R3ACWU: A Lightweight, Trustworthy Authentication Scheme for UAV-Assisted IoT Applications

Muhammad Adil, *Member, IEEE*, Hussein Abulkasim, *Member, IEEE*, Ahmed Farouk, *Senior Member, IEEE*, and Houbing Song, *Fellow, IEEE*

***Abstract*— The technology of Unmanned Aerial Vehicles (UAVs) has sparked a revolution in numerous Internet of Things (IoT) applications, such as flood monitoring, wildfire monitoring, coastal area surveillance, intelligent transportation, and classified military operations, etc. This technology offers several advantages when used as a flying base station to enhance the communication metrics of an employed IoT appplication. However, as an integrated technology (UAV-assisted IoT applications), it suffers from many challenges, and security is one of the foremost concerns. Considering that, in this paper, we proposed a hybrid lightweight key exchange authentication model for UAV-assisted IoT applications to resolve the device-to-device (D2D) authentication and data privacy issues in these networks. The proposed model employs five different security parameters named registration, authentication, authorization, accounting, and cache wash and update (R3ACWU) in coordination with a hash function. The network architecture consists of UAVs, IoT devices, and micro base stations, followed by base stations, authentication servers, and service providers (SP). In this framework, we introduce a concept known as 'dead time', a specific time period after which each device's cache memory is cleared and updated. This practice not only enhances the security of the devices in use but also reduces computational and memory overhead by eliminating the records of devices that haven't participated in the communication process within the specified time frame. Results statistics of our lightweight R3ACWU authentication scheme exhibit notable improvement corresponded to the present authentication schemes in terms of comparative parameters.**

***Index Terms*— UAV-assisted IoT applications, device-to-device authentication, data privacy, R3ACWU protocol, cryptography, public and private key exchange.**

I. Introduction

WITH the advancement in technologies, Internet of Things (IoT) has gained significant popularity and importance across various applications. In some cases, IoT devices collect data from inaccessible areas where human access is impossible [1]. Nevertheless, this technology still manages to fulfill users' demands. To enhance the effectiveness of these applications, we propose the integration of Unmanned Aerial Vehicles (UAVs) as a technological solution. It is worth noting that such integration has the potential to significantly improve the productivity and efficiency of IoT applications used for tasks such as flood monitoring, wildfire surveillance, coastal area observation, intelligent transportation, and classified military operations, etc. However, despite its usefulness in harsh, complex, and inaccessible environments, this technology poses several challenges, including security, authentication, network robustness, and durability [2]. Considering the contributions of these networks, ensuring data privacy and D2D authentication is vital to attain the attention of consumer market stakeholders. Because an adversary attempts to tamper with legitimate devices or communication channels to compromise the system [3], [4]. Considering the potential consequences of possible attacks, security in these networks becomes an indispensable part. Therefore, it is crucial to design lightweight, reliable, and robust authentication schemes to maintain stakeholder trust.

In [5], Tanveer et al. propose an intelligent Authentication Key Exchange (AKE) protocol in conjunction with encryption algorithms to tackle the security challenges within the Internet of Drones (IoD) networks. Zhang et al. [6], continued this conversation and suggested a gateway-oriented key exchange protocol utilizing two server architectures for UAV-assisted IoT applications to resolve the authentication problem. Even though the idea was good, but, the complex authentication process followed by a high implementation cost of the proposed model depreciates its use in the real deployment. Deebak et al. [7], furthermore extend this discussion by proposing a lightweight privacy-preservation scheme (L-PPS) for Smart Internet of Drones (S-IoD) to reduce the authentication complexity during the session establishment phase and improve the validation process. Similarly, Islam et al. [8] introduced a blockchain-based authentication model designed to validate UAVs. This innovative approach employs body sensor hives (BSHs) in conjunction with token shares to facilitate low-power secure communication among connected entities. References [9], [10], and [11] provide a comprehensive overview of current research pertaining to UAV security by setting a trajectory for future study.

Manuscript received 1 July 2023; revised 25 September 2023; accepted 20 November 2023. This work was supported in part by the U.S. National Science Foundation under Grant 2309760 and Grant 2317117. The Associate Editor for this article was M. Bilal. *(Corresponding author: Houbing Song.)*

Muhammad Adil is with the Department of Computer Science and Engineering, University at Buffalo, The State University of New York, Buffalo, NY 14260 USA (e-mail: muhammad.adil@ieee.org).

Hussein Abulkasim is with the College of Engineering and Technology, University of Science and Technology of Fujairah, United Arab Emirates, and also with the Faculty of Science, New Valley University, El-Kharga 72511, Egypt (e-mail: h.hussein@ustf.ac.ae).

Ahmed Farouk is with the Department of Computer Science, Faculty of Computers and Artificial Intelligence, South Valley University, Hurghada 83523, Egypt (e-mail: ahmed.farouk@sci.svu.edu.eg).

Houbing Song is with the Department of Information Systems, University of Maryland, Baltimore County, Baltimore, MD 21250 USA (e-mail: h.song@ieee.org).

Digital Object Identifier 10.1109/TITS.2023.3342831

Taking into account the existing literature gap, our main objective in this study is to create an efficient, lightweight authentication model for UAV-assisted IoT applications. To achieve this, we introduce a hybrid key exchange authentication model that incorporates virtual authentication servers (VAS) within the primary authentication server (AS). This approach is designed to streamline the validation process for legitimate devices while reducing both computational cost and complexity. To explore, Initially, a UAV and IoT device registers its private key with the Authentication Server (AS) via the relevant base station. After that, the AS issues a public key and directs the device to a specific Virtual Authentication Server (VAS) for handling its authentication process within the operational network. Similarly, the Service Provider (Server) (SP_S) ensures the execution and distribution of various services during the communication process. Furthermore, we set up a collaborative environment of Registration, Authentication, Authorization, Accounting, and Cache Wash and Update (R3ACWU) with a "dead time" to validate legitimate devices in the network.

The main contributions of this work are summarized as below:

1) The foremost contribution of this study is the development of a Lightweight Key Authentication Scheme for UAV-assisted IoT applications. This scheme aims to proficiently handle the registration, authentication, authorization, accounting, and cache wash and update (R3ACWU) processes for legitimate devices in a communication friendly environment.
2) Moreover, the R3ACWU authentication scheme collaborates seamlessly with AS, VAS, and SP_{RS} which leads to a substantial reduction in authentication overhead and improve the communication metrics at the endpoint.
3) We employ the Elliptic Curve Diffie-Hellman Problem Algorithm (ECDHPA) and the Elliptic Curve Discrete Logarithm Problem Algorithm (ECDLP) in conjunction with R3ACWU authentication model. This integration serves to ensure the legitimacy of the participating devices with a focus on encryption and decision-making processes to minimize the computation complexity.
4) Finally, we conducted formal and informal security analyses to confirm the usefulness of the proposed model against various types of attacks followed by authentication accuracy, computation complexity, and communication cost in presence of existing state-of-the-art techniques.

The structure of the remaining article is as follows: Section II provides a review of existing authentication schemes, while in Section III, we outline the threat model. The evaluation of the proposed R3ACWU authentication scheme is presented in Section IV, and Section V includes a comprehensive formal and informal security analysis along with comparative results. Finally, Section VI provides the conclusion of the paper.

II. Related Work

In general, the security of UAV-assisted IoT applications differs from that of conventional IoT applications due to the mobility of UAVs. Regardless, researchers have shown a keen interest in this domain by developing cryptographic methods, key exchange protocols, and secure routing mechanisms to ensure data privacy and device-to-device (D2D) authentication within these networks. In this context, reference [12] introduced a secure data retention scheme based on the Advanced Encryption Standard (AES) and Blowfish. Tan et al. [13] introduced a distributed key management and authentication scheme employing a blockchain-based communication infrastructure for flying ad hoc networks (FANET). In this model, FANET participants autonomously to disseminate their public/private keys via cluster heads in the network to identify and relocate malicious devices seamlessly. Yahuza et al. [14] extended this discussion, and proposed a secure lightweight proven authenticated key agreement (SLPAKA) mechanism for Internet of Drones (IoD) networks. However, the complicated authentication process of the suggested model generates network overhead at the client-side, which affects the network performance in terms of effective communication and minimizes its use in real implementation.

Reference [15] proposes an event-triggered super twisting algorithm for authenticating UAVs networks. Alladi et al. [16] present a lightweight mutual authentication model for UAVs that utilizes Physical Unclonable Functions (PUFs) in conjunction with ground stations (GS). Sun et al. [17] proposed a self-characteristic-based authentication model utilizing a watermark framework at both the client-side and VAUs side. This framework ensures the legitimacy of interconnected devices followed by data integrity in the network. A payload-based authentication framework with coordination of Q-band frequency radiation was proposed by Paonessa et al. [18], for UAVs networks. To continue this discussion, a Software-Defined Networking (SDN) authentication paradigm was proposed in [19], to resolve the security issues in UAV-embedded cyber-physical systems (CPS). Likewise, reference [20] suggested a secure data distribution protocol for an Internet of Vehicle network. However, this scheme lacks formal security analysis that doubts the effectiveness of this model against various types of attacks. For the social Internet of Vehicles network, Gulati et al. [21] proposed a module-based communication infrastructure to facilitate secure and reliable data propagation. However, the high computational and communication costs associated with this model limit its practical deployment.

