Enabling Second Factor Authentication for Drones in 5G using Network Slicing

Mai A. Abdel-Malek*, Kemal Akkaya*, Arupjyoti Bhuyan†, Mumin Cebe‡, and Ahmed S. Ibrahim*
*Dept. of Electrical and Computer Engineering, Florida International University, Miami, FL 33174
Email: {mabde030, kakkaya, aibrahim}@fiu.edu
†INL Wireless Security Institute, Idaho National Laboratory, Idaho Falls, ID 83401
Email: arupjyoti.bhuyan@inl.gov
‡Computer Science Department, Marquette University, Milwaukee, WI
Email: mumin.cebe@marquette.edu

2020 IEEE Globecom Workshops (GC Wkshps) | 978-1-7281-7307-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/GCWkshps50303.2020.9367441

***Abstract*—As 5G systems are starting to be deployed and becoming part of many daily life applications, there is an increasing interest on the security of the overall system as 5G network architecture is significantly different than LTE systems. For instance, through application specific virtual network slices, one can trigger additional security measures depending on the sensitivity of the running application. Drones utilizing 5G could be a perfect example as they pose several safety threats if they are compromised. To this end, we propose a stronger authentication mechanism inspired from the idea of second-factor authentication in IT systems. Specifically, once the primary 5G authentication is executed, a specific slice can be tasked to trigger a second-factor authentication utilizing different factors from the primary one. This trigger mechanism utilizes the re-authentication procedure as specified in the 3GPP 5G standards for easy integration. Our second-factor authentication uses a special challenge-response protocol, which relies on unique drone digital ID as well as a seed and nonce generated from the slice to enable freshness. We implemented the proposed protocol in ns-3 that supports mmWave-based communication in 5G. We demonstrate that the proposed protocol is lightweight and can scale while enabling stronger security for the drones.**

***Index terms*—** Authentication, drones, 5G security, second-factor, network slices.

I. Introduction

5G standard comes with significant changes to existing cellular standards 4G/LTE to improve performance, scalability and coverage to support millions of users and devices [1]. In particular, the new 5G architecture comes with a paradigm change which relies on network slicing and virtual network functions that can be utilized on demand. In addition, for radio access, 5G adds the use of millimeter wave (mmWave) frequencies which boost data rates significantly. Finally, 5G offers integration of heterogeneous networks to support enhanced coverage through various types of base-stations.

Due to its ability to provide connectivity anywhere, anytime with much higher capacity and bandwidth capabilities, 5G revolution is enabling machine-to-machine (M2M) communications which will include IoT devices and other machines with cellular connectivity to support seamless services. Such services customized for IoT applications make 5G very attractive for deployment to reduce costs as well as management labor. Immediate examples to this are remote sensors in smart cities [2], connected vehicles on the rural roads [3], drones operating in various missions [4] and power grid devices [5].

While such increased connectedness of everything through 5G resembles the way Internet has grown to end users of various types from its inception in early 90s, this eventually resulted in numerous unprecedented security issues that relate not only to the infrastructure vulnerabilities but also the ability to easily exploit end users devices. The same can be predicted with 5G which will eventually create a wireless Internet of everything (IoE) [6]. This means not only user devices (UEs) but also all the diverse devices/machines that will be connected will pose threats to be exploited to attack 5G network infrastructure as well as exposing users' privacy. Recognizing this risk, current 5G also comes with new security protocols to ensure main security services for confidentiality, integrity and authentication [7], [8]. Nevertheless, current focus of these security services is mostly about users and their data. While there are defined procedures for M2M communication security, their assurance will not be verified until large-scale M2M applications with 5G are deployed. In addition, depending on the specific needs of M2M applications, more stringent security services could be needed beyond 5G security services.

One of such applications is drones (aka unmanned aerial vehicles (UAVs)) which can be used smart cities to emergencies, deliveries and inspections. This is because any security breach with drones may also lead to safety issues as they may be controlled maliciously to perform physical attacks. Therefore, their security procedures in oppose to fixed UEs need to be carefully designed to meet the application requirements as well as ensuring their safe operations. In particular, their identification, authentication and monitoring are crucial for local government, law enforcement and emergency officials. Hence, there is a need for additional security services for drone applications that will utilize separate network slices in 5G. A network slice is defined as a customized virtual network to serve a defined business purpose or customer, consisting of an end-to-end composition of all the various network resources necessary to meet the specific performance and economic needs of that particular class of service or customer application [9]. While network slicing is not a new feature in cellular networks, 5G network structure will extend this virtualization to an end-to-end functional scope and make embedded slicing a core functioning part of the network.

