research is discussed in Section 6.

2 Security attacks and detection schemes

We discuss the issues of cyber security risks targeting UAVs and detection outlines providing safety for the UAV in the following section.

2.1 Security Attacks

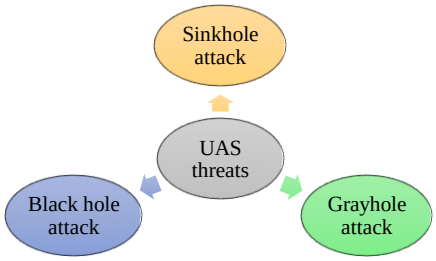
Unmanned Aerial Systems are susceptible to function degradation and cyber-security risks that are active or passive since they depend on wireless channels for communication. in Figure 2 displays a list of main cyber security threats that target UAVs. This study deals with the following susceptibilities:

- *Sinkhole Attack*: One of the main attacks threatening the UAVs is the attack known as the Sinkhole (SH) attack. In these attacks, a malicious node broadcasts illusive information regarding the routings to impose itself as a route towards specific nodes for the neighbouring nodes and thus, attract data traffic. The objective of this process is to draw all the traffic in the network towards the sinkhole node and as a result, alter the packets of data or silently drop them altogether. Sinkhole attacks can increase the network overhead, increase the consumption of energy and decrease the life time of the network , and ultimately annihilate the network [9].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

- *Black Hole Attack:* Here, a forged RREP is transmitted by a BH node once receiving an RREQ packet, requesting an unexpired and shorter route, even by the missed destination entry from the routing table. By reaching the source node by the created RREP packet, a route is established vis this malevolent intermediate node, to remove all legitimate RREP messages conveyed through destination and other intermediate nodes. Hence, through deceiving the source node, the data traffic is successfully attracted by the BH node to that destination. Then, the BH node drops all the data packets in place of forwarding the incoming messages. By forging a transmission route, the hop count is reset by the number of destination sequence to a very high value and the BH node to a very low value for increasing the acceptance chance at the source node. It is also possible for the BH attack to launch from the source node through making fields-source sequence numbers in hop counts and RREQ packets leading to poisoning the routing tables in intermediate nodes and the destination nodes [10].
- *Grey Hole Attack:* In this kind of attack, malicious nodes interrupt the data transmission in the network by conveying false routing data. Due to the unpredictable origin of the malicious nodes, a grey hole attack is considered as a BH attack extension. A node may act as both normal and malicious. Route detection procedure interrupted by is this attack while reducing throughput and packet delivery ratio [11].

Fig. 2 UAV cyber security threats



2.2 Detection schemes

Various security measurements have been developed and used in various ways to address cyber security threats and to protect the UAV against these attacks. It is not a recent issue, and there are extensive studies on it. Different approaches have been suggested by different studies to address these attacks.

This study supports using the movement data and each UAV's residual energy level for guaranteeing high-level communication stability while forecasting a sudden link breakage before occurring. Using a strong route detection process, routing paths are explored to take into account the link breakage prediction, the balanced energy consumption, and the connectivity level of the explored pathways [12].

In [13], a data dissemination method is provided by constructing a virtual topology based on the charge of WSN nodes using software-defined networks (SDNs) via UAVs. Constantly, the topology is monitored and reconfigured if necessary. For facilitating simultaneous communication with the ground nodes, the SDN controller and the base station, the aerial nodes are armed with multiple-input multiple-output (MIMO) antennas. Within the proposed method, an efficient sleep timer and back-off counter approaches are used as well. The topology formation and preservation of a sleep timer and a back-off counter are facilitated by the SDN controller.

The problem is intensified by a sporadic network connection disrupting communication in UAVs. Therefore, a drone requires a deep learning-based, adaptive Intrusion Detection System to recognize its intruders and guarantee its safe return-to-home (RTH). In the suggested IDS, using Self-Taught Learning (STL) with a multiclass SVM, the IDS's high true positive rate is maintained, even in unknown territory. The Deep-Q Network is used by the self-healing technique in the IDS recovery phase that is a deep reinforcement learning algorithm for dynamic route learning facilitating the safe

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

return home of the drone. Based on the simulation outcomes, the effectiveness of the proposed IDS is represented [14].

In [15], the UAV (physical layer security of an unmanned aerial vehicle) network is studied, in which the information is transmitted by a UAV-B (UAV base station) confidential to multiple information receivers (IRs) by assisting a UAV jammer (UAVJ) by existing the multiple eavesdroppers. Here, an optimization problem is formulated to mutually design the trajectories and convey the power of UAV-J and UAV-B for maximizing the minimum average secrecy rate overall IRs. The optimization problem is non-convex with the coupled optimization variables leading to the mathematically inflexible optimization problem. Hence, the optimization problem is decomposed into two subproblems and then solved using the succeeding convex approximation technique and an alternating iterative algorithm.

In [16], two aspects of secure communication and cooperative control are considered. The cooperative control is implemented by a clustering algorithm to increase the speed of converging the multi-UAV formation. Adjusting the flight control factor for accelerating the convergence of multi-UAV, a flock is created by the UAV group. For facilitating secure communication, the hierarchical virtual communication ring (HVCR) strategy is arranged to decrease the boundary of group communication and minimalize the insecure range.

In this paper, a method is proposed to maintain the security in UAV networks within surveillance, by verifying the data regarding events occurring from various sources. Hence, UAV networks are able to adapt peer-to-peer information stimulated by the blockchain principles and to discover the compromised UAVs in terms of trust policies. In the suggested method, secure asymmetric encryption is used with the official UAVs' pre-shared list. This method makes possible to detect the wrong information when hijacking an official UAV physically [17].

In [18], SCOTRES—a trust-oriented system is proposed for secure routing in ad-hoc networks to advance the network entities' intelligence using 5 innovative metrics. The resource consumption of each node is considered by the energy metric to impose similar quantity of collaboration and to increase the network's lifetime. The topology metric knows the positions of the nodes and improves the load balancing. The tolerance in periodic malfunctioning is provided by channel-health metric owing to bad channel circumstances and the network is protected versus jamming attacks. The collaboration of each subject for a particular network operation is evaluated by reputation metric to detect the specific attacks, however, the total compliance is estimated by trust metric, protecting against combinatorial attacks. The system's security features are validated by the Theoretic analysis.

This paper investigates the trajectory design and resource allocating for energy-efficient secure unmanned aerial vehicle (UAV) communication systems in which multiple legitimate ground users are served by a UAV base station while existing a potential eavesdropper. Our objective is to maximize the UAV's energy efficiency while optimization of its user scheduling, transmit power, velocity, and trajectory. The formulation of the design is a nonconvex optimization problem considering the minimum data rate requirement of each user, the maximum tolerable signal-to-noise ratio (SNR) leakage, and the location ambiguity of the eavesdropper. To attain an efficient suboptimal solution, an iterative algorithm is suggested [19].

This paper studies a joint optimization problem of ground terminals (GTs) association under wiretap channels, unmanned aerial vehicle (UAV) flight trajectory, and downlink transmission power. Precisely, a scenario is considered, in which a group of GTs is served by a UAV and the minimum secrecy rate is maximized to guarantee the fairness among GTs. We establish an iterative

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

algorithm in terms of the alternating and successive convex approximation (SCA) approaches for solving the nonconvex optimization problem [20].

Through unmanned aerial vehicles (UAVs), it is possible to support surveillance even in areas with no network infrastructure. By UAV networks, the security challenges are raised as a result of its dynamic topology. In the present study, a method is proposed to maintain the security in UAV networks within the framework of surveillance, by verifying data regarding events from various sources. Thus, UAV networks are able to adapt peer-to-peer general information stimulated by the blockchain ethics in terms of the trust policies. In this technique, secure asymmetric encryption is used with a pre-shared list of official UAVs. This work states detecting the misinformation when hijacking an official UAV physically [21].

In [22], an innovative trust model is proposed for UAVNs in terms of the mobility and performance pattern of UAV nodes and the features of inter-UAV channels. The suggested trust model includes 4 parts of the indirect trust section, the direct trust section, the trust update section, and the integrated trust section. According to the trust model, the perception of a secure link in UAVNs is formulated existing only a trust link and a physical link between two UAVs. Furthermore, the connectivity of UAVNs is analyzed by adapting the metrics of the secure connectivity probability and physical connectivity probability between two UAVs. Utilizing stochastic geometry with Doppler shift or without it, we originate analytical and accurate expressions of the secure connectivity probability and the physical connectivity probability.