In [22], the author proposed an authenticated key agreement (AKA) scheme to resolve the authentication issues in vehicular networks. In the first part of this model vehicles, roadside units (RSU), and TA authenticate each other, while in the second part, key agreement processes were managed to ensure the legitimate entities authentication process. In [23], the author proposed a Chaotic Map-Based Authenticated Key Agreement (CMAKA) scheme for Autonomous Vehicles (AVs) to resolve their authentication issues. Moreover, they used a physical unclonable function (PUF) to generate trusted private keys during the validation process to ensure the legitimate vehicle authentication. In [24], the author discussed several security challenges of UAV-assisted IoT applications.

In [25], a new multi-factor key exchange (T-MFAKE) authentication scheme was proposed for UAV-assisted IoT applications utilizing the real communication threshold values of interconnected devices to ensure the validation, verification, and legitimacy in the network. Reference [26], presents a wireless physical-layer identification (WPLI) technique overview in the context of both theoretical modeling and experiment validation by taking into account real operation scenarios. Although, WPLI is deemed to be a promising technique to resolve the security issues in a wireless network such as UAV-assisted IoT applications. However, it is yet not clear whether the present WPLI approaches can be used in real-world scenarios or not. In [27], a radio frequency (RF) based secure data transmission scheme was proposed utilizing the satellite links with a realistic system. In this scheme, various wireless communication factors were ignored such as fidelity, attenuation, latency, and antenna gain, etc, which can raise questions about the proposed model. Reference [28], provides a comprehensive assessment of civilian drone security, privacy, and safety problems. The author, in particular, outlined physical and cyber threats that could harm the system, operation, and environment.

III. Threat Model

In this section, we discuss the threat model for the R3ACWU authentication prototype. Unlike traditional IoT applications, UAV-assisted IoT applications are expected for pervasive and real-time data collection, and task execution. Consequently, it is reasonable to assume that the Authentication Server (AS) is the sole fully trusted component within the network. Conversely, other entities such as UAVs, IoTs, and various networking components are untrusted due to their susceptibility to be compromise by outside attacker. To explore, let's assume an attacker named AK, interested to hijack the security of the employed UAV-assisted IoT application. To successfully exploit a legal device, an AK can originate fake message requests followed by additional control commands to gete an unauthorized access to the network and disrupt its normal functionality. Considering this scenario, we need to summarize a threat model with possible attacks assumption such as mentioned below:

1) An Ak has the ability to take full or partial control of legitimate devices by intercepting legitimate network traffic and tampering with or replacing authentication keys. (Forgery attack based on tampered/replaced authentication Keys).
2) Next, an AK has the ability to use some power interpretation procedures to identify and extract the authentication parameters of a UAV, IoT, and other networking device to compromise it's security. (Forgery attacks based on UV_{ID} & S_{ID}).
3) An AK has also the understanding how to employ and venture offline/online inference key matching attacks to get access to the network to collect legitimate information. (Forgery attack based on previous authentication keys).
4) An AK has the capabilities to obtain and use random numbers to calculate the session key of a legitimate devices to validate and start communication in the networks via compromised device. (Forgery attack based on random keys generation).

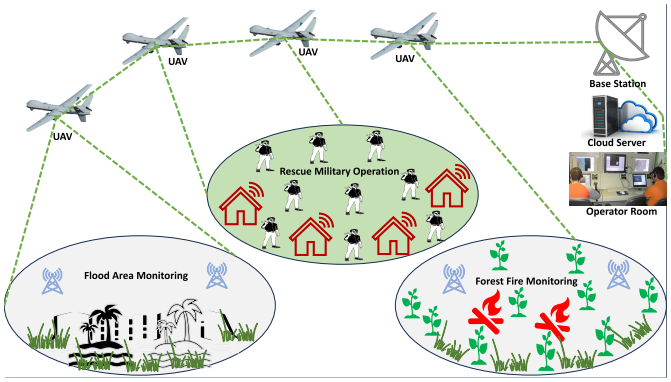

Fig. 1. Graphical representation of UAV-assisted IoT applications.

TABLE I
SYMBOL AND NOTATIONS DESCRIPTION

Notation symbol	Notation symbol description
A_i	authentication server
VA_i	i^{th} virtual authentication server
UV_i&S_i	Number of UAVs and IoT device, where the i^{th} represent the total numbers
SP_S	service provider
MH(.)	One way MD5 hash function
$\|$, $\bigoplus$	Bitwise concatenation and XOR operations
AK_i	i^{th} attacker in the network
$SKV_{i,j}$	session key of two UAVs
EL_c (a,b)	Non-singular elliptic curve $y^2 = x^3 + ax + b * (\text{mod q})$, where a,b $\in Z_q$, which is = $\{0, 1, 2, 3, 4, 5.....q-1\}$
BP	Base point in EL_c (a,b)
x.q	multiplication point in EL_c
BP + q	addition point in EL_c
PK_{pri}	rivate key
PK_{pub}	public key
CMSA	carrier sense multiple access
PK_{pri}, PK_{pub}, VAS	Private and public key pairing in VAS
T_{UV_i},T_{VAS_i}, T_{UV_j},	Current time stamp of paired UV_i and virtual authentication server
$\triangle T_{Trans}$	Transmission delay
G_{rk}	random key generator

IV. Proposed Methodology

In this section, we evaluate the operational phases of our hybrid lightweight key exchange-based authentication model. These strategies encompass the processes of legitimate device registration, verification, authentication, and cache wash and update. However, before going into the detailed discussion, first, we want to show the UAV-assisted IoT applications paradigm in graphical format and show how they work in the cooperative environment Fig 1. Additionally, we will provide an overview of the essential elements of the proposed model and present a table I, that summarizes the symbols and notations used throughout this work.

A. Preliminaries

Our Lightweight-R3ACWU scheme is practiced and executed in collaboration with cryptographic techniques that

incorporate encryption, decryption, aggregation, private and public-key exchange management in an operational network. To narrow down this, in the consequent sections, we will examine the footsteps of the considered preliminaries by following their operational procedure.

1) Bilinear Pairings: Let's assume a cyclic adoptive graph (e) with parameters G, whereas G = e : $G1$ and $G2$, which is generated by a prime number (P_r), whose order is a prime q. The multiplicative group of these parameters for prime q can be written as, e : $G1 \times G1 \Rightarrow G2$, which is further simplified as below:

1) Bi-linearity: Next, let suppose, there are two random integers (RI) in e such that a and b, which are generated by G and simplified with a,b $\in Z_q^e$.
   The given circumstances can be further generalized as, $\exists$, RI($G_1^a \times G_1^b \Rightarrow G_2^{ab}$) = $(G_1 \times G_1 \Rightarrow G_2)^{ab}$
2) As we know, If $G_1^a \times G_1^a \in G_2^{RI}$, then the pairing of RI $\neq 1$, $\forall$ G_1 & $G_2 \neq 1$.

2) Computation Complexity Assumption: In this subsection, we went through the computation assumption parameters. Initally, we assumed the variables a,b,c $\in Z_q^{RI}$. These variables are further interlinked with, e $\exists$ G_1^a, G_1^b&$G_2^{ab} \in G$ in a cyclic graph, which is generalized as, e: $G_1^a \times G_1^b \in G_2^{ab} = RI$ $\forall$ RI $\in$ (S, G). Keeping in view the proposed model, the following assumption have made to calculate the computation complexity:

1) Elliptic Curve Diffie-Hellman Problem Algorithm (ECDHPA): To manage this, let's consider a problem in an additive group (G = $G_1^a \times G_1^b = G_2^{ab}$; ++). Let assume that there are two group of elements such as $\exists$ P and Q. Now find an integer n, which $\in Z_q^{RI}$, while Q = nP, then there exist another integer which is needed to be calculated.
2) Elliptic Curve Discrete Logarithm Problem Algorithm (ECDLP): We take the ECDLP method into consideration in the computation complexity assumption because it is at the heart of current public-key encryption.
3) Decision Diffie-Hellman Problem Algorithm (DDHPA): For the considered integers | a,b, c $\in Z_q^*$ | $P_{(n)}$, where the value of n = a,b,c |, $\exists$ P = $P_{(a)}$, $P_{(b)}$, $P_{(c)}$ to decide whether c = (a $\times$ b $\times$ mod) q or not.
4) Computational Diffie-Hellman Problem Algorithm (CDHPA): For the considered integers: e = a,b $\in Z_q^*$, where P could be summarized as P = $P_{(a)}$, $P_{(b)}$ and can be computed as $P_{(ab)}$.

B. Model Implementation Step-Up

In this section, we talk about the initiation process of the processes of the proposed model. Initially, an AS practices the following actions to choose the system parameters: Initially, an AS considers the scenario of an EL_c (a,b) of the form $y^2 = x^3 + ax + b * (\text{mod } q)$, over a prime number $Z_q = (0, 1, 2, 3, 4, 5, 6 \ldots\ldots\ldots.., q-1)$ with a BP. Similarly, an AS picks an arbitrary $UV_i \in UV_n$ or $S_i \in S_n$ with random key $\in PK_{pri} \in Z_q$ to calculate PK_{pub} at point BP such as PK_{pub} = BP. PK_{pri}. Consequently, an UAV, IoT device, and AS uses MD5 hash algorithm in terms of one-way hash function e.g MH = $\{0, 1\}^*$ to exerts a stochastic number of data to generate a fixed-length of message-digest such as MH = $\{0, 1\}_{fix}^{length}$.