Therefore, in this paper, we propose a second factor authentication scheme to verify the authenticity of legal drones as a part of the 5G network. This second factor is inspired from multi-factor authentication mechanisms currently employed in IT systems for enhanced security. The goal is to double check the authenticity of a drone by utilizing different factors from the primary authentication that comes with 5G authentication services. Different from second-factor systems where the entire authentication depends on both first and second factors, the proposed mechanism will be in addition to primary one. The main challenges for such an authentication scheme is twofold: 1) To provide a lightweight scheme that will not bring additional burden to drones; and 2) to be able to integrate the mechanisms to current 5G standard based on 3GPP specifications [7].

To this end, for the first challenge, we propose a challenge-response based protocol that conforms with the current 5G authentication standard that utilizes drones' digital IDs which will be enforced by FAA in the US [10]. We include mechanisms such as simultaneously using a seed and nonce to prevent any replay attacks. For the second challenge, we exploit the re-authentication triggering mechanism currently in place for 5G. Basically, it is used to trigger and initiate our second-factor authentication without making any other changes in the system.

We implemented the proposed approach in ns-3 simulation environment which supports 5G radio access. The evaluation indicated that the proposed approach brings almost negligible overhead in terms of both computation and communication and can be easily integrated with network slicing in place.

This paper is organized as follows: In the next section, we provide the related work. Section III is dedicated to preliminaries to understand 5G authentication procedures. In Section IV, we explain the details of our proposed approach. Section V is dedicated to evaluation of the approach. Finally, Section VI concludes the paper.

II. Related Work

The authentication in 4G network included a unified authentication framework, better UE identity protection, enhanced home-network control, and more key separation in key derivation [1]. The proposed authentication for 5G core network is based on a service-based architecture (SBA), which enhances the previous variant currently used in 4G. The 5G network standardized the 5G Authentication and Key Agreement (AKA) protocols for this purpose [11]. These protocols work with the new structure of 5G that includes the subscribers, the Serving Networks (SNs) that have nearby base stations, and Home Networks (HNs) that corresponds to the subscribers' carriers. The AKA protocols enable the subscribers and HNs to mutually authenticate each other and to let the subscribers and SNs establish a session key [12].

There are some recent studies on the authentication aspects of 5G. For instance, software-defined networking (SDN) is utilized to enable efficient authentication hand-over and privacy protection in [13]. The authors proposed a simplified authentication handover by global management of 5G Heterogeneous Networks (HetNets) through sharing of user dependent security context information among related access points. Furthermore, in [14], the authors proposed a secure service-oriented authentication framework for IoT services in 5G network where, a privacy-preserving slice selection mechanism is introduced to allow fog nodes to select proper network slices. The work in [15] proposes a two-factor authentication but it is for wireless sensor networks (WSN) integrated with 5G. The authentication is done for the user accessing this WSN which is different than our work which explores two-factor authentication within the 5G network itself. As seen, two-factor authentication has not been considered at all for 5G core applications. Therefore, our work fills an important gap to bring strengthened security to 5G systems, especially for drone IoT applications.

III. Preliminaries

A. Background on 5G Authentication

The 5G authentication structure is a unified framework to support both 3GPP access and non-3GPP access networks such as Wi-Fi. The 5G authentication structure supports Extensible Authentication Protocol (EAP) that is also in use for IEEE 802.11 (WiFi) standard. In this regard, the 5G EAP authentication protocol supports both EAP-Transport Layer Security (TLS) and EAP-AKA protocols, where authentication process is executed between the UE (a client device) and the Authentication Server Function (AUSF)/Unified Data Management (UDM) (i.e., Home Network) through the Security Anchor Function (SEAF)/Access & Mobility Management Function (AMF) (i.e., Serving Network) as an EAP authenticator [1].