In [23], a security model is suggested in terms of Identity Based (IB) authentication outline for UAV-integrated HetNets. the AVISPA tool is used to screen the absolutism of such a proposed scheme and some of its results indicated that our outline is resistant to the susceptibilities of intruders like replay, and impersonation.

Security threats targeting UAV systems are analysed in [24] and a cyber-security threat model is proposed that illustrates plausible attack directions.

In [25], the authors proposed a security framework to provide protection against malicious behavior targeting SFA communication systems in aircrafts. Table 1 present the previous works to design IDS for the UAV.

Table 1: Comparison between detection schemes for UAV

References	Attack type	Complexity	Robustness
[12]	Flooding	High	Medium
[13]	Hybrid	High	High
[14]	Jamming attacks	Low	Low
[15]	Physical layer security	Medium	Medium
[16]	Hidden terminals	Medium	High
[17]	Sybil, Blackhole attack	Medium	High
[18]	Signature-based	Low	Low
[19]	Physical layer	High	High
[20]	DoS	Medium	High
[21]	Physical-layer security	Low	High
[22]	GPS spooling attack and the Wi-Fi attack	High	Low
[23]	Packet modification attacks	Medium	Medium

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

3 Hash Function Algorithm (HFA)

Exhibiting the equivalence in the equation $H(m1) = H(m2)$, where m1 and m2 are two distinct input messages is troublesome through employing hash functions, since in a hash function, the delay of the processing speed must be minimised so that it can be computationally efficient [26]. A next generation of standard in security employed in electronic communications is SHA-3, which transforms the digital messages into "message digests" to register digital signatures. To facilitate the detection of modifications in the message originally sent, changing the original message modifies the message digest. The detailed necessities of the algorithms for secure routing as well as the crucial management services to achieve *encryption*(*encr*), *authentication*(*aut*) *and mechanism* for *digital signature*(*DS*) are presented in the subsection. This algorithm generates digital signature for any information given, transmitted between the source UAVs and the destination UAVs. In general form, the ECDSA is implemented in three phases: 1) Key pair generation, 2) Signature Generation (SG), and 3) Signature Verification (SV). It should be noted that phases 1 and 2 are performed in the source UAV, while the destination UAV carries out the last phase.

4.1 Registering with Trusted Authority

As demonstrated in Eq. (1), all the UAVs present in the network must select a random point, e.g. $Pi_R \in Zi_P^*$, to generate a private key and calculate the public key, e.g. Pi_{UTA}, via multiplying the private key with point generator, to perform registration.

$$Pi_{UTA} = Pi_{RTA} * Zi \tag{1}$$

For instance, registering to the network for an arbitrary source UAV (A) requires transmitting the identity (IDi_A) to the Trusted Authority (Ti_A). Once the identity is received, the trusted authority will then calculate the source node identity via multiplying the identity with the trusted source private key (Pi_{RTA}). Finally, the trusted authority will transmit back the generated $IDi_A^{'}$, as shown in the Eq. (2), to the source node.

$$IDi_A^{'} = IDi_A * PRi_{TA} \tag{2}$$

Moreover, the identity is checked in the source UAV (A) via multiplying $ID_A^{'}$ with the point generator. Furthermore, the identity of the source UAV is checked in the source UAV as well via multiplying ID_A with the public key from the trusted authority. The positive acknowledgment according to Eq. (3) is replied only if both points are equal.

$$IDi_A^{'} * Zi = IDi_A * PUe_{TA} \tag{3}$$

All the intermediate nodes as well as the destination UAVs must register to the network according to the similar procedure discussed in [26].

4.2 Employing EC Cryptography to Authenticate Nodes

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

Following the registration with trusted authority, a mechanism to authenticate all the registered UAVs is required to secure a network. To perform such authentication, public and private keys must be generated by all registered UAVs (e.g. to authenticate one hop source UAV and destination UAV) according to the ECC algorithm discussed in Sect. 4.1. First, the identity of the trusted authority (P) and the identity of source (K) are calculated, and are summed up to result in the secure identity information $D^{'}$. Next, the security associated information $A_{sum}\ (D^{'}, M, U)$ are sent to the destination UAV D from the source UAV. In this security associated information, $(D^{'})$ is the secure identity information, N is such that $M \in Zi_p^{*}$, p is a random associated number, and T is the time stamp, where:

$$P = ID_A^{'} * P\,\mathrm{Re}_A * PUe_B \tag{4}$$
$$K = ID_A * M * PUe_{TA}$$
$$D^{'} = P + K$$

Once the information $A_{sum}\ (D^{'}, M, U)$ is received, the target UAV verifies the timestamp. If the timestamp is verified, the validity of the security identity information $B^{'}$ is further checked by calculating $D^{'}$. Moreover, the identity of A $(ID_A^{'})$ is also calculated in the destination node via multiplying the random number with the generator point F. If the calculation results for $D^{'}$ and F are equal, the node A will be authenticated, and will be rejected otherwise. In Eq. (5), the complete procedure is demonstrated.

$$D = ID_A^{'} * P\,\mathrm{Re}_B * PUe_A \tag{5}$$
$$D^{'} = D^{'} - D$$
$$F = ID_A^{'} * M * G$$

To obtain an authenticated network, all the nodes registered in the network should follow the same procedure in a similar manner for mutual authentication [26].

4 The proposed SAUAS schema

A cyber-security threats-immune schema is designed in the following section utilizing the HFA algorithm. The suggested technique contains two phases including an overview of the SAUAS model discussed in Phase 1 and details of SAUAS schema explained in Phase 2. Within the proposed SAUAS, a hybrid solution is presented to protect the unmanned aerial systems that are effective in two aspects: First, it contains high detecting accurateness and low false-negative and positive rates and second, it quickly discovers and isolates attacks. Within the suggested scheme, the security issues are prevented including BH, SH, and GH attacks able to target the UAV. To discover the cyber-attacks with high precision, it is possible to add other properties rather than Table 2.

Table 2 Cyber-security attacks features

Cyber security threats	Features
Sinkhole attack	Data injection rate
Blackhole attack	Data injection rate
Grayhole attack	Data injection rate

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

4.3 Phase 1: Overview of the SAUAS model for detecting malicious UAV

The proposed algorithm is implemented on AODV protocol. According to the behavior of the nodes in the UAV network, we try to be able to identify and to eliminate malicious nodes in order to prevent of presenting wrong information to the checker UAV nodes in this algorithm. When the number of malicious nodes is increasing, as a result the number of sending request for comment would be increased. The overhead is increasing by an increase in the number of malicious nodes, because more nodes can start the process of sending request for comment or judgement. The more the overhead, the more the delay is. Therefore, by identifying malicious nodes the overhead and then the delay can be decreased. As the number of malicious nodes increases, the overhead of the algorithm becomes larger, as a result the identification of the malicious nodes becomes more difficult, for this purpose the delivery rate of the data decreases when the malicious nodes increase.

In order to detect the malicious UAV, we utilize the following rules in the proposed SAUAS:

- The UAV node has sent a number of data packet towards other UAV nodes, cannot be the malicious one.
- The UAV node has received many of the data but did not send them back, it may be the malicious one.
- The malicious UAV is one that has sent at least one RREP packet.
- The UAV node has received many of the data but did not send them back, and has sent at least one RREP packet, it is surely the one malicious UAV node.
- The node sending at least one response message of the route to the sender node of the route request earlier than other nodes, may be the malicious UAV node.
- The node that comprises the greatest sequence number and the lowest hop-count in its route response message, may be the malicious UAV node.

The principles of the proposed SAUAS are expressed as follows:

- The data corresponding to the activities of the nodes (number of sending data, number of receiving data, and number of receiving responses) is saved and investigated.
- Regarding the node of a neighbor that has sent at least one RREP packet, the requesting packet for comments among neighbors could be sent.
- The data saved in neighboring nodes regarding the sender node of RREP packet are received.
- The received data and declaration for the comment associating with the UAV node which is being malicious, are investigated.
- A packet of warning message was sent to quarantine the malicious UAV node and it`ll be distributed across the network.
- The UAV nodes under quarantine would be eliminated during routing process.