C. Registration and Service Provider Server Functions

In this phase, we concentrate on the operational footprints of AS and service provider how the R3ACWU scheme is accomplished and executed by an SP_S and AS for a UV_i and S_i in terms of signature, timestamps, and key matching, etc. Reference [11], signature model is used in the R3ACWU scheme to manage the PK_{pri} and PK_{pub} keys followed by timestamps and signature of legitimate devices such as UV_i and S_i in AS and SP_S. The following steps will be executed in this process:

Initially, each $UV_i \in UV_n$ and $S_i \in S_n$ share their identity (ID_{UV_i}, ID_{S_i}) and PK_{pri} with an AS through a secure communication channel for registration, where the i^{th} term represents the total number of UAVs and IoTs participating in the network, such as $i = \{1, 2, 3, 4, 5, \ldots, n\}$. Upon registration, an AS initiates dynamic secret identities for a UV_i and S_i with timestamp i.e. T.$PK_{pub} \in Z_q^*$ in coordination with VAS and SP_S. After that an AS calculate the BP timestamp for an UV_i and S_i such as $T_{UV_i} *$, $T_{S_i} *$ at BP, where the authentication parameters for an UV_i massage (MSG) is $MSG_{UV_i} = T_{UV_i} \oplus$ CSMA $\oplus$ MH.($ID_{UV_i} \parallel T_{UV_i} \parallel T_{AS}$).BP (mod. q) and $S_i = T_{S_i} \oplus$ CSMA $\oplus$ MH.($ID_{S_i} \parallel T_{S_i} \parallel T_{AS}$).BP (mod. q). Similarly, the authentication parameter of an VAS with timestamp is simplified as T_{VAS} = T.(T_{VAS}) * BP, So the message of an VAS can be managed as follow; $MSG_{VAS_i} = T_{VAS_i} \oplus$ CSMA $\oplus$ MH.($ID_{AS_i} \parallel ID_{VAS_i} \parallel$ (T_{UV_i} or T_{S_i}) $\parallel T_{AS}$).BP (mod. q)

Next, the role of SP_S is as follows: An AS sends messages, MSG_{UV_i} = (T_{UV_i} or T_{S_i}) $\oplus$ CSMA $\oplus$ MH.(ID_{UV_i} or ID_{S_i}) $\parallel$ (T_{UV_i} or T_{S_i}) $\parallel T_{AS}$).BP (mod q) and $MSG_{VAS_i} = T_{VAS_i} \oplus$ CSMA $\oplus$ MH.($ID_{AS_i} \parallel ID_{VAS_i} \parallel$ (T_{UV_i} or T_{S_i}) $\parallel T_{AS}$).BP (mod q) to an SP_S via a secure communication channel for further processing, such as policy implementation and execution. Upon reception, SP_S updates its memory with respect to policy implementation and follows these policies in the next phase to ensure the operation of the R3ACWU scheme. Algorithm 1 illustrates the collaborative steps of an AS, VAS, and SP_S.

D. Authentication and Key Matching Phase

In this phase, we evaluate the steps taken in the mutual authentication process between paired UV_i, S_i, AS, and SP_S. Upon key matching, the paired $UV_i \in UV_n$ or $S_i \in S_n$ establishes a session key to maintain information confidentiality during communication. Let's assume that an UV_i initiates a validation request with UV_j or S_j by generating a random key G_{rk} = (MH.($UV_{i_{rk}} \parallel VU_i \parallel$ (T_{VU_j} or T_{S_j})).BP) for the communicating UV_j or S_j, including a timestamp (T_{UV_j} or T_{S_j}) and T_{VAS_j}. With these attributes, an UV_i sends a validation request to UV_j or S_j in the form of a message, such as $MSG_{UV_i} = T_{UV_i} \oplus$ CSMA $\oplus$ MH.($ID_{UV_i} \parallel$ (ID_{UV_j} or ID_{S_j}) $\parallel T_{AS} \parallel T_{VAS}$).BP (mod q).

Algorithm 1 AS, VAS and SP_S Collaborative Steps

Require: Registration of an UV_i or S_i
Ensure: Policy Implementation and Execution in SP_S
Start
$VU_i \Leftarrow$ Initiates registration request with AS
AS $\Leftarrow$ checks $MSG_{(UV_i)}$ or $MSG_{(S_i)}$
AS $\Leftarrow$ Registers (ID_{UV_i} or ID_{S_i}) & PK_{pri}
$G_{rk} \Leftarrow PK_{pub}$
Next
AS Forward $\Leftarrow$ (ID_{UV_i} or ID_{S_i}) & PK_{pub}
(ID_{UV_i} or ID_{S_i}) & $PK_{pub} \Leftarrow$ VAS & SP_S
$SP_S \Leftarrow$ Store & apply RAAACWU polices
$SP_S \Leftarrow$ Broadcast (ID_{UV_i}) or ID_{S_i} & PK_{pub}
$UV_i \in UV_n$ or $S_i \in S_n \Leftarrow$ Updates their per-store
End
return Current information of AS, VAS, and SP_{RS}

Upon receiving MSG_{UV_i} at time T_{UV_i}, VAS_i and SP_S collaborate to validate the authentication request from UV_i to UV_j or S_j by matching their identities, keys, and timestamps. This validation includes checking if (T_{VU_i} - T_{VU_i}) < T_{VAS_i}. If the authentication parameters of UV_i and UV_j or S_j match, AS and SP_S generate a random secret key (R_{sec_k}) for UV_j or S_j in Z_q^* using the current timestamp (T_{VAS_i}), SP_S, VAS_{VU_j} or VAS_{S_j}, and $SP_{RS_{VU_j}}$ or $SP_{RS_{S_j}}$. Following this, the management of R_{sec_k} for UV_j is as follows:

Let's assume there is a generator, G_{rk}, in an AS that generates a message for UV_j or S_j with a random key, such as $R_{sec_{k_{UV_j}}}$ or $R_{sec_{k_{S_j}}}$, which is calculated as MH.(PK_{pub} $\|$ (T_{VU_i} or T_{S_i}) $\|$ VAS_i).BP. With this key, it prepares the message MSG_{VU_i} with authentication attributes, including MSG_{VU_i} = MH.($PK_{pub_{VU_j}}$ or $PK_{pub_{S_j}}$) $\|$ T_{VU_i} $\|$ VAS_i $\|$ ID_{UV_i} $\|$ (ID_{UV_j} or ID_{S_j}) $\|$ SP_{RS}).BP (mod.q).

Similarly, when UV_j or S_j receives the message from UV_i, it matches the necessary security parameters in its per-store register to verify the legitimacy of the requesting UV_i. Once the authentication parameters of MSG_{VU_i} match in UV_j or S_j, communication between UV_j or S_j and requesting UV_i matches in the network, then communication process begin.

In contrast, an AK initiates an authentication request with $UV_i \in UV_n$ or $S_i \in S_n$ by sending a message, MSG_{AK} = T_{AK} $\oplus$ CSMA $\oplus$ MH.(ID_{AK} $\|$ (T_{UV_j} or T_{S_j}) $\|$ AK_{pri} $\|$ T_{AS}).BP (mod. q).

The AK's message request is first received by the concerned AS. Upon reception of MSG_{AK}, an AS and VAS check the security parameters of AK, including T_{AK}, ID_{AK}, AK_{pri}, T_{AS}, etc. If the AK's message parameters do not match in the pre-store of an AS and, in particular, VAS, based on the aforementioned metrics, the designated AS and VAS deny the validation request of an AK and forward the identity of AK in the network to acknowledge its (malicious) presence to other legitimate devices.

Upon reception of the alarming message regarding an AK. All the legitimate UV_i and S_i update their per-store registers regarding the parameters of an AK such as ($T_{AK_i'}$ - $T_{AK_i^*}$) < $\triangle$T to deny its communication request in future.

Algorithm 2 UAV Authentication Steps

Require: Authentication of UAVs in the network.
Ensure: Validation of legitimate UAVs in the network
Start
$VU_i \Leftarrow$ Initiates authentication request
VU_i send Message $\Leftarrow$ UV_j or S_j
First
$MSG_{UV_i} \Leftarrow$ receives by an AS
AS & VAS $\Leftarrow$ Checks MSG_{UV_i} Security parameters
AS & VAS $\Leftarrow$ Matches MSG_{UV_i} Identities
such as T_{UV_i} $\|$ ID_{UV_i} $\|$ T_{UV_j} $\|$ ID_{UV_j} $\|$ T_{AS}
If
MSG_{UV_i} Security parameters $\Leftarrow \in$ AS & VAS
Then
AS & VAS $\Leftarrow$ Validated UV_i with UV_j or S_j
Else
MSG_{UV_i} Security parameters $\Leftarrow \notin$ AS & VAS
AS & VAS $\Leftarrow$ Denies MSG_{UV_i} request
AS & VAS $\Leftarrow$ Broadcast acknowledgment packet
End If
$UV_i \in UV_n$ or $S_i \in S_n \Leftarrow$ Updates their per-store
End
return Updated information of $UV_i \in UV_n$ or $S_i \in S_n$

E. Dynamic Device Adaptation

In this phase, we emphasize how a new UAV or IoT device can be added to an operational UAV-assisted IoT application network.