As 5G-AKA is widely used, we provide more info about its details, which is also shown in Fig. 1: 5G-AKA structure allows the SEAF function to trigger the authentication process once receiving any accessing message from a UE. In this message, the UE has to send its Global Unique Temporary Identifier (5G-GUTI) temporary identifier to initiate the authentication procedure. If the UE is not provided with a 5G-GUTI, a Subscription Concealed Identifier (SUCI) can be used. The SUCI is an encrypted version of the Subscription Permanent Identifier (SUPI) provided to each UE using the public key of the home network (i.e., it is encrypted using this key). Note that, the SUPI should never be sent in plaintext to ensure UE's privacy. Once SEAF receives the message, the authentication process is initiated by the SEAF function and an authentication request is sent to the AUSF function in the home network. The AUSF then verifies that the serving network request is authorized. If it is a legitimate request, the AUSF proceeds with the authentication procedure by sending an authentication request to UDM. Next, the Subscription Identifier De-Concealing Function (SIDF) validates the SUCI by decrypting the SUCI and obtains the corresponding SUPI and selects the authentication method configured for the corresponding subscriber which is 5G-AKA for our case. The UDM then sends an authentication response to the AUSF including an AUTH token, an $XRES$ token, the key K_{AUSF}, and the SUPI if not using the 5G-GUTI. K_{AUSF} is an important key material which can be further used to derive subsequent keys for different purposes. We will rely on this key in our approach.

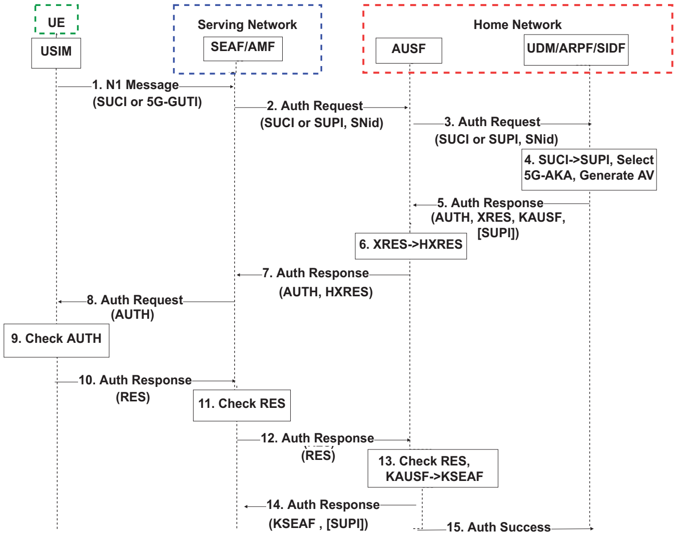

Fig. 1. 5G-AKA Procedure for authentication [1].

B. System and Attack Model

For our work, we assume a 5G cellular infrastructure where drones could get connected through the standard 5G Authentication procedure (EAP-AKA or EAP TLS). Each drone is assumed to have a digital ID assigned by Federal Aviation Administration (FAA). This is due to recent announcement from FAA in the US that each drone accessing the 5G system shall be assigned a drone Remote Identifier (Remote ID) to legally register drones [10]. We assume that the 5G Core network is trusted. However, the drones are not trusted and they can try to bypass the system to become part of the 5G network. Basically, we would like to identify any malicious drone that is not pre-registered in a database of friendly drones administered by a third party and is trying to access the network to communicate with other parties. We also assume that adversaries may try to impersonate a drone or core network to replay authentication messages back and forth.

In 5G, the service model is based on virtual network slices [16] which allows flexibility for providing differentiated services based on the needs and requirements of applications. Basically, a virtual network slice is a network customized to serve a defined business purpose or customer, consisting of an end-to-end composition of all the available network resources required to satisfy the specific performance [16]. This is the major re-haul from 4G/LTE systems and enables a lot of flexibility and efficiency. Each network slice is identified by a Single Network Slice Selection Assistance Information (S-NSSAI) which could be used by a user device (i.e., UE) when requesting access to 5G Core and 5G-RAN [17]. In our case, we assume that there is a specific network slice for the drones to provide additional authentication services as shown in Fig. 2 managed by the third party mentioned above. This slice information is provided by a drone when connecting 5G core.

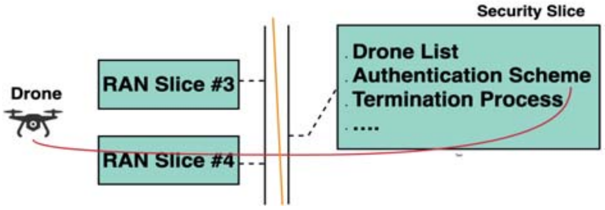

Fig. 2. Assumed network slicing for drones.