4.4 Phase 1: Details of the proposed SAUAS for purging malicious UAV from network

In this section, the Secure AODV focused on SAUAS which is based on smart agent's use. We used smart agents to aware the nodes from their valid neighbors to avoid of listening to the data generated by the malicious nodes. We will also explain the description and definitions regarding designing of agents and structure of memory in nodes, and how to use smart agents.

2.1.1 Designing of an agent

The smart agents are new and smart samples for distributed applications, and they are able to do their duties instead of human. In fact, a smart agent is a programmatic or executor code that

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

migrates among nodes as a form of agent packets. The format of the agent packets will be discussed in the next section. Following the paper, we will use agents and changeable packets of agents. There are two tangible and major differences among agents and other existing solutions that we will express them in the proposed method. First of all, in contrast to some of the previous methods, we only use one agent to identify the malicious UAV nodes. Second, the agents are not associated with each other, and instead they are in fact associated with each other through the fixed nodes. Therefore, the agent programs will be accelerated, as a result computational costs also are going to be decreased, and extra overheads won`t be existed in the agents` communications. Such optimization results in decrease in the energy consumption among nodes.

The format of the agent packets: The agent packet is encapsulated in an agent packet object as shown in (figure 3). As we use only one type of agent, therefore there is no need to save extra fields for type of an agent and their communicational style. Both the fields of the source node and the target node number are utilized for storing the source and target node identifier in a move, respectively. The agent program comprises a hash function -which is called “hash function of an agent”-, and two unique codes - which is called “code 3 and data code”. As code 3 is an output of different hash functions executions (e.g. Hash function of node), a copy of them (hash function of node and code 2) is stored in the node memory as a different unique code -which is called “code 2”. A hash function of agents and unique codes (e.g. Code 3) are utilized for interacting with valid nodes in the network and to create an availability set for sections corresponding with the data of the agents` packets. The data section consists of two fields: A valid bit and an agent bit. As these and those fields storing in the nodes are the same, we will explain them in the section of the node memory structure in more detailed. If the data section existing in the text is simple, an aggressor can easily use the data. Therefore, data code is used to encode the data and to ensure integrity of the data section in an agent packet. If an identified agent is a reliable node, then it sends the data code towards the node. Hence, the node can extract the correct data from agent packets by reducing the volume of the data section from data code. Nevertheless, if a reliable node seeks for adding new data to the data section, first of all it has to add them to the data code, and then, to interpolate them inside the agent packets. The data code and the data section have both the same size (e.g. 1 bit). The upper and lower boundary of the data section are used for storing a valid bit and an agent bit, respectively. At the time being, as an aggressor is not able to limit the smart agents, so it can not to access to the data code and real data within the data section. Moreover, an aggressor is not permission to fill a part of data with fake values. Finally, an object of agent packet provides some of the auxiliary methods to communicate with the nodes. The agent packet format used in this paper are provided Fig. 3.

Fig. 3 Agent packet format.

Agent packet	No of source UAV	No of destination UAV	Code 3	Hash function output	Data code	data

Agent migration: In our solutions, migration means to move an agent from one agent node to the one-hop neighbor’s node and to come back to its main node. Regarding this topic, there is no need to store a route for an agent migration within agent packet, since a smart agent moves only towards the one-hop nodes and comes back to its main own node. An agent node is a valid and general node which maintains a smart agent. In other words, an agent node accounts for a general node receiving an agent packet from its neighbor. The other definitions provided by us is term cycle of an

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

agent referring to it`s all one-hop neighbors as an agent migration as well as an agent node. In our solution, after a smart agent is placed on a node, then term cycle of an agent is randomly performed per 5-10 seconds. In addition, if two or Multi-agents deal with each other at the same time, the node will maintain one of them which had recently received with a little difference in time, and will ignored the rest. For this purpose, if an agent within a migration did not come back to its initial node after a particular time, the neighbor node won`t be immediately account for an aggressor; instead the given agent will be twice re-sent in the randomized periods. If an agent did not come back to its initial node afterwards, then it will be known as an aggressor.

2.1.2 The node memory structure

In contrast with other methods, we only maintain one-hop neighbors of a node in a table called "neighboring matrix" to store memory consumption within the node. The data stored in neighboring matrix included: the number of the node, a valid bit, and an agent bit. The number of the node is the number of one-hop neighbor node identification. A valid bit is utilized for determining reliable neighbors, the valid nodes within Secure AODV only receive or send the data from reliable nodes, and never communicate with the malicious nodes. As the smart agents are responsible for determining the reliability of a node, the agent nodes, in order to receive secure data only from the smart agents, have to be recognized. As a result, an agent bit is used for identifying the agent nodes. Before spreading the nodes in the environment, there are two unique codes in this regard: The code 1 and 2 are entering into their memory. The code 1 is not encoded, yet but it is in a simple case. While, code 2 within code 1 accounts for an output of an agent hash function execution (e.g. an agent hash function). The nodes were also equipped using hash function called "the node hash function" in which it is varying from those stored in the agents` program. These two codes and hash function are usable for detecting fake agents, and interaction with the valid nodes within network. Ultimately, the smaller section of the memory is used for habitation of roaming smart agents.

2.1.3 The Algorithm

Our proposed SAUAS is consisted of two steps for identifying and preventing of attacks in the following: The network deployment step illustrates how the network is being configured, while the UAS network maintenance step is an indicator of how the network security is maintained.

Deployment phase of the network: At the beginning, the nodes are uniformly distributed in the UAS network. Consequently, the neighbor randomly selects a number of the UAV nodes from smart agents according to the desired percentage to send the agent packets. When a node receives an agent packet from its neighbor that is known as an agent node. Following this step, all nodes spread the HELLO packets for finding existing nodes in the radio frequency range as well as creating neighboring matrix. It is completely clear that the malicious nodes can convert themselves into neighboring matrix. After finding neighbors process, each input within neighboring matrix encompasses identifiers of one-hop neighbor nodes, but an agent bit and the valid bit remain incorrect, yet. There is a wireless network as shown in the figure. Table 3, in which the nodes A & H show an agent, and E accounts for a malicious node, while the Table 4 indicates the nodes A & H, Then the node B after sending HELLO packets becomes neighboring matrix. As shown in the figure, during the present step, all neighboring nodes are found in which range consists of malicious cases.

Maintenance phase of the network: After deployment phase of the network, the agents commence with an agent cycle. But, before presenting any information to a node or receiving from that, a

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

three-step negotiation known as (the process of confidence-building) is performed among an agent and the node. If a node is reliable, the interactions would be commenced; otherwise, the given node accounts for an aggressor. If, the neighbor node is reliable, then valid bit related is correctly altered in neighboring matrix (for instance; one) when an agent returns to the main node (or an agent node), and Otherwise, it remains incorrect (e.g. zero); Moreover, if the neighbor node is an agent node, as a result it`s agent bit remains true (true). As only a percent of the nodes comprises an agent (like; agent nodes), the agent nodes send a packet of (confidence packet) to it`s all reliable neighbors after determining the reliability among neighbors, and tries to aware the agents of reliable and malicious nodes. Additionally, as the nodes are moving, the received signal strength is calculated when an agent returns towards an agent node. If it is lower than one threshold, the neighbor node is then eliminated from neighbors' list, but, it is assumed that the nodes are moving. If the eliminated agent bit of the node is incorrect, an agent node sends a control packet for re-performing of seeking for neighbor process, because it has no agent and by no means can be covered by smart agent.

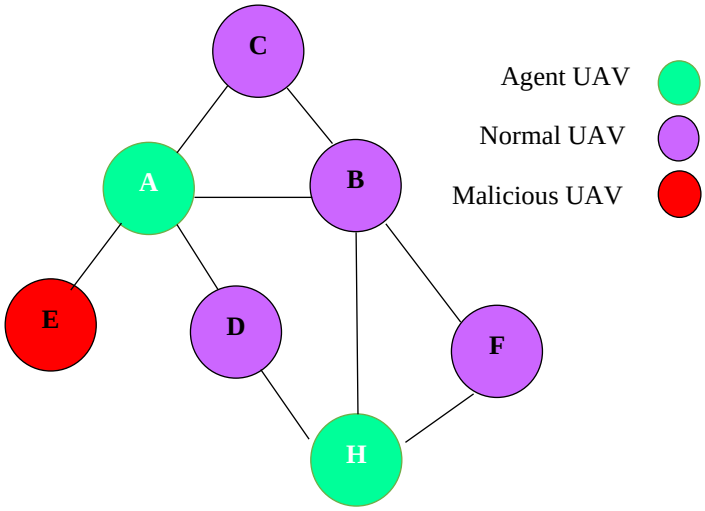


Fig. 4 UAS with agent UAV, normal UAV, and malicious UAV.