Let's assume a new UV_i or S_i wants to participate in the existing network. For this, the new UV_i or S_i generates a registration message ($MSG_{UV_{new}}^{ID}$) with an AS. The UV_i or S_i message contains information such as $MSG_{UV_{new}}$ = (UV_{new}^{ID} $\|$ CSMA $\|$ $T_{UV_{new}^{ID}}$ $\|$ AS $\|$ $UV_{new}^{pk_{pri}}$) that would be sent to an AS, herein, for generality we only assumed UV_i, the process is same for S_i as well. Upon reception of $MSG_{UV_{new}}$, the AS generates a random key for UV_{new} for new timestamp ($T_{UV_{new}^{ID}}$), where G_{rk} of $UV_{new}^{ID} \in Z_q^*$:. $\rightarrow$ $T_{UV_{new}^{ID}}$ = ($T_{UV_{new}^{ID}}$).BP and UV_{new}^{ID} = $T_{UV_{new}^{ID}}$ MH.(UV_{new}^{ID} $\|$ CSMA $\|$ $T_{UV_{new}}$ $\|$ T_{AS} $\|$ $T_{AS_{VAS}}$).BP (mod. q). Next, the registration process completes, and AS send a confirmation message to UV_{new} followed by broadcasting a message in the network, which contains information such as MSG_{AS} = UV_{new}^{ID} MH.($T_{UV_{new}^{ID}}$ $\|$ CSMA $\|$ $T_{UV_{new}}$ $\|$ $UV_{PK_{pub}}$ $\|$ $T_{AS_{VAS}}$).BP (mod. q). Afer reception of this message, all $UV_j \in UV_n$ update their per-store register to legitimately communicate with UV_{new} in the future.

F. R3ACWU Scheme Execution Steps

In this section, we explain how our proposed lightweight authentication model (R3ACWU) executes different steps in an operational network. After registration and the validation of legitimate devices, the primary concern of an authentication scheme is its complexity. To address this issue, the proposed lightweight authentication model defines a time frame T_{TF} for UV_i or S_i devices to clear and update their cache memory in

coordination with AS, VAS, and SP_{RS}. An $UV_i \in UV_n$ or $S_i \in S_n$ executes the memory clearing after defined interval of time. This time is defined as $MSG_{AS_{T_{TF}}} = T_{TF}$ MH.($UV_i \parallel$ CSMA $\parallel T_{AS_{VAS}} \parallel UV_{i_{PK_{pub}}} \parallel T_{AS_{VAS}}$).BP (mod. q). Upon receiving the message $MSG_{AS_{T_{TF}}}$, the $UV_i \in UV_n$ or $S_i \in S_n$ devices clear and update their pre-store memory by removing the information of those legitimate devices with whom they have not communicated within the T_{TF} time period. This updated memory of $UV_i \in UV_n$ or $S_i \in S_n$ devices allows them to process authentication requests quickly since there is no extraneous information in the memory of legitimate devices. This ensures the minimal complexity of the proposed model during the validation process.

V. Formal and Informal Security Analysis

In this section, we discuss both the formal and informal security analyses to examine the effectiveness of the proposed model in mitigating various attacks. These analyses (formal and informal) are presented in separate upcoming subsections.

A. Security Model for Formal Security Proof

In this section, we perform a formal security analysis of network entities including AS, SP_s, UV_i, and S_i with a focus on the authentication process. This analysis aims to demonstrate the security resilience of our scheme against an adversary (AK), which is generalized with the symbol (Ψ), within the given scenario. Each of the Authentication Server (AS), Virtual Authentication Server (VAS), and SP_i stores the secret keys and authentication parameters of legitimate devices, denoted as UV_i and S_i, and summarized as: ($UV_{ID} \parallel UV_{PK_{pub}} \parallel T_{VU_i} \parallel S_i \parallel VU_j^{ID} \parallel S_i$). Each of these entities simultaneously executes Ψ across multiple instances, which is assumed an oracle with three possible states: accept, reject, and fault message. Consequently, we summarize and assess the following three case studies as part of the formal security analysis.

Case 1: Now, let's consider an AK with the capability to query the oracle using the following steps to gain knowledge of the session keys of legitimate devices.

1. Execution: When AK conducts passive attacks on the device by initiating a query to access the legitimate authentication parameters, it follows these steps: First sends a query to the oracle to obtain the authentication parameters. Upon receiving a response from the oracle with either MSG_{UV_i} or MSG_{S_i}, AK proceeds to simulate an active attack. In the active attack scenario, AK sends a MSG_{AK} to the device for validation. The oracle's response to AK's message depends on the defined policies. If the message matches the policies, the oracle will accept and respond; otherwise, it will reject AK's query.

2. Subsequently, when AK initiates this query with UV_i and S_i to acquire their authentication parameters. In this case, these devices will be in an accept state for the MSG_{AK}, and will grant AK access to the secret keys stored within them. At this point, AK has determined the following scenarios: If MSG_{UV_i} or $S_i = 0$, AK has successfully obtained the authentication key of a legitimate device through this query. If MSG_{UV_i} or $S_i = 1$, AK has acquired communication information of a legitimate device via this query. If MSG_{UV_i} or $S_i = 2$, AK has obtained the identity of a legitimate device through this query. Should all the parameters above match, AK will be capable of compromising the device.

Device-Test: In this segment, AK examines the stored security attributes of UV_i^{PK} and S_i^{PK} (or $MSG_{VU_i}^{PK}$, $MSG_{S_i}^{PK}$) to extract the authentication parameters. If a bit (b) attribute yields an output of b = 1, AK extracts the authentic session key. Conversely, if b = 0, then a RI string is generated by legal devices, which is basically corrupted.

Case 2: (Strong Security Freshness): An instance of UV_i^{PK} and S_i^{PK} (or $MSG_{VU_i}^{PK}$, $MSG_{S_i}^{PK}$) maintains strong security freshness until one of the following situations occurs:

Corrupt Message: When an AK queries MSG_{VU_i}, S_i with values (0, 1, 2), indicating a corrupt query for MSG_{VU_i}, S_i before the test concludes.

Case 3: (Semantic Safety and Security): This refers to the ability of an AK to compromise a legitimate device with the specific attributes of b used in the test case. Given that, the operation of AK is generalized as $AK_{\Psi}^{MSG} = (0,1,2) \times (P_{ro}$ $[b = \dot{b}]) \times \Psi$, where security is preserved of the original key and message. However, the size of AK_{Ψ}^{MSG} exceeds that of UV_i^{PK} and S_i^{PK} (or $MSG_{VU_i}^{PK}$, $MSG_{S_i}^{PK}$) key lengths, denoted by $q_{se}(\frac{1}{Key_l}, \frac{1}{2^b})$, where q_{se} and Key_l represent the query bound and length of a secret key, respectively.

1) Formal Security Proof:

Lemma: In an elliptic curve EL_c, the notation Key_l signifies the length of a secure key ($q_s \cdot \Psi$) distributed within the proposed model. An AK is capable of compromising Ψ within an upper bound time (U_{Δ_t}), as follows:

$$\begin{aligned} AK_{\Psi}^{key_l} \leq\ & 2q_{MH}((q_{se}+q_{ex})^2+1) \times (t_{EL_c} + t_{MSG}(q_{ex} + q_{se})) \\ & + 2max\{q_{se}\frac{1}{key_l}, \frac{1}{2^{S_i}})\} + \frac{2q_{se} + q_{MH}^2 + q_{VU_{id}}^2}{2^{key_l}} \\ & + (\frac{q_{ex} + q_{se}}{P_{ro}}) \end{aligned} \tag{1}$$

In equation 1, we symbolize the execution time of hash queries as q_{MH}, whereas the send query and time cost of a message are denoted by q_{se} and t_{MSG}, respectively, within a cyclic group (CG).