IV. Slice Specific Second-Factor Authentication

When a drone acting as a UE requests connectivity through a specific slice in 5G, the slice manager may also want to further authenticate the device for increased security in addition to 5G primary authentication. Note that this is somewhat analogous to second-factor (or multi-factor) authentication concept used in modern IT systems. However, it is in addition to primary one (i.e., primary and secondary are not linked) which is not the case in IT systems. In a sense, it can be considered as a re-authentication mechanism for more specific purposes. Nevertheless, there needs to be diversity in this additional authentication request as in the case of second-factor authentication where the goal is to increase security by using a different factor each time (e.g., asking for a text message after entering your password). To support this concept in our case, we would like to request information from the drones in this second authentication that is different from the primary authentication (e.g., ID, keys, fingerprints, etc.) while still following the EAP-based authentication used in the 3GPP standard. However, as the current 3GPP specifications do not explicitly support this type of second-factor authentication [17], we propose utilizing certain existing 3GPP procedures to integrate our second-factor authentication protocol to the current standard. Next, we explain how we can trigger this second factor by following the standard's specifications (i.e., not requesting any changes) and then we explain our authentication protocol in details.

A. Initiating Slice Specific Second-Factor Authentication

Current 3GPP standard specifications allow a re-authentication procedure for a device based on its S-NSSAI [17]. Specifically, if there is a specific S-NSSAI for drones other than the default one, this specific drone slice could dictate the Authentication Authorization and Access (AAA)-Server (AAA-S) to initiate another application specific authentication. Note that AAA-S is in charge of authentication in 5G and may be sitting in the operator's network or in a separate third party network. This procedure is called AAA-S triggered *Network Slice-Specific Re-authentication and Re-authorization procedure* [17] and its details are shown in Fig. 3. We adopt this procedure for our initiation purposes so that our approach can be easily integrated with the envisioned implementations of 5G Core.

In our approach, before AAA-S initiates the second-factor authentication, the slice functions will need to check whether

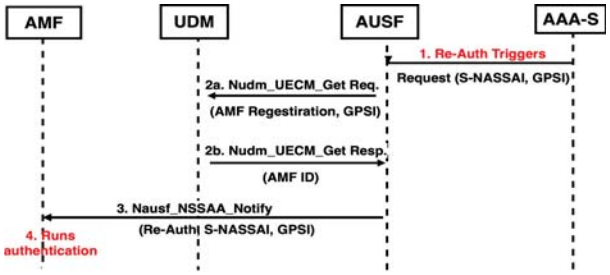

Fig. 3. AAA-S triggered Network Slice-Specific Re-authentication and Re-authorization procedure in 3GPP. We use this procedure to integrate our second-factor into the system.

the SUPI of the device registered exists in a drone database created in advance. If the SUPI of the registered drone is within this database, then a second-factor will be mandated. This is exactly where our approach kicks in: We exploit the 5G standard's ability to do re-authentication to trigger a mandatory second-factor authentication for drones to further secure their communication. This approach will be initiated by AAA-S, which requests Generic Public Subscription Identifier (GPSI) for the devices. Note that GPSI is used for addressing a 3GPP subscription in different data networks outside of the 3GPP system. But since the 3GPP system stores within the subscription data the association between the GPSI and the corresponding SUPI, it is easy to make this mapping. The initiation process follows the procedure in 3GPP standards, where AAA-S informs the AMF to request registration from the drone. AMF will initiate a *challenge-response* EAP protocol to the newly authenticated drone that will follow our own proposal to differentiate it from the primary one. This EAP-compliant procedure is explained in the next subsection.

B. Second-factor authentication protocol

Our protocol follows a challenge-response type authentication procedure since it needs to conform with the current EAP framework for 5G authentication. This framework is flexible in the sense that it allows replacing the underlying authentication protocol such as AKA or TLS. The main motivation behind our approach is to enable a more restricted authentication specific to our application that will rely on different information from the primary 5G authentication. To this end, we utilize two new items that have no relationship with the prior material generated during primary authentication: 1) The digital ID of the drone: As mentioned before, this unique ID will be different than any other IDs that might be assigned by the 5G system; and 2) A new symmetric key different from the existing key hierarchy: This key is produced from a unique seed generated by the machine managing the related network slice function so that it will not have any relation with the key seed K_{SEAF} produced during the primary authentication.