Table 3 the UAV_A, UAV_B, UAV_H are the neighboring matrix after sending HELLO packets.

UAV_A

UAV_{ID}	UAV_B	UAV_C	UAV_D	UAV_E
Valid	0	0	0	0
Agent	0	0	0	0

UAV_B

UAV_{ID}	UAV_A	UAV_C	UAV_F	UAV_H
Valid	0	0	0	0
Agent	0	0	0	0

UAV_H

UAV_{ID}	UAV_B	UAV_F	UAV_D
Valid	0	0	0
Agent	0	0	0

Table 4 the UAV_A, UAV_B, UAV_H are the neighboring matrix after an agent migration.

UAV_A

UAV_{ID}	UAV_B	UAV_C	UAV_D	UAV_E
Valid	1	1	1	0
Agent	0	0	0	0

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

UAV_B

UAV_{ID}	UAV_A	UAV_C	UAV_F	UAV_H
Valid	1	0	0	1
Agent	1	0	0	1

UAV_H

UAV_{ID}	UAV_B	UAV_F	UAV_D
Valid	1	1	1
Agent	0	0	0

Table 5 the UAV_B neighboring matrix after receiving reliable packets from the UAV_A, UAV_H.

UAV_B

UAV_{ID}	UAV_B	UAV_C	UAV_D	UAV_E
Valid	1	1	1	1
Agent	1	0	0	1

Table 5 illustrates the neighbors' matrix for the UAV_A, UAV_B, UAV_H just after an agent migration. The bits of an agent and the valid are filling with the relating values. But, as the node B has no agent, it will not enjoy any information about the UAV_C, UAV_F. So, the agent nodes (here UAV_A, UAV_H) send the confidence packets towards their reliable neighbors (as B). After receiving confidence packets, the UAV_B updates it`s matrix, as shown in (Fig. 4). Therefore, all nodes are aware of all of their one-hop neighbors, and the malicious nodes are being well identified in this regard. According to this case, the nodes receive/send routing data and data packets from their reliable neighbors as well, and they will not listen to the aggressors. A significant point including (the network longevity, energy level of the nodes, special agent nodes), if it goes away resulting in their demolition, as a result we designed one scheme for this task very well. When the energy between the general nodes is lower than the threshold, the neighbors will deal with a death packet, and then the death node is eliminated of the neighbors' list. But there is a little difference for the agent nodes. When an agent node is dying, it sends it`s agent to one of the reliable neighbors having no agent, and then acts as the general nodes. So, an agent is placed on the new node and operates like this.

2.1.4 Detection malicious attacks

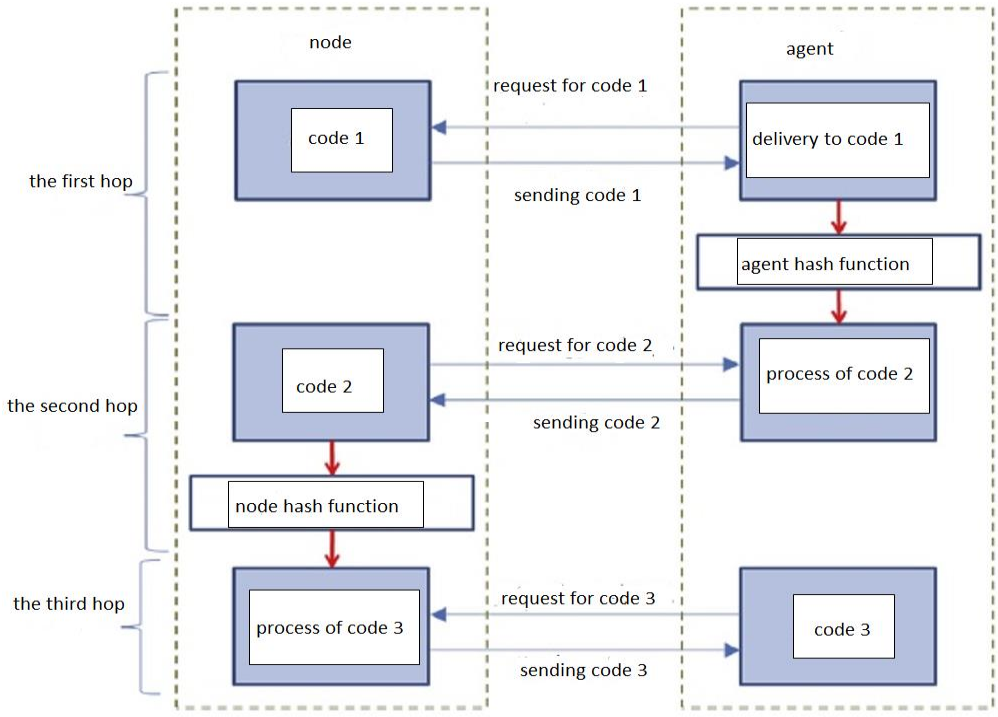
We explain how a SH UAV is being identified using codes (code 1, 2, and 3). Regarding movement of the nodes, the maintenance stage of the network has to be periodically performed during network longevity. Therefore, a malicious node is able to fake the identification of a valid node, it also can be placed in the list of valid neighbors of the nodes in which tries to configure neighboring matrix. As an aggressor does not know when a valid node wants to perform reconfiguration (means to update neighboring matrix), hence it has to permanently listen to the network traffic, so for this purpose this can lead to the rapid decrease in its energy, and it is immediately getting lost. As recently mentioned, before that a node attempting to send some data to a mobile agent, the node and the agent need to be confidence of each other. This is a confidence method as shown in figure 5. As shown in figure 5, a valid node includes main codes like; code 1, and code 2 while a valid agent has only code 3. In fact, we use this type of method in order to identify enemy nodes. The confidence method is expressed as follows:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

As shown in figure 5, after that an agent is going to be placed in the node, the agent then sends a request for code 1, and code 2 according to its unique hash function (e.g. Agent hash function) which is considered as stage 1, afterwards the agent sends code 2 to the node at the next stage. If it is matched with the code (2) stored in the node, it may trust to this agent, and also may conduct (for instance; node hash function) the code 3 using it`s unique hash function regarding code 2. It is the time for an agent to trust the node. In the last stage of the process of confidence, the code 3 is sent towards an agent. If it is equaled to a code stored in the source code of an agent, the agent then trusts to the node and sends the data code towards it. At the time being, there are three situations regarding confidence method:

A valid agent inside a valid node: In this situation, after that an agent was placed in the node, it is able to send a request for code 1, and code 2 is conducted as more recently expressed. Consequently, the representative sends the code 2 towards the node at the next stage. As the code 2, and the other one sent and stored inside the node are the same, the node will trust an agent. At the next stage, the valid node generates code 3 based on code 2, and then can send it towards an agent. Because of this code equals to the other one stored in an agent program, an agent would trust this node, and will send its data code towards it. Then, the node can extract the correct data from an agent packet or is able to add new data to that.

Fig. 5 The process of confidence from a smart Agent and the UAV.



A valid agent inside the enemy UAV: Similar to the previous situation, the agent requests for code 1, and then generates the code 2 using the agent hash function as well as sending it towards the UAV, and ultimately waits for delivering code 3. But the UAV is unable to send such correct code, because of the lack for the main code 1, and the method of UAV hash function. Thus, after that the agent received incorrect code 3, it will not trust again the given UAV and do not send its

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

data code any way. As a result, an aggressor UAV can`t mine or change the correct data from agent packets` data section.

A fake agent inside a valid UAV: In such circumstance, after that the fake agent was placed in such UAV, and when received code 1, then it won`t be able to present the correct code 2 for the UAV at the next stage, because this type of agent has no main method of the agent hash function. As such code is not the same was stored in the UAV, therefore the UAV does not trust the agent, and the agent is consequently ignored. Additionally, an aggressor UAV may involve the agent with some difficulty instead of returning it to the initial UAV or to conduct it towards wrong routes. In this regard, if an agent did not return after a particular time (according to migration`s definition), the neighbor UAV is then considered as an aggressor. The flowchart of proposed SAUAS is given in Fig. 6.