Proof: Under the specified conditions within CG, we assume the existence of CG_i, where CG_i $(0 \leq i \leq 4)$ to evaluate the security parameters. Let $P_{ro}[UV_i, S_i]$ represent the probability of an AK correctly guessing the value of b for the legitimate device during the test query. The time difference between $P_{ro}[UV_i, S_i]$ and $P_{ro}([UV_i, S_i] - 1)$ is denoted as (Δ_t^i). For CG_0: When an oracle operates as a protocol within CG_0 according to the considered scenario, we can simplify an AK's attack as follows:

$$\begin{aligned} & AK_{\Psi}^{PK} \\ & = 2(P_{ro}[UV_0, S_0] - 1) - 2(P_{ro}[UV_0, S_0] \\ & \quad - 2(P_{ro}[UV_4, S_4]) = 2(P_{ro}[UV_4, S_4] - 1) + 2 \times \sum_{i=1}^{4} \Delta_t^i \end{aligned} \tag{2}$$

For CG_1: Within CG_1, a specific device maintains a hashing list. When an AK attempts to compromise the hash function, it initiates a query using a string (S_{trg}) with MH. Upon receiving this query, a device verifies the tuple MH[(S_{trg}),S_{trg}] within its security table. If a match is found, the device responds with MH(S_{trg}). Otherwise, it stores the received string in its updated list and responds to the AK's request with a random MH[(S_{trg}),S_{trg}]. Consequently, an AK would find it highly challenging to distinguish between CG_0 and CG_1.

$$\Delta_t^1 = (P_{ro}[UV_1, S_1] - P_{ro}[UV_0, S_0]) \tag{3}$$

an AK simulates fake attributes of a device, but the attributes of the AK, such as ($UV_{ID}, S_{ID}, VU_{PK}, T_{UV_i}, \leq \frac{2q_{se}+q_{MH}^2+q_{V_{Uid}}^2}{2^{key_l}} + (\frac{q_{ex}+q_{se}}{P_{ro}})$, etc.), do not meet the previously mentioned condition. As a result, an AK cannot distinguish between CG_1 and CG_2, which leads to the undermentioned scenario that:

$$\Delta_t^2 = (P_{ro}[UV_2, S_2] - P_{ro}[UV_1, S_1]) \leq \frac{q_{se}}{2P_{ro}} \tag{4}$$

For CG_3: In this part, an AK replicates the conditions of CG_2 and successfully obtains two attributes of the devices such as (UV_{ID}, S_{ID} & UV_{pk}, S_{pk}). Subsequently, suppose the AK proceeds with a corrupt query, denoted as corrupt(UV_{ID}, S_{ID} & UV_{pk}, S_{pk}). At this point, the Double Decisional Hard Problem Assumption (DDHPA) of a device is triggered, and it verifies the following conditions:

1. Initially, the AK executes a q_{se} with corrupt(UV_i, S_i) to guess the PK_{pri} of legitimate devices with a correct choice, which is denoted as $\Rightarrow \frac{q_{se}}{Key_l}$.

2. Next, the AK proceeds with another corrupt(UV_i, S_i) and selects one of the following two cases to crack the (UV_{ID}, S_{ID}), although these guesses do not exist:

a). The AK sends q_{se} queries to the legitimate devices with guessed ID, each with a probability of $\frac{q_{se}}{2^{UV_{ID}}}$ OR $\frac{q_{se}}{2^{S_{ID}}}$.

b). Alternatively, the AK provides its own ID with a false probability (ξ). In these cases, the maximum (P_{ro}) is determined as $q_{se}.max(\frac{1}{key_l}, \frac{1}{2^{UV_{ID}}}, \xi)$.

To extract the precise authentication attributes, an AK needs to employ the ECDHPA, DDHPA, and CDHPA. By following these steps, an AK can potentially obtain the following:

$$\Delta_t^3 = (P_{ro}[UV_3, S_3] - P_{ro}[UV_2, S_2]) \leq q_{se}.max(\frac{1}{key^l}, \frac{1}{2^{UV_{ID}}}, \xi) + q_{MH}.AK_{ID}^{DDHPA}(t_{EL_c} + T_{MSG}(q_{ex} + q_{se})) \tag{5}$$

Following cases 1 and 2, a corrupted device is required to perform an oracle test to verify an authentication request, which may be CG_4 in order to recognize previous simulations. Consequently, CG_3 can be determined as q_{MH} with probabilities x and y for a single session, specifically $\frac{1}{(q_{se}+q_{ex})^2}$. Which is summarized as below:

$$\Delta_t^4 = (P_{ro}[UV_4, S_4] - P_{ro}[UV_3, S_3]) \leq (q_{se} + q_{ex})^2 \times AK_{ID}^{DDHPA} \times (t_{EL_c} + T_{MSG}(q_{ex} \tag{6}$$

In the final stage, we have $P_{ro}[UV_4, S_4] = \frac{1}{2}$, which affirms that an AK cannot meet the CG conditions. Therefore, this lemma demonstrates that an AK is incapable of compromising a legitimate device.

B. Informal Security Analysis

In this section, we conduct the informal security analysis to verify the resilience of the proposed model against the threat model attacks. The objective is to showcase its superiority over existing authentication schemes. To maintain generality in the model, we have simplified the authentication parameters in the simulation environment, as follows:

1) **Private-Key Generation Step:** The $UV_i \in UV_{n-1}$ executes the encryption algorithm to generate the corresponding identity keys, denoted as $UV_{ID} \rightarrow PK_{pri}^{ID}$. The key generation (G) algorithm, upon receiving a request from UV_i, performs the following steps:
Upon receiving the UAV request, G computes a random PK_{pri} such that $G_{rk} = UV_{pri} \in Z_{uq}^*$.
$UV_{pri} = (G_{rk_i}^{UV_{ID}}) \cdot MH = (UV_i \rightarrow UV_{ID}) \cdot MH$.
Subsequently, UV_i sends $(UV_i \rightarrow UV_{ID}) \cdot MH$ to an AS for further processing.
2) **Public-Key Generation Step:** Upon receiving the corresponding identities of UV_{ID} or S_{ID}, then the generator algorithm follows these steps to generate a public key of these devices:
The Key generation (G) within AS performs the following steps to create a PK_{pub} for a UAV private key. G_{rk} is executed for UV_{pri} to generate $UV_{pub} \in Z_{uq}^*$ such that $UV_{pub} = (G_{rk_i}^{UV_{ID}} \rightarrow UV_{ID} \rightarrow UV_{pri}) \cdot MH = (AS_{ID} \rightarrow UV_{ID} \rightarrow UV_{pri} \rightarrow VAS_{ID} \rightarrow SP_{RS}) \cdot MH$.

Upon completing this step, an AS disseminates the UV_{pri} in the network, and the connected devices update their prestore register. This process enhances secure authentication and communication within an open communication environment. Consequently, a UV_i within the network initiates a successful authentication request when it satisfies the comparison parameters of $(AS_{ID} \rightarrow UV_{ID} \rightarrow UV_{pri} \rightarrow VAS_{ID} \rightarrow SP_{RS})$.MH.

C. Forgery Attack Utilizing Tampered/Replaced Public Key

Let's assume a scenario where a malicious device with the identity AK_{ID} and PK_{pri} aims to impersonate a legitimate $UV_i \in UV_n$ by initiating an authentication request with it. Initially, an AK intercepts the legitimate device message to extract valid authentication parameters, denoted as $MSG_{UV_i} \in Z_{pub}^*$, and UV_{ID}^i. Subsequently, the AK_i forges the UV_{pub}^i and UV_{ID}^i by replacing it's authentication parameters. The steps involved in this process are summarized as follows:

1) **Steps:** Let an AK intercept the actual message of a legitimate device during the communication process between UV_i and UV_j by launching an eavesdropping attack. The intercepted information is denoted as $MSG_{VU_i} = UV_{ID} \rightarrow UV_{PK_{pub}} \rightarrow T_{VU_i} \rightarrow UV_j^{ID})$. Utilizing the legitimate device authentication parameters, AK generates a similar validation signature or message with

several irrelevant pieces of information, such as VAS_{ID}, $AS_{T_{VAS}}$, and T_{UV_j}, etc., to compromise the security of the device.

2) $MSG_{AK} \in Z^*_{VU_i}$ $(PK_{pub} \parallel UV_i)$, where $MSG^*_{AK_i} \neq MSG_{AK_i}$
3) Next, the AK sets the authentication message as $MSG_{AK} = MH \cdot (PK^{VU_i}_{pub} \oplus UV^{ID}_j \oplus CSMA \rightarrow T_{AK} \oplus T_{UV_i} \oplus AK_{pri})$ mod q and sends it to UV_j through the relevant AS.
4) **Verification Phase:** Upon receiving $MSG_{AK} = MH \cdot (PK^{VU_j}_{pub} \oplus UV^{ID}_j \oplus CSMA \rightarrow T_{AK} \oplus AK_{pri})$ mod q, the AS performs authentication parameter verification by comparing it with registered UAV IDs, private keys, timestamps, VAS timestamps, requesting AK authentication parameters. During the verification process, it becomes evident that the AK private key, timestamp, registration BP, as well as VU_j timestamp and T_{AVS} do not meet the validation requirements. Consequently, the AS rejects the authentication request for AK's forged public key replacement with legal UAVs and broadcasts an alarm message containing AK_{ID}, BP, and timestamp to inform the legitimate device in the network about the intrusion.

D. Forgery Attack Based on an UAV Previous Public Key

Let's assume that an AK captures and intercepts a previously communicated message from a legitimate device that containing the authentication parameters. This message is denoted as $MSG_{VU_i} = UV_{ID} \rightarrow UV_{PK_{pub}} \rightarrow T_{VU_i} \rightarrow UV^{ID}_j$, where $MSG_{UV_i} \in Z^*_{pub}$ and UV^i_{ID}. Subsequently, AK extracts the UV_{pri} and UV_{ID} of the legal UAV from this intercepted message to initiate an authentication request within the network. The step-by-step operation is summarized as follows:

1) During an eavesdropping attack on an open communication channel, an AK intercepts a valid public key from a UAV's past messages, denoted as $MSG_{UV_i} \in Z^*_{pri}$ and UV_{ID}. It is worth noting that while this key and message were used by a UAV for authentication in the past, they are not currently in use by an UV_i for communication with UV_j. However, based on the historical assumptions regarding UV_i's private key, the authentication parameters within MSG_{UV_i} have been successfully authenticated by relevant authorities, including AS, AS_{VAS}, and UV_j.
2) **Next,** An AK uses a private key, simplified as $AK_{pri} \in Z_{VU_i}$ to initiate an authentication request with UV_j in the network.
3) **Forged Public key verification phase:** Upon receiving the message MSG_{AK} = MH.$(PK^{VU_j}_{pub} \oplus UV^{ID}_j \oplus$ CSMA $\parallel T_{AK} \oplus AK_{pri})$.BP mod q, which contains the forged public key information of UV_j. The authentication server (AS) performs the following steps: First, the it matches the validation parameters provided by AK with the registered UAV IDs, private keys, timestamps, VAS timestamps, requesting AK private keys, and the updated information in its pre-stored table.