Our challenge-response protocol kicks in after AMF (i.e., the party responsible to handle the process after AAA-S informs it about second authentication request) follows the standard registration procedure for the drone. Basically, it sends an EAP ID request message and gets a EAP Response from the drone which is passed to ASF and AAA-Server as part of the initiation procedures. This process is shown in Fig. 4 in black messages. The rest of the authentication process between the AAA-S and the drone, shown as blue in the same figure, is detailed below. Note that we could directly initiate the authentication from the AAA-S without resorting to EAP-Request and Response messages. Since this is part of the initiation process, we basically follow the standard's messages to ensure that our protocol can be fully integrated.

- **Challenge from AAA-S**: The AAS-S prepares a challenge to be relayed to the drone by the intermediate components AUSF/UDM and AMF/SEAF. This includes a random number R and a seed $Seed$ generated by the hosting computer using pseudo-random generator each time there is a need for a secondary authentication. The $Seed$ is encrypted using the symmetric key, K_{SEAF} which was produced in the primary authentication phase and then sent to drone D_{ID} along with R. Moreover, ID is the FAA remote identifier assigned to the drone:
$$AAA - S \rightarrow E_{K_{SEAF}}(Seed), R \tag{1}$$
- **Challenge Response Preparation**: The receiving drone, D_{ID} calculates the challenge reply based on a unique symmetric key T which is created by using private ID, $Seed$ and R sent to it:
$$T = F(Seed, ID, R) \tag{2}$$
where F is a deterministic random bit generator (DRBG) function [18]. The drone then uses this T as a symmetric key and m as a dummy message and creates a secure message authentication code (HMAC) [19]. This HMAC and m are then relayed to the AAA-S as follows: The drone then uses this T and creates a secure message authentication code (HMAC) [19] message using T as a symmetric key and m as a dummy message.
$$HMAC(T|m), m \rightarrow AAA - S \tag{3}$$
- **Response Verification**: AAA-S receives these $HMAC(T|m)$ and m pair and recomputes a new HMAC by using the info stored locally in the database (i.e., drone ID, $Seed$ and R to re-generate T). If the new HMAC and the received one matches, then it sends an ACK message to the drone to finish the second-factor authentication:
$$ACK \rightarrow Drone \tag{4}$$
If they do not match, then a de-registration procedure is initiated. The AAA-S contacts AMF to initiate this process for the UE, which is already part of the 3GPP standard.

C. Security Analysis of the Proposed Protocol

Our second-factor authentication utilizes unique information from drone and AAA-S. Therefore, any drone whose unique ID is not in the database will be de-registered from the network when our second-factor is triggered. The protocol is also resilient against any replay or integrity attacks. Any adversary that tries to create an HMAC will fail due to lack of access to the secret key T. In addition, each time the AAA-S will generate a new seed $Seed$ so any replay attack from an imposter server will fail due to mismatching of $Seed$

