Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 8984789, 12 pages
https://doi.org/10.1155/2022/8984789

Research Article

DIM-Based Random Number Generation Using Quantum Noise Resources

Hansaem Wi,[1] **Seyoon Lee**,[1] **and Okyeon Yi**[2]

[1]*Department of Financial Information Security at Kookmin University, Seoul 02707, Republic of Korea*
[2]*Department of Information Security Cryptology and Mathematics at Kookmin University, Seoul 02707, Republic of Korea*

Correspondence should be addressed to Okyeon Yi; oyyi@kookmin.ac.kr

Received 29 March 2022; Revised 17 August 2022; Accepted 28 September 2022; Published 9 November 2022

Academic Editor: Yan Huo

Currently, unmanned aircraft systems (UASs) or drones are in service in various industrial fields, and each UAS operator establishes and operates their own independent drone system. These individual drone systems interact only with their own components without any integrated management. As the number of UASs is increasing due to the expansion of the drone industry, standardized operation is required. Therefore, to integrate and manage existing drone systems, the Federal Aviation Administration and National Aeronautics and Space Administration devised UAS Traffic Management (UTM). The drone identity module (DIM), which is being developed as a drone identification device, securely stores the remote identification (RID) of each drone and performs a cryptographic operation to secure information between the drone and UTM infrastructure. The DIM performs cryptographic authentication protocols to achieve cryptographic identification and authentication with the UTM infrastructure, which requires random numbers. Modern cryptographic systems rely on difficult computations, and an environment capable of generating secure cryptographic random numbers must be configured to provide high computational costs to attackers. In this paper, we explain the need for random numbers in the DIM, analyze random number generators used in related drone-based studies, and analyze the characteristics of noise resource generation devices that can be used in existing drone systems. Subsequently, based on the analysis results, existing methods are used to generate random numbers in the DIM, and limitations are derived. To overcome these limitations, we propose a method of generating random numbers in the DIM using quantum noise resources. For our proposal, we conduct an analysis of the physical specifications of noise resource generation devices, DIM prototypes, and quantum noise resource generators in existing drone systems, and we present the results of NIST 800-90B entropy measurement using data collected from quantum random number generators.

1. Introduction

With the advent of 4th Industrial Revolution, data security in the IoT environment is becoming important, and various security technologies are being actively studied accordingly [1, 2]. In addition, drone systems are used in various industrial fields in interaction with IoT and are becoming one of the most important industrial fields in the 4th Industrial Revolution. It is estimated that the number of unmanned aircraft systems (UASs), i.e., drones, currently operated at low altitude for recreational and commercial purposes in the United States will increase from approximately 2 million in 2021 to approximately 3 million by 2023 [3]. To design an architecture capable of integrating and managing such an increasing number of UASs, the Federal Aviation Administration (FAA), a US airspace management agency, announced Concept of Operation v2.0 (ConOps v2.0) [3], which explains the necessity of UAS Traffic Management (UTM), details of UTM design, drone operation scenarios within UTM, and requirements for drones participating in UTM. Drones are a well-known example of UASs, and UTM includes systems for operating drones. From an information protection and cryptographic perspective, the most important of the various requirements described in Ref. [3] are identification and certification of drones performing services within UTM. The expansion of the drone

industry, such as drone taxis, drone delivery, and precision agriculture using drones, will necessarily require UTM, thereby making it an important infrastructure for the country. However, security incidents caused by security threats in critical infrastructure can cause enormous economic and human damage. Accordingly, security for data communicated between drones and UTM is essential. To form a secure channel between drones and UTM, identification and authentication are essential.

However, existing drone systems use products released by large drone manufactures, such as DJI and Parrot, or use open platforms such as Pixhawk and Raspberry Pi, and there is no standard between these heterogeneous drones. These drones cannot be included and operated in UTM because they only use functions dependent on existing platforms without requirements for separate identification and authentication functions. Therefore, for existing drone systems to be included in UTM and perform services, an authentication method that can integrate each drone system is required, and ISO 23629-8 standardizes requirements for remote identification of drones [4]. In addition, ISO/IEC 22460-2 standardizes requirements and details for the dataset, cryptographic operation function, and hardware of the drone identity module (DIM), a standardized device for identifying drones included in UTM [5]. Currently, standardization of the DIM is in the preparatory stage, and if standardization is completed and applied to drones, it will be a security-only module that provides information security for communication between drones and UTM, not just identification information storage for a drone.