During the validation process, it's discovered that the parameters AK_{ID}, AK_{pri}, T_{AK}, AK_{pub}, T_{AS}, and T_{VAS} are not present in the AS's pre-stored table. This is because the timestamp associated with this private key has already expired in the AS's records. In the context of the R3ACWU time frame (T_{TF}), the captured UAV private key had not engaged in any communication with VU_j. Therefore, during the update period, it was removed from the pre-stored table of the AS. As a result, the AS successfully rejects the validation request from AK, who attempted to use a forged key of a legitimate device in the operational network.

E. Forgery Attacks Based on UAV ID

In this scenario, an AK is aware of the authentication parameters (forged identity) of a legitimate device communicating in the network. By employing the forged UAV ID and key, AK intends to establish communication with UV_j. The steps involved in this process are summarized as follows:

1) An AK_i initiates an eavesdropping attack on the open communication channel to obtain the key and ID of a legitimate device. Through in this attack, AK captures the message of UV, which contains information such as $MSG_{VU_i} = UV_{ID} \parallel UV_{PK_{pub}} \parallel T_{VU_i} \parallel VU^{ID}_j$). It should be noted that $MSG_{UV_i} \in Z_{pub}$ and $UV^i_{ID} = MSG_{UV_i} \in Z_{pub}$ and $(UV_{ID} \parallel UV_{pub}$.
2) Upon capturing MSG_{VU_i} an AK generates the same authentication message with the help of key generator $\rightarrow$: . $AK^{ID}_{pub} \in (Z^*_{VU_i} \parallel (PK_{pub}) \parallel UV_{ID})$.
3) **Next,** The AK initiates the authenticating requesting with UV_j by sending a message $\rightarrow MSG_{AK} = AK_{ID} \parallel AK_{PK_{pub}} \parallel T_{AK} \parallel VU^{ID}_j$), in the network.
4) Upon receiving a message from AK such as MSG_{AK}, the AS verifies the security parameters by comparing them with its pre-stored data. Specifically, the AS validates the parameters $AK_{PK_{pub\&pri}}$, AK_{ID}, T_{AK}, T_{AS}, T_{VAS}, B.P, VU_j, SP_{RS}, $T_{SP_{RS}}$, SP_{ID}, and T_{TF}. During the parameter matching process, it is observed that the authentication parameters of AK, such as T_{AS}, T_{VAS}, B.P, SP_{RS}, $T_{SP_{RS}}$, SP_{ID}, and T_{TF}, do not match with the information stored in the AS's pre-store database. Consequently, the attempted forged ID attack is effectively thwarted in the proposed model.

F. Comparative Analysis of Forged Attacks

In this section, we evaluated the proposed lightweight authentication scheme with the existing state-of-the-art schemes by launching various attacks during the simulation environment based on the aforestated attacks (forgery attacks) to verify its effectiveness. During comparative analysis, we take into account Tan et al. [13], Sun et al. [17], and Paonessa et al. [18], schemes to prove the reliability of our model. Table II and Fig 2, 3, illustrates the result statistics of the proposed scheme in presence of the competitors schemes.

TABLE II
INFORMAL SECURITY COMPARATIVE ANALYSIS

Scheme Name	Forgery attacks utilizing tampered/replaced Public Key	Forgery attacks based on previous UAVs Public Key	Forgery attacks based on UAV ID
Tan et al. [13]	No	No	No
Sun et al. [17]	No	Yes	No
Paonessa et al. [18]	NO	No	No
Lightweight-R3ACWU	Yes	Yes	Yes

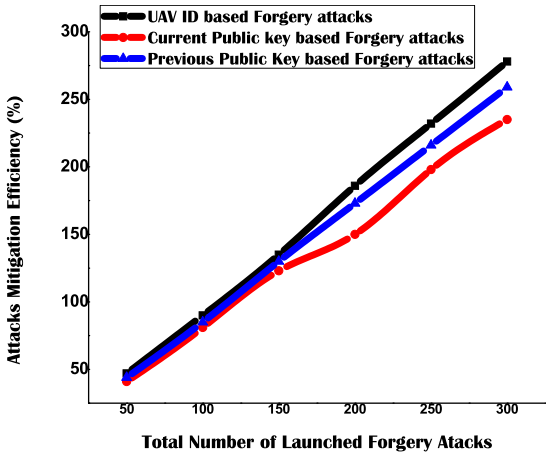

Fig. 2. Different kind of forgery attacks detection Efficiency.

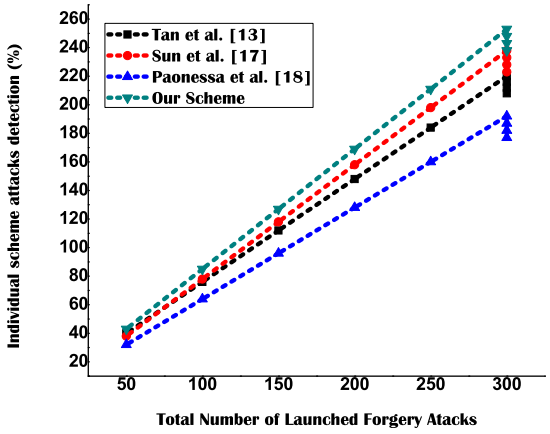

Fig. 3. Comparative schemes attacks detection efficiency analysis.

G. Computation Complexity

Authentication complexity is a crucial factor in assessing the dependability of any cryptographic or authentication model, especially when comparing it with the state-of-the-art schemes. In our analysis, we determined that the authentication complexity of our system is less compared to other schemes. For Proof: we consider T_{BLP} for bilinear pairing operations, T_{Erp} and T_{Drp} for encryption and decryption operations, and T_{pub} and T_{pri} for key initiation. The time for point of operation, transmission, and reception are symbolized with the notations, T_{BP}, T_{Tx}, and T_{Rx}, respectively. Additionally, ΔT_{AS}, $\Delta T_{SP_{RS}}$, and ΔT_{UV_j} is used to represented the time

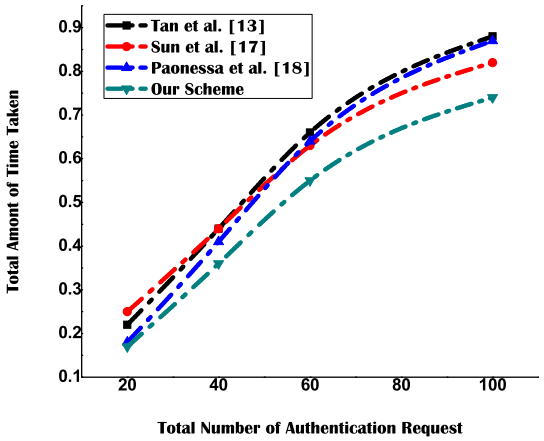

Fig. 4. Computation cost Comparative Results Analysis.

for AS, SP_{RS}, and UV_j, while the $\oplus$ operator is used for modular addition operations. To effectively assess computation complexity, we applied these notations to each comparative scheme. The findings of the computation complexity analysis are presented in Table III, and Fig 4, provides a graphical representation of the comparative results. For computation, we estimated execution times, considering cryptographic operations such as T_{Erp} and T_{Drp} as approximately equivalent to MH_{XOR}. Similarly, T_{pub} and T_{pri}, as well as (ΔT_{AS}, $\Delta T_{SP_{RS}}$, and ΔT_{UV_j}), were approximated as $T_{k(XOR)}$. Bilinear pairing operations (T_{BLP}) and point operations (T_{BP}) were estimated as approximately equivalent to KM_{XOR}.

H. Communication Cost

In this part, we look at the communication metrics of our lightweight key authentication strategy to see how reliable it is in the presence of Tan et al. [13], Sun et al. [17], and Paonessa et al. [18] approaches. For evaluation, we have assumed multiple factors along with authentication key and message length in terms of bit sizes followed by a timestamp, UAV Identity, bilinear pairing, key matching/handshaking, and MD hash fixed length message digest.

In Lightweight-R3ACWU, the message MSG_i length size is managed for the aforementioned scenario as $(512 + 32) = 544$ bits $(512 + 128 + 32) = 672$ bits, and $(512 + 128 + 128 + 32) = 800$ bits, which leads to an cumulative communication cost as $(544 + 672 + 800) = 2016$, wherein MSG_i, the i^{th} term represent the total number of messages. Similarly, we have checked the communication cost of the Tan et al. [13] scheme, which was found about $(512 + 512 + 32) = 1076$ for an individual message payload with a total cost of 5380 bits for 5 messages, whereas the message payload for Sun et al. [17], and Paonessa et al. [18], was 694 bits and 672, consequently.