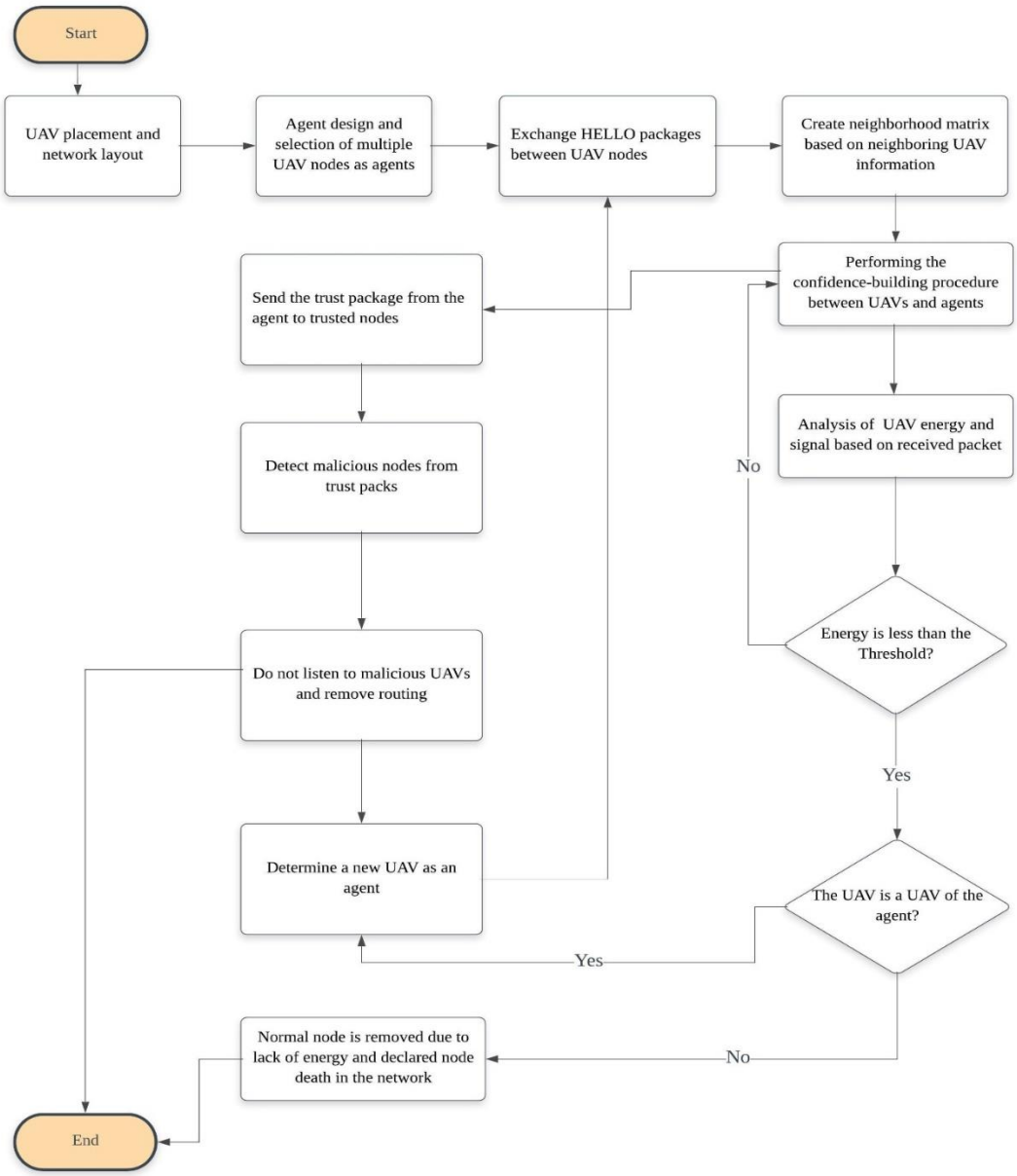


Fig. 6 Flowchart of the SAUAS.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

5 Evaluating the Performance

The SAUAS performance is assessed in the following section to avoid the cyber security attacks.

5.1 Performance metrics

Here, the performance and effectiveness of our suggested SAUAS method are systematically assessed with complete simulations. A comparison is performed between the results and with CS-AVN and CST-UAS methods proposed in [24], and [25], respectively. The false negative, false positive, and detection ratio are assessed. The meaning of notations and abbreviated used in the equations are given in Table 6 and Table 7.

Table 6 The parameters specified for *PDR*

Notations	Means
X_i	Number of packets received by node i
Y_i	Number of packets sent by node i
n	Experiments

Table 7 Abbreviated notations

Parameters	Description
FPR	*False positive rate*
FNR	*False negative rate*
TPR	*True positive rate*
TNR	*True negative rate*
DR	*Detection rate*
PDR	*Packet delivery rate*

5.1.1 PDR

PDR is the division of the total data packets received at the destination UAV, to the total number of data packets transmitted by the source UAV, described in percentage. The average PDR obtained for N experiments is demonstrated by Eq. (6).

$$PDR = \frac{1}{n} * \frac{\sum_{i=1}^{n} X_i}{\sum_{i=1}^{n} Y_i} * 100\% \quad (6)$$

5.1.2 FPR

The FP is calculated by the total number of UAVs wrongly detected as the malicious UAVs divided by the total number of normal UAVs [27,28]. Therefore, the is defined as illustrated in Eq. (6).

$$FPR = \left(\frac{FPR}{FPR + TNR} \right) * 100 \quad \text{Where:} \quad TNR = \left(\frac{TNR}{TNR + FPR} \right) * 100 \quad (7)$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

5.1.3 FNR

The rate of the malicious UAV to total normal UAVs that were mistakenly marked as a normal UAV [29-32]. Eq. (7) demonstrates the calculation.

$$FNR = \left(\frac{TPR + TNR}{All}\right) * 100 \qquad \text{Where:} \qquad TPR = \left(\frac{TPR}{TPR + FNR}\right) * 100 \tag{8}$$

5.1.4 DR

It is determined as the ratio of the number of lethal attack UAVs marked to the total number of existing lethal attack UAVs in the UAS. DR is calculated by Eq. (8). Table 8 lists the parameters used for DR [33-47].

$$DR = \left(\frac{TPR}{TPR + FNR}\right) * 100 \qquad \text{where} \qquad All = TPR + TNR + FPR + FNR \tag{9}$$

Table 8 The parameters specified for *DR*

Parameters	Description
TP	The *TP* is obtained from the whole number of marked lethal attack UAVs divided by the whole number of the lethal attack UAVs.
FP	The *FP* is obtained by the total number of UAVs improperly recognized as the lethal attack UAVs divided by the whole number of normal UAVs.
TN	The rate of the lethal attack UAVs being properly marked as a lethal attack UAV.
FN	The rate of the lethal attack UAV to whole normal UAVs being wrongly marked as a normal UAV.

5.2 The simulation environment

Because of the difficulty in debugging and implementing UAS in real networks, it is necessary to view simulations as a basic design tool. The primary benefit of simulation is that analysis is simplified and protocol is verified, mostly, it is evident in systems in large scales [40-44]. The performance of the suggested method is assessed in this part by the use of NS-3 as the simulation means, and the discussion on the obtained results is presented. It should be noted that it is assumed that all SAUAS, CS-AVN and CST-UAS settings and parameters are equal.

5.3 Simulation results

In this section, we analyze the security performance of SAUAS under the four attack scenarios (described in Table 9). These attacks are categorized into DoS attack. Some important parameters are listed in Table 9.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

Table 9: Parameters used.

Parameters	Value
Channel type	Channel/Wireless channel
MAC Layer	MAC/802.11. b
Traffic type	CBR
UAV speed	180 m/s
Layer of Transmission	UDP
Size of packet	512 Byte
Malicious rate	10%, 20%, 30%
Type of attacks	SH, BH, GH
Transmission range	30 M
Selection of target UAV	Random

Table 10-12 compares the performance of SAUAS with that of CS-AVN and CST-UAS in terms of FPR, FNR, and DR.

Table 10 Detection rate (10% malicious UAV of overall UAVs) of various approaches with varying degree of UAVs.