ConOps v2.0 states that cryptographic authentication protocols are required for identification and authentication of drones flying under UTM [3]. In other words, the DIM installed on the drone must form a secure data communication channel based on cryptographic authentication with the UTM authentication server. In this case, the cryptographic authentication protocol requires a random number, and accordingly, the cryptographic random number generation function is essential for the authentication server of the DIM and UTM. Cryptographic random numbers are used not only as time variants in cryptographic authentication protocols but also to generate security parameters, such as cryptographic keys that determine the safety of the cryptographic system. Determining the security strength of cryptographic random numbers is the entropy of the noise resources constituting the seed for generating random numbers. Therefore, the DIM should be equipped with a device capable of generating sufficient noise resources to satisfy the security strength of the cryptographic random number along with the cryptographic random number generation function. However, there are no published data on a method of generating random numbers for the DIM. Moreover, the specific details of available noise resource generating devices and related research are insufficient. In this study, we conducted an analysis focusing on how to generate noise resources to derive the requirements necessary to propose a random number generation method for the DIM. We analyzed the limitations of noise resource generation methods using existing drone systems, based on which we propose a random number generation method for the DIM using quantum noise resources.

The contributions of this study can be summarized as follows:

(i) Analysis of differences in operating environment between existing drone systems and UTM: from the perspective of random number generation methods, we analyzed the differences between existing drone systems and the operating environment of drone systems in UTM, and we analyzed how these differences affect random number generation methods

(ii) Analysis of usage limits for random number generator in existing drones: based on the contents of related studies, we analyzed the limitations of methods used for random number generation in existing drone systems when applied to the DIM, and based on this, we derived the requirements necessary to prepare a random number generation method in the DIM

(iii) Proposal of DIM using quantum noise source: based on the analysis results of the limitations of noise resource generation methods using existing drone systems, we propose a random number generation method for the DIM using quantum noise resources

2. Background

2.1. UTM System. ConOps v2.0 shows the interaction between participants in UTM through the notional UTM architecture and describes the functions and requirements of each participant in UTM [3].

Each participant in UTM described in Figure 1 refers to a system developed and distributed by the FAA and industry as a specific function for UTM operation, and UTM is operated through organic data communication between these systems. The main participants that make up UTM include flight information management system (FIMS), UAS service supplier (USS), supplementary data service provider (SDSP), and UAS operators. FIMS is a system operated by a national airspace management agency and is UTM's best management system to record UAS operations and track data that can be used for incident management and audit in the future. The USS provides a network that allows multiple UAS operators to receive real-time flight information from FIMS and SDSP databases and provides flight management. UAS operators can receive auxiliary data, such as current flight plans and weather management required for flight through the USS. For organic data communication of these components, UTM recommends configuring secure channels based on cryptographic identification and authentication between entities [3].

2.2. Drone Identity Module. Currently, there is no standard document for reference to the operation of the DIM. However, some information can be obtained from Dr. Tak's presentation at the Unmanned System Congress held in