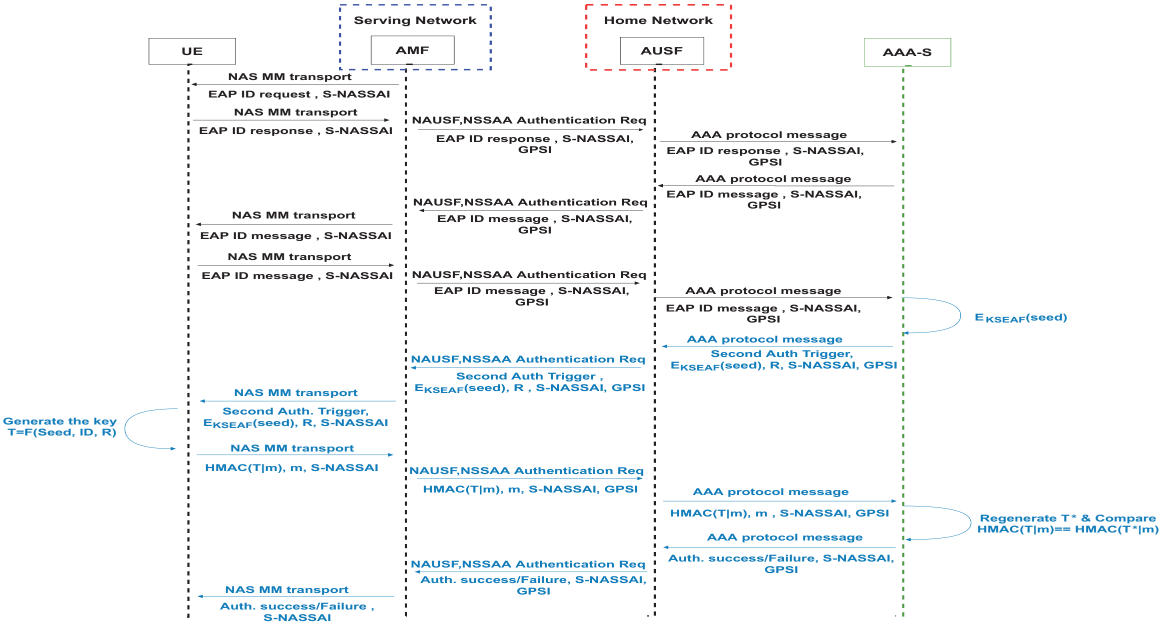
Serving Network
Home Network
UE
AMF
AUSF
AAA-S
NAS MM transport
EAP ID request , S-NASSAI
NAS MM transport
EAP ID response , S-NASSAI
NAUSF,NSSAA Authentication Req
EAP ID response , S-NASSAI, GPSI
AAA protocol message
EAP ID response , S-NASSAI, GPSI
AAA protocol message
EAP ID message , S-NASSAI, GPSI
NAUSF,NSSAA Authentication Req
EAP ID message , S-NASSAI, GPSI
NAS MM transport
EAP ID message , S-NASSAI
NAS MM transport
EAP ID message , S-NASSAI
NAUSF,NSSAA Authentication Req
EAP ID message , S-NASSAI, GPSI
AAA protocol message
EAP ID message , S-NASSAI, GPSI
E_KSEAF(seed)
AAA protocol message
Second Auth Trigger, E_KSEAF(seed), R, S-NASSAI, GPSI
NAUSF,NSSAA Authentication Req
Second Auth Trigger , E_KSEAF(seed), R , S-NASSAI, GPSI
NAS MM transport
Second Auth. Trigger, E_KSEAF(seed), R, S-NASSAI
Generate the key T=F(Seed, ID, R)
NAS MM transport
HMAC(T|m), m, S-NASSAI
NAUSF,NSSAA Authentication Req
HMAC(T|m), m, S-NASSAI, GPSI
AAA protocol message
HMAC(T|m), m , S-NASSAI, GPSI
Regenerate T* & Compare HMAC(T|m)== HMAC(T*|m)
AAA protocol message
Auth. success/Failure, S-NASSAI, GPSI
NAUSF,NSSAA Authentication Req
Auth. success/Failure, S-NASSAI, GPSI
NAS MM transport
Auth. success/Failure , S-NASSAI

Fig. 4. Second factor authentication registration shown in black messages and proposed protocol shown in blue messages

values. Similarly, since the drone is using a new R each time, any replay attack from drone side will also not be possible. These values ensure that authentication messages are all fresh. Finally, even if a drone ID is compromised, this can not be used in future authentications because the system requires new $Seed$ and R values (i.e., forward secrecy).

V. Experimental Evaluation

A. Experiment Setup

In order to assess the performance of the proposed second factor authentication, we utilized ns-3 network simulator which has recently implemented 5G RAN module [20]. Nevertheless, it still does not support new 5G Core and thus we needed to simulate the slicing on the server side. Specifically, we created a UE node (node 1) to represent a drone and another server node (node 2) to represent the core network's AUSF server all of which serve as ns-3 nodes. This AUSF is connected to another server (node 3) which will represent the AAA-S and specific network slice for drones. We created an Ethernet connection from the AUSF server to AAA-S to indicate connections between them assuming AAA-S can act as a representative in a virtual function. The overall architecture for this implementation setup is shown in Fig. 5. In this implementation, we initiate the process by sending a message from UE to AUSF assuming this will be the completion of primary authentication, which then contacts AAA-S through the Ethernet connection for slice specific authentication. Our implementation starts with AAA-S contacting core network (i.e., AMF) to contact the UE which will start running messaging shown in Fig. 4. Table I lists the system parameters for ns-3 simulation as well as the bit sizes for keys used in the experiments.