Number of UAV	Detection rate (%)		
	CST – UAS	*CS – AVN*	*SAUAS*
50	69.5	72.4	88.5
100	70.4	72.6	88.8
150	71.5	73.6	89.4
200	72.4	74.3	90.3
250	73.5	74.8	90.4
300	74.3	75.8	90.89
350	75.7	76.3	91.4
400	76.9	78.7	92.6
450	77.06	79.6	92.7
500	77.2	80.3	93.2

Table 11 Detection rate (20% malicious UAV of overall UAVs) of various approaches with varying degree of UAVs.

Number of UAV	Detection rate (%)		
	CST – UAS	*CS – AVN*	*SAUAS*
50	65.3	69.2	85.06
100	66.1	69.9	86.34
150	67.7	70.3	87.5
200	68.7	70.6	88
250	69.4	71.3	89.3
300	70.1	71.7	89.5
350	71.4	72.6	90.4
400	72.3	72.78	91.5
450	73.2	73.1	92.7
500	74.1	74.1	93.8

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

Table 12 Detection rate (30% malicious UAV of overall UAVs) of various approaches with varying degree of UAVs.

Number of UAV	Detection rate (%)		
	CST – UAS	*CS – AVN*	*SAUAS*
50	58.32	67.1	81.06
100	58.97	68.2	81.34
150	59.25	68.8	82.5
200	60.5	69.3	83
250	61.5	69.7	84.3
300	62.6	70.3	85.5
350	63.4	71.5	86.9
400	64.6	73.5	87.6
450	65.4	74.8	88.09
500	66.8	75.9	89.76

The average values of all methods under security threats is given in Table 13.

Table 13 Average values.

Schemes		*Detection rate*	*FNR*	*FPR*	*PDR*
CST – UAS	Number of UAVs (10% of overall nodes)	73.846	25.958	15.2	72.916
	Number of UAVs (20% of overall nodes)	69.83	21.394	21.87	70.106
	Number of UAVs (30% of overall nodes)	62.134	17.232	24.77	65.299
CS – AVN	Number of UAVs (10% of overall nodes)	75.84	22.23	14.04	77.737
	Number of UAVs (20% of overall nodes)	71.558	16.749	17.232	72.04
	Number of UAVs (30% of overall nodes)	70.91	13.53	21.392	69.227
SAUAS	Number of UAVs (10% of overall nodes)	90.819	12.583	9.911	95.327
	Number of UAVs (20% of overall nodes)	89.41	9.629	14.13	85.394
	Number of UAVs (30% of overall nodes)	85.005	8.305	15.68	78.947

DR: Figure 7 provides a comparison between the SAUAS suggested scheme, CS-AVN and CST-UAS models based on DR. (a): 10% malicious UAV, (b): 20% malicious UAV, and (c): 30% malicious UAV respectively. Based on the diagrams, the detecting rate in every 3 approaches is reduced in terms of scenarios, particularly with the high number of attacks. For the CS-AVN, this reduction is much higher compared to the other mechanisms. By the suggested design, it is possible to detect all the above attacks at a detection rate of over 95%. This finding is attained by the rate of malicious UAVs and the number of normal UAVs as 30% and 600, respectively. The proposed design is superior as a result of the fast detecting the malicious UAVs and removing them by mapping the antigenic and unsafe routes detected by the antibody-trained model and eliminated from the operation cycle.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

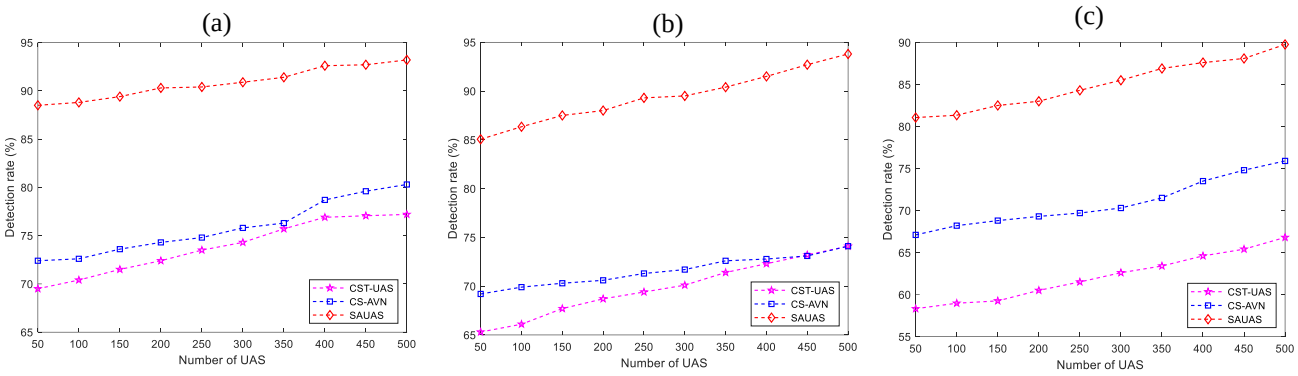


Fig. 7 Comparison of the SAUAS, CST-UAS and CS-AVN models in term of DR.

FNR: Figure 8 represents a comparison between the SAUAS suggested scheme, CST-UAS and CS-AVN models based on FNR in lethal attacks. (a): 10% malicious UAV, (b): 20% malicious UAV, and (c): 30% malicious UAV respectively. According to the diagrams, the FNR of the SAUAS suggested scheme incremented slightly, however, this value is much higher in the CST-UAS and CS-AVN. In Figure 8(a), the suggested scheme contains the FNR of less than 1.5% by the number of normal UAVs of 600, however, it is 12 and 17% respectively for the other two methods. In Figure 8(b), by the malicious UAVs rate of 15%, it is less than 4% in the suggested design that is 22% and 7% for the other two approaches respectively. In Figure 8(c) the FNR is explained under security threats with the 21% malicious UAV). Based on the results in Figure 8(c), it is indicated that in the conventional method, over security threats, the FNR at Misbehaving UASs ratio of 0.05 is around 1.3% increasing to around 9% at 0.30 in a misbehaving UAVs ratio condition.

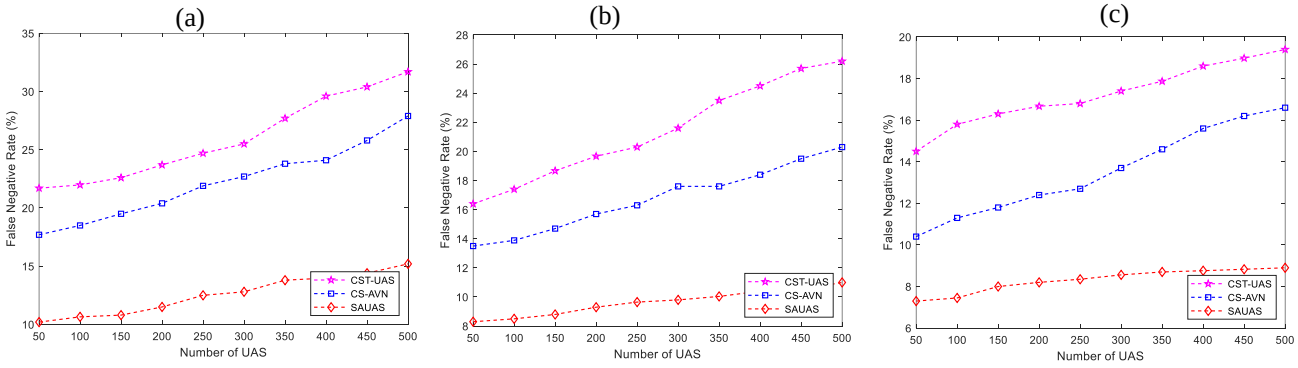


Fig. 8 Comparison of the SAUAS, CST-UAS and CS-AVN models in term of FNR.