In Table IV, we illustrate the statistical results for communication cost of our Lightweight-R3ACWU in presence of rival schemes.

I. Contention, Congestion and Packet Lost Ratio Comparative Analysis

In this section, we appraised the offered lightweight key authentication prototype for contention and congestion,

TABLE III
COMPUTATION OVERHEAD COMPARATIVE STATISTICAL ANALYSIS

Scheme Name	Client side	Next Hop Count/Requesting UAV	Server and Service Provider Side	Total Cost
Tan et al. [13]	$2MH_{XOR} + 4\,T_{k(XOR)} + 2KM_{XOR}$	$2MH_{XOR} + 4\,T_{k(XOR)} + 2KM_{XOR}$	$4MH_{XOR} + 4\,T_{k(XOR)} + 4KM_{XOR}$	$8MH_{XOR} + 12\,T_{k(XOR)} + 8KM_{XOR}$
Sun et al. [17]	$2MH_{XOR} + 3\,T_{k(XOR)} + 3KM_{XOR}$	$3MH_{XOR} + 3\,T_{k(XOR)} + 4KM_{XOR}$	$4MH_{XOR} + 3\,T_{k(XOR)} + 4KM_{XOR}$	$9MH_{XOR} + 9\,T_{k(XOR)} + 11KM_{XOR}$
Paonessa et al. [18]	$2MH_{XOR} + 2\,T_{k(XOR)} + 4KM_{XOR}$	$2MH_{XOR} + 3\,T_{k(XOR)} + 4KM_{XOR}$	$2MH_{XOR} + 3\,T_{k(XOR)} + 4KM_{XOR}$	$6MH_{XOR} + 8\,T_{k(XOR)} + 12KM_{XOR}$
Our Scheme	$1MH_{XOR} + 2\,T_{k(XOR)} + 1KM_{XOR}$	$2MH_{XOR} + 2\,T_{k(XOR)} + 2KM_{XOR}$	$2MH_{XOR} + 2\,T_{k(XOR)} + 3KM_{XOR}$	$5MH_{XOR} + 6\,T_{k(XOR)} + 6KM_{XOR}$

TABLE IV
LIGHTWEIGHT-RAAACWU COMMUNICATION COST COMPARATIVE ANALYSIS

Scheme Name	Total number of messages	Total Communication cost (in bits)
Tan et al. [13]	5	5380
Sun et al. [17]	4	3176
Paonessa et al. [18]	5	3160
Our scheme	3	2016

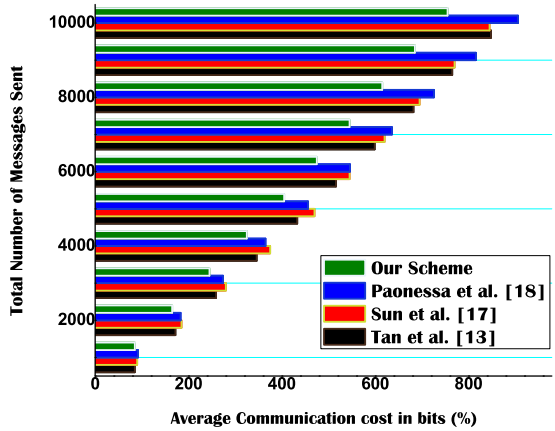

Fig. 5. Contention and congestion results statistics.

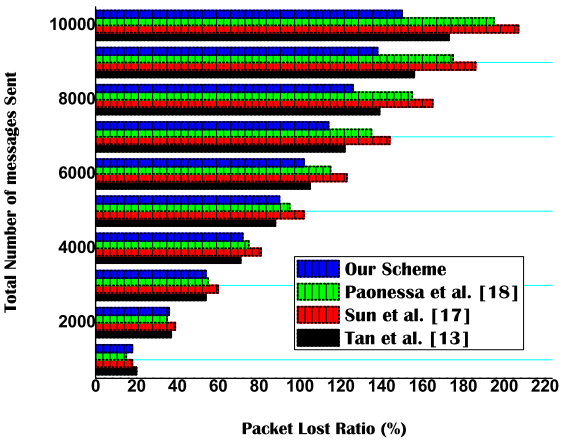

Fig. 6. Packet lost ratio results statistics.

because the efficacy of a solid authentication model is greatly resilient on these two metrics. During analysis, we check the network traffic, while initiating validation requests among legal devices. In the next step, we increased the validation requests by increasing the number of devices in the network. Throughout this phase, we have witnessed that the inbound authentication requests from legitimate devices were entertained by an AS consistently, which ensured the efficacy of this scheme.

After that, we have applied the T_{TF} during the operational network to check, how this property would be helpful in delay-sensitive authentication among legitimate devices. The validation statistics noted for paring devices in the AS and VAS showed consistent results, while the competitors' authentication model lacks at some stage to maintain such a delay-sensitive authentication in large networks. The comparative analysis was made on the basis of the total authentication request, successful authentication, and not respond requests. Due to the enormous number of authentication requests, the rival schemes somehow were unable to accommodate them, as a result, they create contention and congestion as shown in Fig 5. In the consequent step, we have verified this quality of our proposed model by finding the packet lost ratio through simulation. Likewise authentication, we have increased network traffic with the passage of time to check the successful packet delivery ratio. Fig 6, demonstrates the results statistics for this part.

VI. CONCLUSION

In this research, we proposed a lightweight R3ACWU key exchange authentication scheme for UAV-assisted IoT applications to handle the authentication challenges in these networks. In this model, UAVs, IoTs, and other network entities initially register with an Authentication Server (AS), which generates public keys for each registered device. Following this, the AS assigns and directs each registered device to a Virtual Authentication Server (VAS) and a service provider server. These servers are responsible for managing the validation and service allocation process in a secure setting. Furthermore, we enabled the legitimate devices to periodically wash and update their cache memory to enhance their processing and computational efficiency followed by reduced authentication complexity in an operational environment. Given that, the validation of legitimate devices was ensured in a delay-sensitive environment with minimal communication and computation costs. Considering that, the proposed Lightweight-R3ACWU authentication scheme showed significant results for communication metrics, due to set functionalities, because most of the validation requests are entertained during the operational network. In addition, the ordinary communication of the

network also showed remarkable results, which are shown in the comparative analysis section in the presence of rival schemes.

REFERENCES

[1] M. Bilal and S. Pack, "Secure distribution of protected content in information-centric networking," *IEEE Syst. J.*, vol. 14, no. 2, pp. 1921–1932, Jun. 2020.

[2] L. Nkenyereye, S. M. R. Islam, M. Bilal, M. Abdullah-Al-Wadud, A. Alamri, and A. Nayyar, "Secure crowd-sensing protocol for fog-based vehicular cloud," *Future Gener. Comput. Syst.*, vol. 120, pp. 61–75, Jul. 2021.

[3] X. Wu, X. Xu, and M. Bilal, "Toward privacy protection composition framework on Internet of Vehicles," *IEEE Consum. Electron. Mag.*, vol. 11, no. 6, pp. 32–38, Nov. 2022.

[4] I. Butun, M. Erol-Kantarci, B. Kantarci, and H. Song, "Cloud-centric multi-level authentication as a service for secure public safety device networks," *IEEE Commun. Mag.*, vol. 54, no. 4, pp. 47–53, Apr. 2016.

[5] M. Tanveer, A. H. Zahid, M. Ahmad, A. Baz, and H. Alhakami, "LAKE-IoD: Lightweight authenticated key exchange protocol for the Internet of Drone Environment," *IEEE Access*, vol. 8, pp. 155645–155659, 2020.

[6] H. Zhang, S. Kumari, M. S. Obaidat, and F. S. Wei, "Gateway-oriented two-server password authenticated key exchange protocol for unmanned aerial vehicles in mobile edge computing," *IET Commun.*, vol. 14, no. 15, pp. 2427–2433, Sep. 2020.

[7] B. D. Deebak and F. Al-Turjman, "A smart lightweight privacy preservation scheme for IoT-based UAV communication systems," *Comput. Commun.*, vol. 162, pp. 102–117, Oct. 2020.

[8] A. Islam and S. Young Shin, "A blockchain-based secure healthcare scheme with the assistance of unmanned aerial vehicle in Internet of Things," *Comput. Electr. Eng.*, vol. 84, Jun. 2020, Art. no. 106627.

[9] M. Adil, H. Song, S. Mastorakis, H. Abulkasim, A. Farouk, and Z. Jin, "UAV-assisted IoT applications, cybersecurity threats, AI-enabled solutions, open challenges with future research directions," *IEEE Trans. Intell. Vehicles*, to be published.

[10] M. Adil, M. A. Jan, Y. Liu, H. Abulkasim, A. Farouk, and H. Song, "A systematic survey: Security threats to UAV-aided IoT applications, taxonomy, current challenges and requirements with future research directions," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 1437–1455, Feb. 2023.

[11] N. Wang, P. Wang, A. Alipour-Fanid, L. Jiao, and K. Zeng, "Physical-layer security of 5G wireless networks for IoT: Challenges and opportunities," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8169–8181, Oct. 2019.