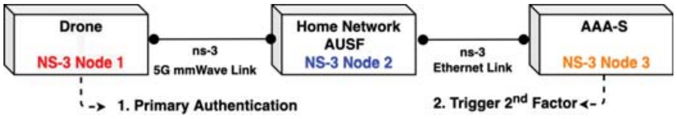

Fig. 5. ns-3 implementation setup.

TABLE I
SIMULATION PARAMETERS

Parameter	Value
Packet size	1000 bit
Data rate	30 Mb/sec
Background nodes traffic	10
gNodeB distance	300 m
inter packet interval	100
Seed size	440 bits
Remote ID size	32 bit
K_SEAF size	256 bit
HMAC type	SHA256

B. Metrics and Baselines

To assess the overhead of our proposed authentication mechanism, we considered the *total authentication time*, which includes all the communication and computation delays during the authentication process. Note that due to limited battery and resources on a drone, the computational delay is crucial in determining the overhead of the proposed authentication scheme. Hence, toward a more realistic assessment, we used a Raspberry-Pi3 IoT device to mimic the behavior of the drone.

C. Performance Results

1) Drone Computational Overhead: The computational delays experienced by the drone through the second-factor authentication are in Table II. Hence, the total processing delay for our proposed secondary authentication is 0.942msec. Moreover, the utilization of the DRBG hash provides a faster computational time, and hence, the total computational time is less than 1msec which is even less than the total time for primary 5G-AKA authentication.

TABLE II
COMPUTATIONAL OVERHEAD COMOARISON

Approach	Operation	Delay (msec)
2^{nd} Factor	DRBG-Hash	0.16
2^{nd} Factor	HMAC	0.78
2^{nd} Factor	**Total**	**0.94**
5G-AKA	**Total**	**1.02**

2) Communication Delay: The communication delays experienced between the AAA-S and the drone throughout the second-factor authentication are in Table III. As seen, the total authentication delay is 7msec for one drone authentication. Note that since our approach also uses challenge-response based authentication, the communication delay is almost similar due to the same number of messages exchanged for primary authentication. The only additional delay for our approach is the triggering time, which is 3.62msec in total. Overall, the total time of 7msec is an important figure since this is the amount of time provided to drone to act maliciously until the second-factor authentication de-registers it if the drone is malicious. During that 7msec, it is not possible for the drone to collect and transmit any meaningful data, which indicates the effectiveness of our approach.

TABLE III
SECOND-FACTOR COMMUNICATION OVERHEAD

Approach	Connection	Delay(msec)
2^{nd} Factor	AUSF to AAA-S Ethernet	1.50
2^{nd} Factor	Drone to AUSF	1.12
2^{nd} Factor	TCP Handshake Time	2.18
2^{nd} Factor	**Total Communication Delay**	**7.00**
5G-AKA	**Total Communication Delay**	**3.38**

3) Impact of Background Traffic Delay: Another factor, we investigated is the impact of background traffic from other existing nodes within the same cell during the second-factor authentication. To further investigate this point, we simulated both Uplink and Downlink background traffic connecting to the AUSF server simultaneously while starting the second-factor authentication. The traffic frequency at each node is set to 1msec interval between packets transmissions and the maximum number of packets sent by each node is set to 100000. This setup is considered a heavy bulk background traffic over the server. As shown in Table IV, the total authentication delay based on the high background traffic up to 100 nodes is within 0.4 µsec. Hence, under a heavy background traffic the additional delay is negligible, which means no extra delay overhead on the proposed authentication.

TABLE IV
DELAY UNDER VARYING BACKGROUND TRAFFIC

Background Nodes	Delay (msec)
10	7.000753
50	7.000810
100	7.000968

VI. CONCLUSION

In this paper, we presented a new security scheme for 5G that will enable second-factor authentication for drones. This authentication further strengthens the security of M2M-based applications based on their slice-specific restrictions on security. Therefore, we designed the approach in such a way that it can be integrated to current standard. In addition, we utilized different information to make this second-factor authentication totally different from the primary default authentication of 5G. We implemented the approach in ns-3 using 5G mmWave radio access. The evaluation of the approach indicated its efficiency and feasibility.

VII. ACKNOWLEDGEMENTS

This work is supported by a grant from the Idaho National Lab Wireless Security institute (INL WSI). The work of Ahmed Ibrahim is supported in part by the National Science Foundation under award number CNS-1618692.