FPR: in Figure 9, the comparison is provided between the suggested SAUAS framework against two methods of one risk-based Algorithm and game theory-based techniques. According to the Figure 9(a), by the number of normal UAVs within the range of 100-400 and increasing the rate of malicious UAVs from 0 to 30%, a slight and moderate growth existed in the FPR created by the suggested design in comparison to the other two designs. By the malicious UAVs rate and the number of normal UAVs of 30% and 400, respectively, the FPR of the SAUAS is less than 3%. Though, this quantity is 28% for the CST-UAS and 32% for the CS-AVN. The proposed design is superior as a result of its fast detection of malicious UAVs and removing them by collaboration between normal UAVs and ground stations that the process is carried out by the trained rules stored

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 November 2020 doi:10.20944/preprints202001.0316.v2

in memory. Moreover, it is superior by the fact that the suggested algorithm discovers the security threats and separates them from the UAS network, hence, the FPR caused by the attacks is reduced. Based on Figure 9(b) and (c), SAUAS reduces the FPR by over 29 and 39% compared to the CS-AVN and CST-UAS models, respectively.

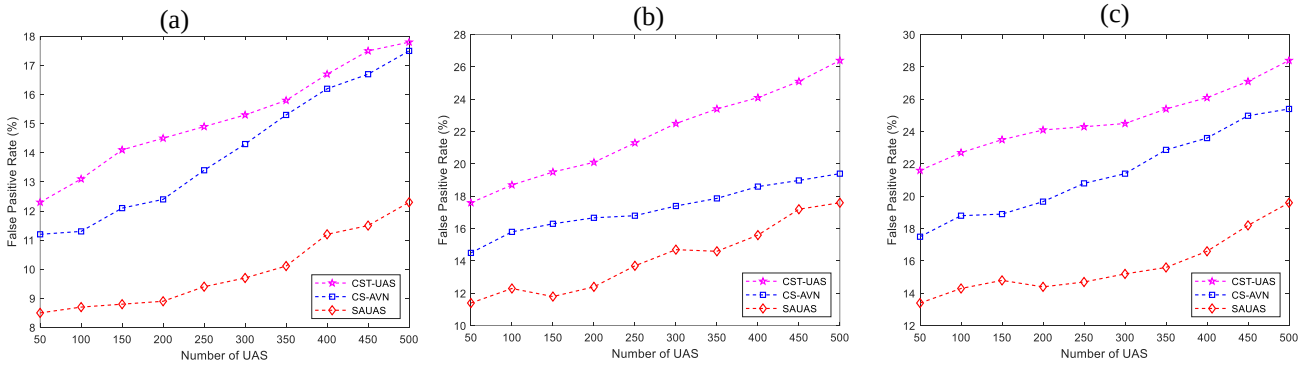


Fig. 9 Comparison of the SAUAS, CST-UAS and CS-AVN models in term of FPR.

In Figure 10, the association between the number of UAVs and PDR is represented. By the number of UAVs as 50, the PDR of CST-UAS and CS-AVN are relatively low, since some packets are not able to reach the destination prior to expiring the timeout period. By incrementing the number of UAVs, most packets can be delivered to the destination, hence, a slight enhancement is observed in the PDR. A slight degradation is observed in the packet delivery ratio of SAUAS, by the number of UAVs as 50 and 100 appearing due to random factors in simulation. Based on the overall trend, SAUAS outperforms both CST-UAS and CS-AVN based on the packet delivery ratio by the number of UAVs exceeding 150-500. According to Figure 10(a), (b) and (c), SAUAS reduces the PDR by over 32% and 22% compared to the CST-UAS and CS-AVN models, respectively.

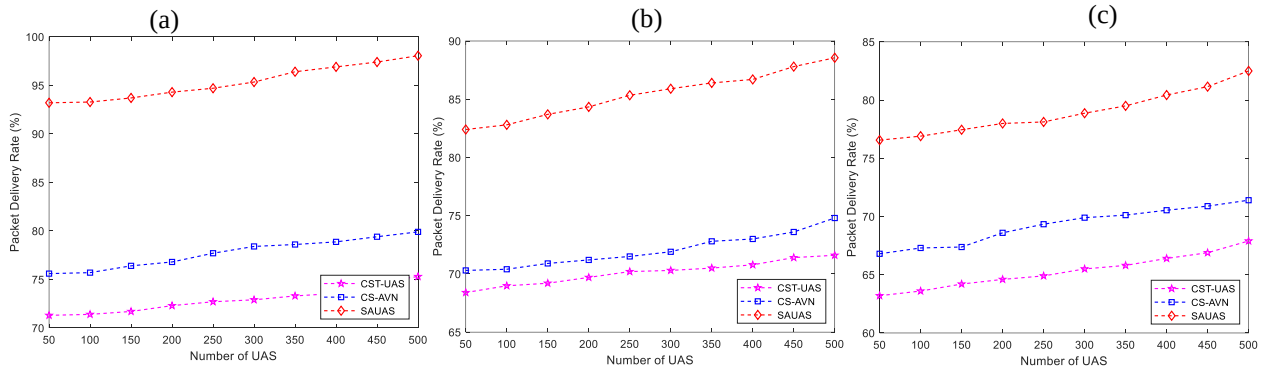


Fig. 10 Comparison of the SAUAS, CST-UAS and CS-AVN models in term of PDR.

6 Conclusion

By an increase in the use of UAV as well as to an easy implementation of these networks, these networks are being increased by day-to-day. Therefore, the security was known as a necessary need for providing the protected communications among UAVs. In order to overcome the challenges, there is a need to create a secure multi-mode solution achieving both vast protected mode and the performance of the desired networks. In this study, an intrusion detecting system was suggested in the

 doi:10.20944/preprints202001.0316.v2

SAUAS technique for protecting against the blackhole, sinkhole, and gray hole attacks utilizing the Hash Function Algorithm (HFA). The malicious UAV in the SAUAS, is strong against 4 lethal attacks (blackhole, gray hole, and sinkhole), hence, intrusive operations are rapidly recognized and eliminated from the spying missions or top-secret data surveillance. using the intrusion detection systems, the attempts to compromise the target system are detected and responded. In the first phase, we used a number of rules and principles for detecting these attacks. In the second phase, smart agent-based negotiation process for three-steps was used to design a defense mechanism for preventing of these attacks. We investigated the SAUAS scheme performance using NS-3. According to the results of the simulation, the SAUAS was highly powerful against security threats. It was demonstrated that it enjoys a low FPR (below 5.62%) and a high level of security, high detection rate (above 94.93%), and low FNR (below 2.92%) in comparison with present methods.

Conflict of Interest

None.

Reference

1. Sun, X., Ng, D. W. K., Ding, Z., Xu, Y., & Zhong, Z. (2019). Physical layer security in UAV systems: Challenges and opportunities. IEEE Wireless Communications, 26(5), 40-47.
2. Fu, Z., Mao, Y., He, D., Yu, J., & Xie, G. (2019). Secure Multi-UAV Collaborative Task Allocation. IEEE Access, 7, 35579-35587.
3. Shang, B., Liu, L., Ma, J., & Fan, P. (2019). Unmanned Aerial Vehicle Meets Vehicle-to-Everything in Secure Communications. IEEE Communications Magazine, 57(10), 98-103.
4. Won, J., Seo, S. H., & Bertino, E. (2019). A Secure Shuffling Mechanism for White-box Attack-resistant Unmanned Vehicles. IEEE Transactions on Mobile Computing.
5. Atoev, S., Kwon, O. J., Kim, C. Y., Lee, S. H., Choi, Y. R., & Kwon, K. R. (2019, July). The Secure UAV Communication Link Based on OTP Encryption Technique. In 2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN) (pp. 1-3). IEEE.
6. Sedjelmaci, H., & Senouci, S. M. (2018). Cyber security methods for aerial vehicle networks: taxonomy, challenges and solution. The Journal of Supercomputing, 74(10), 4928-4944.
7. Mitchell, R., & Chen, R. (2013). Adaptive intrusion detection of malicious unmanned air vehicles using behavior rule specifications. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 44(5), 593-604.
8. Oubbati, O. S., Mozaffari, M., Chaib, N., Lorenz, P., Atiquzzaman, M., & Jamalipour, A. (2019). ECaD: Energy-efficient routing in flying ad hoc networks. International Journal of Communication Systems.
9. Sayeed, M. A., Kumar, R., & Sharma, V. (2020). Efficient data management and control over WSNs using SDN-enabled aerial networks. International Journal of Communication Systems.
10. Arthur, M. P. (2019, August). Detecting Signal Spoofing and Jamming Attacks in UAV Networks using a Lightweight IDS. In 2019 International Conference on Computer, Information and Telecommunication Systems (CITS) (pp. 1-5). IEEE.
11. Zhou, X., Wu, Q., Yan, S., Shu, F., & Li, J. (2019). UAV-enabled secure communications: Joint trajectory and transmit power optimization. IEEE Transactions on Vehicular Technology, 68(4), 4069-4073.
12. Wu, J., Zou, L., Zhao, L., Al-Dubai, A., Mackenzie, L., & Min, G. (2019). A multi-UAV clustering strategy for reducing insecure communication range. Computer Networks, 158, 132-142.
13. García-Magariño, I., Lacuesta, R., Rajarajan, M., & Lloret, J. (2019). Security in networks of unmanned aerial vehicles for surveillance with an agent-based approach inspired by the principles of blockchain. Ad Hoc Networks, 86, 72-82.
14. Hatzivasilis, G., Papaefstathiou, I., & Manifavas, C. (2017). SCOTRES: secure routing for IoT and CPS. IEEE Internet of Things Journal, 4(6), 2129-2141.