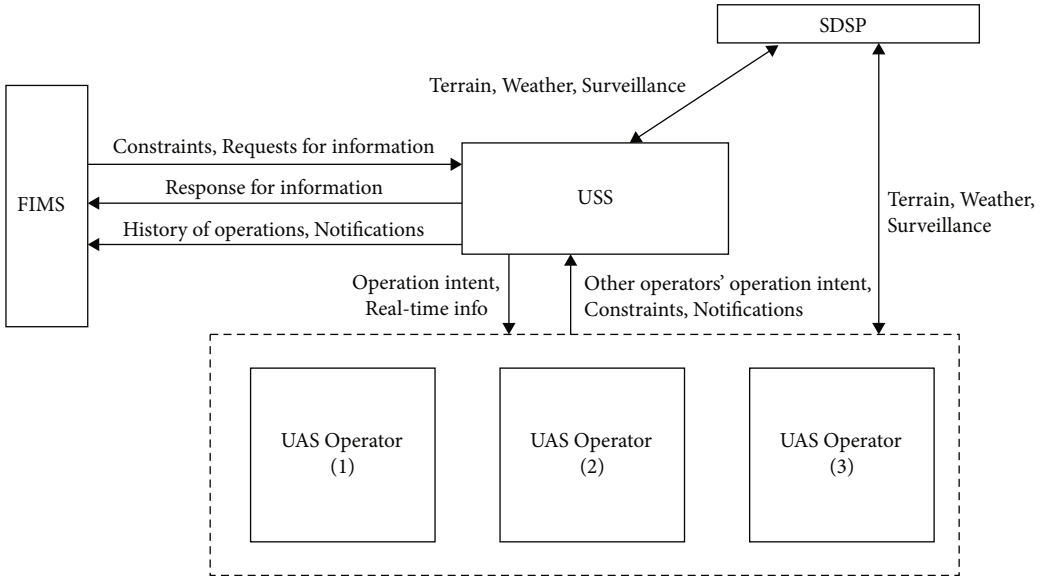

Figure 1: Notional UTM architecture [3]. This figure shows the general operation structure of UTM. FIMS requests USS to record flights performed by constrain and UAS operators at UTM, and USS, which manages UAS operators, requests such information back to UAS operators and delivers it to FIMS. FIMS stores and manages records of all flights occurring in UTM and uses these records as data for future accidents or audits. SDSP can deliver data related to terrain, weather, surveillance, etc. that can support drone flight to UAS operators through USS or directly to UAS operators. For the above interaction, drones operated by UAS operators must perform the flight permit authorization process to participate in UTM.

Korea in 2021. Dr. Tak is the convener of ISO/IEC JTC 1/SC 17/WG 12, which performs the standardization of ISO/IEC 22460. According to his presentation, drones must obtain flight permission from the UTM Flight Permit Authority Server to fly under UTM, and in the process, cryptographic mutual authentication between the DIM and UTM Flight Permit Authority Server is performed. The Flight Permit Authority Server does not exist in existing drone systems, and the process of obtaining a drone's flight permission is an additional process in UTM. Existing drone systems consisting of commercialized platforms initiate flights in a manner specified by the platforms used by each UAS rather than an integrated flight permit process, and identification and authentication processes using cryptographic methods are not necessarily required. However, Dr. Tak's presentation showed that for drones to start flying in UTM using the DIM, they must perform a Flight Permit Authority process using standardized RIDs and cryptographic authentication protocols [4, 6]. This process is to prevent unauthorized illegal drones from entering the UTM and securely manage drones that perform legitimate services, and this process of preflight licensing is the biggest difference between existing drone systems and UTM operations. Figure 2 shows the components of the DIM. The DIM uses the drone's RID to perform flight authorization, and the identification issued after registration allows UTM to manage drone behavior while the drone is flying. As described in ISO/IEC 22460-2, the DIM is a module for drones that can securely store identification information and perform cryptographic functions necessary for information security [5].

Two essential functions are required from the DIM [5]. First, a function capable of storing information necessary for drone identification and authentication is required. Here, the information required for identification and authentication includes identification information, registration information, a certificate, and an encryption key that allows the drone to obtain airspace flight permission and maintain its status during flight. The DIM should be in the form of a black box so users are not authorized to access such information. Second, a cryptographic operation function for the information security function performed by the DIM is required. A drone uses the DIM to perform identification and certification processes with UTM infrastructure before and during flight using RIDs stored in the DIM. When performing the identification and authentication process, mutual authentication is performed through a cryptographic protocol, and the DIM must be able to execute cryptographic functions, such as electronic signature, hash function, and random number generation at this time.

2.3. Entropy Sources, Random Numbers, and Quantum Random Number Generator. When building a cryptosystem, generating random numbers that can satisfy unpredictable security strengths with the computing power of an attacker is as important as accurately implementing a cryptographic algorithm. Cryptographic algorithms in modern cryptosystems, such as encryption, message authentication code generation, and public key systems, provide security against computational complexity assuming that keys generated in a cryptographically secure manner are securely stored [7, 8]. In other words, if a cryptographic random number generator is implemented where bits of security parameters are predictable enough to make an attack meaningful or biased

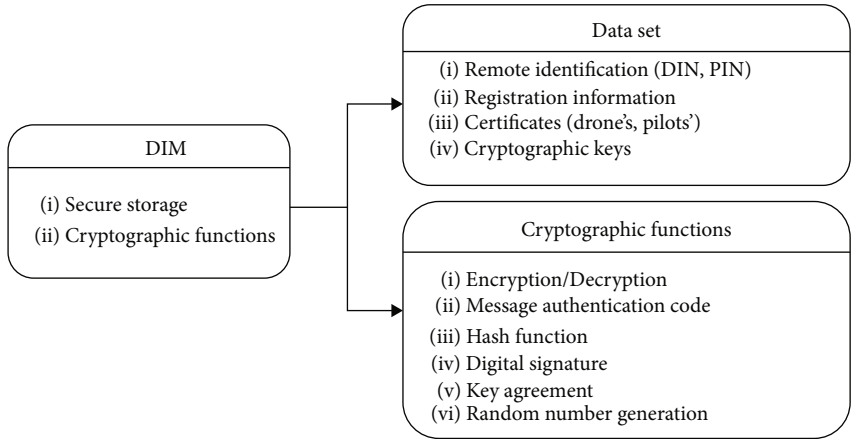

Figure 2: DIM components. It must have a black box type storage that can securely store identification information and security parameters and a cryptographic function for performing information security functions [5].

enough to not be used as security parameters, the cryptographic system may be destroyed by an attacker.