REFERENCES

[1] I. Inside, “A comparative introduction to 4g and 5g authentication.” WINTER 2019.

[2] S. H. Alsamhi, O. Ma, M. S. Ansari, and F. A. Almalki, “Survey on collaborative smart drones and internet of things for improving smartness of smart cities,” *IEEE Access*, vol. 7, pp. 128 125–128 152, 2019.

[3] H. Ullah, N. G. Nair, A. Moore, C. Nugent, P. Muschamp, and M. Cuevas, “5g communication: an overview of vehicle-to-everything, drones, and healthcare use-cases,” *IEEE Access*, vol. 7, 2019.

[4] A. Koubâa, B. Qureshi, M.-F. Sriti, A. Allouch, Y. Javed, M. Alajlan, O. Cheikhrouhou, M. Khalgui, and E. Tovar, “Dronemap planner: A service-oriented cloud-based management system for the internet-of-drones,” *Ad Hoc Networks*, vol. 86, pp. 46–62, 2019.

[5] R. A. Fernandes, “Monitoring system for power lines and right-of-way using remotely piloted drone,” Apr. 4 1989, uS Patent 4,818,990.

[6] I. Ahmad, T. Kumar, M. Liyanage, J. Okwuibe, M. Ylianttila, and A. Gurtov, “Overview of 5g security challenges and solutions,” *IEEE Communications Standards Magazine*, vol. 2, no. 1, pp. 36–43, 2018.

[7] P. Schneider and G. Horn, “Towards 5g security,” in *IEEE Trustcom/BigDataSE/ISPA*, vol. 1. IEEE, 2015, pp. 1165–1170.

[8] R. P. Jover and V. Marojevic, “Security and protocol exploit analysis of the 5g specifications,” *IEEE Access*, vol. 7, pp. 24 956–24 963, 2019.

[9] K. Sparks, M. Sirbu, J. Nasielski, L. Merrill, K. Leddy, P. Krishnaswamy, W. Johnston, R. Gyurek, B. Daly, M. Bayliss, J. Barnhill, and K. Balachandran, “5G network slicing whitepaper.”

[10] Federal Aviation Administration, “UAS remote identification,” https://www.faa.gov/uas/research_development/remote_id/, Mar 2020.

[11] A. Koutsos, “The 5g-aka authentication protocol privacy,” in *IEEE European Symposium on Security and Privacy (EuroS P)*, 2019.

[12] D. Basin, J. Dreier, L. Hirschi, S. Radomirovic, R. Sasse, and V. Stettler, “A formal analysis of 5g authentication,” in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS ’18. New York, NY, USA: Association for Computing Machinery, Jan 2018, p. 1383–1396. [Online]. Available: https://doi.org/10.1145/3243734.3243846

[13] X. Duan and X. Wang, “Authentication handover and privacy protection in 5g hetnets using software-defined networking,” *IEEE Communications Magazine*, vol. 53, no. 4, pp. 28–35, Apr 2015.

[14] J. Ni, X. Lin, and X. S. Shen, “Efficient and secure service-oriented authentication supporting network slicing for 5g-enabled iot,” *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, 2018.

[15] S. Shin and T. Kwon, “Two-factor authenticated key agreement supporting unlinkability in 5g-integrated wireless sensor networks,” *IEEE Access*, vol. 6, pp. 11 229–11 241, 2018.

[16] X. Li, M. Samaka, H. A. Chan, D. Bhamare, L. Gupta, C. Guo, and R. Jain, “Network slicing for 5g: Challenges and opportunities,” *IEEE Internet Computing*, vol. 21, no. 5, pp. 20–27, Sep 2017.

[17] “Ts 23.502 - procedures for the 5g system.” [Online]. Available: https://www.tech-invite.com/3m23/tinv-3gpp-23-502.html

[18] E. Barker, L. Feldman, and G. Witte, “Recommendation for random number generation using deterministic random bit generators,” National Institute of Standards and Technology, Tech. Rep., Aug 2015.

[19] S. Kelly and S. Frankel, “Using hmac-sha-256, hmac-sha-384, and hmac-sha-512 with ipsec,” RFC 4868, May, Tech. Rep., 2007.

[20] N. Wireless and the University of Padova, “mmWave cellular network simulator,” https://apps.nsnam.org/app/mmwave/, Sep 2018.