15. Zaminkar, M., Sarkohaki, F., & Fotohi, R. A method based on encryption and node rating for securing the RPL protocol communications in the IoT ecosystem. International Journal of Communication Systems, e4693.
16. Faraji-Biregani, M., & Fotohi, R. (2020). Secure communication between UAVs using a method based on smart agents in unmanned aerial vehicles. The Journal of Supercomputing, 1-28.
17. Fotohi, R., Nazemi, E., & Aliee, F. S. (2020). An Agent-Based Self-Protective Method to Secure Communication between UAVs in Unmanned Aerial Vehicle Networks. Vehicular Communications, 100267.
18. Zaminkar, M., & Fotohi, R. (2020). SoS-RPL: Securing Internet of Things Against Sinkhole Attack Using RPL Protocol-Based Node Rating and Ranking Mechanism. WIRELESS PERSONAL COMMUNICATIONS.
19. Sarkohaki, F., Fotohi, R., & Ashrafian, V. (2020). An efficient routing protocol in mobile ad-hoc networks by using artificial immune system. arXiv preprint arXiv:2003.00869.
20. Mabodi, K., Yusefi, M., Zandiyan, S., Irankhah, L., & Fotohi, R. (2020). Multi-level trust-based intelligence schema for securing of internet of things (IoT) against security threats using cryptographic authentication. The Journal of Supercomputing, 1-26.
21. Seyedi, B., & Fotohi, R. (2020). NIASHPT: a novel intelligent agent-based strategy using hello packet table (HPT) function for trust Internet of Things. The Journal of Supercomputing, 1-24.
22. Fotohi, R., & Bari, S. F. (2020). A novel countermeasure technique to protect WSN against denial-of-sleep attacks using firefly and Hopfield neural network (HNN) algorithms. The Journal of Supercomputing, 1-27.
23. Fotohi, R. (2020). Securing of Unmanned Aerial Systems (UAS) against security threats using human immune system. Reliability Engineering & System Safety, 193, 106675.
24. Javaid, A. Y., Sun, W., Devabhaktuni, V. K., & Alam, M. (2012, November). Cyber security threat analysis and modeling of an unmanned aerial vehicle system. In 2012 IEEE Conference on Technologies for Homeland Security (HST) (pp. 585-590). IEEE.
25. Sedjelmaci, H., & Senouci, S. M. (2018). Cyber security methods for aerial vehicle networks: taxonomy, challenges and solution. *The Journal of Supercomputing*, *74*(10), 4928-4944.
26. Dilli, R., & Reddy, P. C. S. (2019). Robust Secure Routing Protocol for Mobile Ad Hoc Networks (MANETs). In *Innovations in Electronics and Communication Engineering* (pp. 393-399). Springer, Singapore.
27. Fotohi, R., Firoozi Bari, S., & Yusefi, M. (2020). Securing wireless sensor networks against denial-of-sleep attacks using RSA cryptography algorithm and interlock protocol. International Journal of Communication Systems, 33(4), e4234.
28. Jamali, S., Fotohi, R., & Analoui, M. (2018). An artificial immune system based method for defense against wormhole attack in mobile adhoc networks. Tabriz Journal of Electrical Engineering, 47(4), 1407-1419.
29. Jamali, S., & Fotohi, R. (2017). DAWA: Defending against wormhole attack in MANETs by using fuzzy logic and artificial immune system. the Journal of Supercomputing, 73(12), 5173-5196.
30. Fotohi, R., Heydari, R., & Jamali, S. (2016). A Hybrid routing method for mobile ad-hoc networks. Journal of Advances in Computer Research, 7(3), 93-103.
31. Jamali, S., & Fotohi, R. (2016). Defending against wormhole attack in MANET using an artificial immune system. New Review of Information Networking, 21(2), 79-100.
32. Fotohi, R., Ebazadeh, Y., & Geshlag, M. S. (2016). A new approach for improvement security against DoS attacks in vehicular ad-hoc network. International Journal of Advanced Computer Science and Applications, 7(7), 10-16.
33. Fotohi, R., & Jamali, S. (2014). A comprehensive study on defence against wormhole attack methods in mobile Ad hoc networks. International journal of Computer Science & Network Solutions, 2, 37-56.
34. Fotohi, R., Jamali, S., Sarkohaki, F., & Behzad, S. (2013). An Improvement over AODV routing protocol by limiting visited hop count. International Journal of Information Technology and Computer Science (IJITCS), 5(9), 87-93.
35. Fotohi, R., Jamali, S., & Sarkohaki, F. (2013). Performance Evaluation of AODV, LHC-AODV, OLSR, UL-OLSR, DSDV Routing Protocols. International Journal of Information Technology and Computer Science (IJITCS), 5, 21.
36. Fotohi, R., & Effatparvar, M. (2013). A cluster based job scheduling algorithm for grid computing. International Journal of Information Technology and Computer Science (IJITCS), 5(12), 70-77.

37. Cai, Y., Wei, Z., Li, R., Ng, D. W. K., & Yuan, J. (2019). Energy-efficient resource allocation for secure UAV communication systems. arXiv preprint arXiv:1901.09308.
38. Li, Z., Chen, M., Pan, C., Huang, N., Yang, Z., & Nallanathan, A. (2019). Joint Trajectory and Communication Design for Secure UAV Networks. IEEE Communications Letters, 23(4), 636-639.
39. García-Magariño, I., Lacuesta, R., Rajarajan, M., & Lloret, J. (2019). Security in networks of unmanned aerial vehicles for surveillance with an agent-based approach inspired by the principles of blockchain. Ad Hoc Networks, 86, 72-82.
40. Yuan, X., Feng, Z. Y., Xu, W. J., Wei, Z. Q., & Liu, R. P. (2018). Secure connectivity analysis in unmanned aerial vehicle networks. Frontiers of Information Technology & Electronic Engineering, 19(3), 409-422.
41. Rashid, A., Sharma, D., Lone, T. A., Gupta, S., & Gupta, S. K. (2019, July). Secure communication in UAV assisted HetNets: a proposed model. In International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage (pp. 427-440). Springer, Cham.
42. Dasgupta, D. (Ed.). (2012). Artificial immune systems and their applications. Springer Science & Business Media.
43. Chen, J., Feng, Z., Wen, J. Y., Liu, B., & Sha, L. (2019, March). A Container-based DoS Attack-Resilient Control Framework for Real-Time UAV Systems. In 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE) (pp. 1222-1227). IEEE.
44. Lei, K., Zhang, Q., Lou, J., Bai, B., & Xu, K. (2019). Securing ICN-Based UAV Ad Hoc Networks with Blockchain. IEEE Communications Magazine, 57(6), 26-32.
45. Yihunie, F. L., Singh, A. K., & Bhatia, S. (2020). Assessing and Exploiting Security Vulnerabilities of Unmanned Aerial Vehicles. In Smart Systems and IoT: Innovations in Computing (pp. 701-710). Springer, Singapore.
46. Haque, M. S., & Chowdhury, M. U. (2019, November). Ad-Hoc Framework for Efficient Network Security for Unmanned Aerial Vehicles (UAV). In International Conference on Future Network Systems and Security (pp. 23-36). Springer, Cham.
47. Lodeiro-Santiago, M., Caballero-Gil, P., Aguasca-Colomo, R., & Caballero-Gil, C. (2019). Secure UAV-Based System to Detect Small Boats Using Neural Networks. Complexity, 2019.