[12] G. Raja, S. Anbalagan, A. Ganapathisubramaniyan, M. S. Selvakumar, A. K. Bashir, and S. Mumtaz, "Efficient and secured swarm pattern multi-UAV communication," *IEEE Trans. Veh. Technol.*, vol. 70, no. 7, pp. 7050–7058, Jul. 2021.

[13] Y. Tan, J. Liu, and N. Kato, "Blockchain-based key management for heterogeneous flying ad hoc network," *IEEE Trans. Ind. Informat.*, vol. 17, no. 11, pp. 7629–7638, Nov. 2021.

[14] M. Yahuza, M. Y. I. Idris, A. W. A. Wahab, T. Nandy, I. B. Ahmedy, and R. Ramli, "An edge assisted secure lightweight authentication technique for safe communication on the Internet of Drones network," *IEEE Access*, vol. 9, pp. 31420–31440, 2021.

[15] B. Tian, J. Cui, H. Lu, L. Liu, and Q. Zong, "Attitude control of UAVs based on event-triggered supertwisting algorithm," *IEEE Trans. Ind. Informat.*, vol. 17, no. 2, pp. 1029–1038, Feb. 2021.

[16] T. Alladi, G. Bansal, V. Chamola, and M. Guizani, "SecAuthUAV: A novel authentication scheme for UAV-ground station and UAV-UAV communication," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 15068–15077, Dec. 2020.

[17] J. Sun et al., "A data authentication scheme for UAV ad hoc network communication," *J. Supercomput.*, vol. 76, no. 6, pp. 4041–4056, Jun. 2020.

[18] F. Paonessa et al., "Design and verification of a Q-band test source for UAV-based radiation pattern measurements," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 12, pp. 9366–9370, Dec. 2020.

[19] Y. Guan, L. Zhao, J. Hu, N. Lin, and M. F. Alhamid, "Softwarized industrial deterministic networking based on unmanned aerial vehicles," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5635–5644, Aug. 2021.

[20] T. Limbasiya and D. Das, "IoVCom: Reliable comprehensive communication system for Internet of Vehicles," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 6, pp. 2752–2766, Nov. 2021.

[21] A. Gulati, G. S. Aujla, R. Chaudhary, N. Kumar, M. S. Obaidat, and A. Benslimane, "DiLSe: Lattice-based secure and dependable data dissemination scheme for social Internet of Vehicles," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 6, pp. 2520–2534, Nov. 2021.

[22] L. Wei, J. Cui, H. Zhong, Y. Xu, and L. Liu, "Proven secure tree-based authenticated key agreement for securing V2V and V2I communications in VANETs," *IEEE Trans. Mobile Comput.*, vol. 21, no. 9, pp. 3280–3297, Sep. 2022.

[23] J. Cui, J. Yu, H. Zhong, L. Wei, and L. Liu, "Chaotic map-based authentication scheme using physical unclonable function for Internet of Autonomous Vehicle," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 3, pp. 3167–3181, Mar. 2023.

[24] A. Fotouhi et al., "Survey on UAV cellular communications: Practical aspects, standardization advancements, regulation, and security challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3417–3442, 4th Quart., 2019.

[25] W. Li, H. Cheng, P. Wang, and K. Liang, "Practical threshold multi-factor authentication," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 3573–3588, 2021.

[26] W. Wang, Z. Sun, S. Piao, B. Zhu, and K. Ren, "Wireless physical-layer identification: Modeling and validation," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 9, pp. 2091–2106, Sep. 2016.

[27] Á. Vázquez-Castro and M. Hayashi, "Physical layer security for RF satellite channels in the finite-length regime," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 4, pp. 981–993, Apr. 2019.

[28] R. Altawy and A. M. Youssef, "Security, privacy, and safety aspects of civilian drones: A survey," *ACM Trans. Cyber-Phys. Syst.*, vol. 1, no. 2, pp. 1–25, Apr. 2017.

Muhammad Adil (Member, IEEE) received the B.S. and M.S. degrees in computer science from the Virtual University of Lahore, Pakistan, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree with the Department of Information Systems, University of Maryland, Baltimore County (UMBC), USA. He received the Chair's Fellowship from the department in 2022. He has CCNA and CCNP certifications. His research interests include networking, cybersecurity, cyber-physical systems (CPS), unnamed aerial vehicles (UAVs), the Internet of Things (IoT), and wireless sensor networks (WSN). He has many publications in prestigious journals, such as IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS OF INTELLIGENT TRANSPORTATION, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, IEEE SENSOR JOURNAL, *IEEE Network* magazine, *IEEE Micro* magazine, *IEEE Consumer Elect Magazine*, *ACM Transactions on Sensor Networks*, *Computer Networks* (Elsevier), and *Sustainable Cities and Societies*. In addition, he is a member of the IEEE Computer Society, IEEE Industrial Electronics, IEEE Cybersecurity, IEEE Young professionals, and London Journal Press Club-U.K., as an Honorary Member. He is reviewing for prestigious journals, such as IEEE INTERNET OF THINGS JOURNAL, IEEE SENSORS JOURNAL, IEEE SYSTEMS JOURNAL, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, IEEE WIRELESS COMMUNICATIONS LETTERS, *IEEE Communication Magazine*, *IET Communication*, *Computer Networks* (Elsevier) journals, and *Telecommunication System*.

Hussein Abulkasim (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in computer science from South Valley University, in 2004, 2012, and 2016, respectively. He was a Lecturer with the College of Computer Sciences and Information Systems, Jazan University, from 2013 to 2014. He was a Lecturer with the Department of Mathematics and Computer Science, Assiut University, from 2014 to 2017. From 2017 to 2019, he was an Assistant Professor with the Department of Mathematics and Computer Science, New Valley University. He is currently a Post-Doctoral Research Fellow with the Cybersecurity Research Laboratory, Ryerson University, Canada. His current research interests include quantum cryptography, quantum computation and communication, blockchain technology, and the IoT security.

Ahmed Farouk (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees from Mansoura University, Egypt. He is currently an Assistant Professor, before that he was a Post-Doctoral Research Fellow with Wilfrid Laurier University and Ryerson University, Canada. He is one of the Top 20 technical co-founders of the Quantum Machine Learning Program by Creative Destruction Laboratory, University of Toronto. Furthermore, he is selected as Top 25 of Innovate TO 150 Canada to showcase the best of Toronto's next generation of changemakers, innovators, and entrepreneurs. He is exceptionally well known for his seminal contributions to theories of quantum information, communication, and cryptography. He has published 62 articles in reputed and high impact journals, such as *Nature Scientific Reports* and *Physical Review A*. The exceptional quality of his research is recognized nationally and internationally. He selected by the scientific review panel of the Council for the Lindau Nobel Laureate Meetings to participate in the 70th Lindau Nobel Laureate Meeting. His volunteering work is apparent since he appointed as the Chair of the IEEE Computer Chapter for the Waterloo-Kitchener Area and an Editorial Board Member for many reputed journals, such as *Nature Scientific Reports*, *IET Quantum Communication*, and IEEE ACCESS. Also, he selected for IEEE and IET Young Professional Ambassador and as a moderator for the new IEEE TechRxiv. Recently, he appointed as an Associate Editor of the IEEE Canadian Review (ICR).

Houbing Song (Fellow, IEEE) received the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, in August 2012. He is currently a Tenured Associate Professor in AI and the Director of the Security and Optimization for Networked Globe Laboratory (SONG Lab, www.SONGLab.us), University of Maryland, Baltimore County (UMBC), Baltimore, MD, USA. Prior to joining UMBC, he was a Tenured Associate Professor in electrical engineering and computer science with Embry-Riddle Aeronautical University, Daytona Beach, FL, USA. His laboratory graduates work in a variety of companies and universities. Those seeking academic positions have been hired as tenure-track assistant professors at U.S. universities, such as Auburn University, Bowling Green State University, and The University of Tennessee. He has served as an Associate Technical Editor for *IEEE Communications Magazine* (since 2017), an Associate Editor for IEEE INTERNET OF THINGS JOURNAL (since 2020), IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS (since 2021), and IEEE JOURNAL ON MINIATURIZATION FOR AIR AND SPACE SYSTEMS (since 2020), and a Guest Editor for IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE SENSORS JOURNAL, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, and *IEEE Network* magazine. He is an editor of eight books. He is the author of more than 100 articles and the inventor of two patents (U.S. & amp; WO). His research interests include cyber-physical systems/the Internet of Things, cybersecurity and privacy, AI/machine learning/big data analytics, edge computing, unmanned aircraft systems, connected vehicles, smart and connected health, and wireless communications and networking. His research has been sponsored by federal agencies (including National Science Foundation, U.S. Department of Transportation, Federal Aviation Administration, Air Force Office of Scientific Research, U.S. Department of Defense, and Air Force Research Laboratory) and industry. His research has been featured by popular news media outlets, including IEEE GlobalSpec & 39;s Engineering360, Association for Uncrewed Vehicle Systems International (AUVSI), *Security Magazine*, *CXOTech Magazine*, Fox News, U.S. News & amp; World Report, The Washington Times, New Atlas, Battle Space, and Defense Daily.