The random number generator required by ISO/IEC 19790 sets an international standard for the security of cryptographic modules in the form of software and hardware in various devices, including IT systems. ISO/IEC 19790 requires a random number generator that refers to the model presented in NIST 800-90A [9]. The model is a structure that generates a seed based on noise resources and enables repeated random number generation using a deterministic random number generator. The deterministic random number generator operates based on block cipher, Hash, and HMAC, and the security of the generated random number depends on the seed. The entropy of the seed input to the deterministic random bit generator (DRBG) must satisfy the security strength of random number to be generated. Figure 3 shows the random number generation method given by NIST 800-90A [10].

Entropy resource refers to a bit string after postprocessing, such as a process for removing bias that may occur due to noise resource characteristics. Assuming that the DRBG described in Figure 3 is implemented without errors, the entropy of noise resources with nondeterministic properties determines the security of the corresponding random number generator. Noise resources available in random number generators used in general IT systems use data indicating the state of software running on the operating system (OS) or values input through user interfaces, such as mouse and keyboard, with entropy characteristics. Noise resources obtained based on hardware include the thermal noise of a resistor and atmospheric noise of a diode. The collected noise resources are used either directly to construct the seed of the random number generator or as entropy resources after increasing the entropy ratio through a postprocessing process [11]. Random numbers generated through the process shown in Figure 3 are used to generate security parameters in the cryptosystem. A well-known example is that when executing cryptographic authentication protocols, the provider generates random numbers to defend against attacker replay and interleaving attacks and uses them as time variants to provide unity and timeliness of the protocol being executed [12].

Quantum random number generator (QRNG) refers to a device that generates unpredictable random numbers based on quantum noise. It is assumed that the usage structure of QRNG described in this paper follows the structure of the random number generator shown in [10], as described in Figure 4. The reason for this assumption is that the type of random number generator suggested in Ref. [10] follows the structure referenced in [9], which is an international standard, so the QRNG must have this structure to be applied from private facilities to important national infrastructure. Well-known QRNG methods are to use noise resources generated during beam splitter transmission of a single photon and radioisotope decay [13–15]. Quantum noise is a signal that is collected based on the phenomenon that protons, which are very small particles, inherit the uncertainty of both. Therefore, compared to noise obtained from general software and hardware, it is possible to create an entropy source with high entropy, and because quantum particles are very small, it is possible to construct an environment that can generate noise in a physically small space [13].

3. Related Work

3.1. Authentication Scenarios with DIM [16]. In [16], DIM standardization editors propose the cryptographic authentication structure of the DIM with UTM's authentication server and implemented the authentication structure using the oneM2M platform, a protocol for application data communication of IoT devices. In [16], a certificate-based authentication protocol between the DIM and authentication server is proposed, and the implementation process and experimental results are presented. It should be noted that in [16], the proposed authentication structure attempted to provide freshness for sessions that perform authentication processes using random numbers generated by the DIM and authentication server, and normal random number generation is premised.

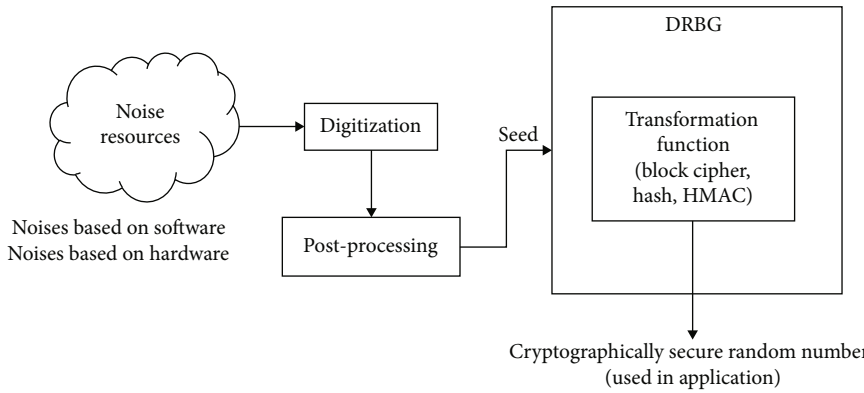

Figure 3: Random number generation process in cryptographic system using DRBG [10]. The security strength of the random number depends on the seed obtained by processing the noise resources. In general, noise resources may be obtained from an environment for operating an existing IT system or a device of an IoT system. In some cases, a dedicated device for generating noise resources is configured and used to generate random numbers.

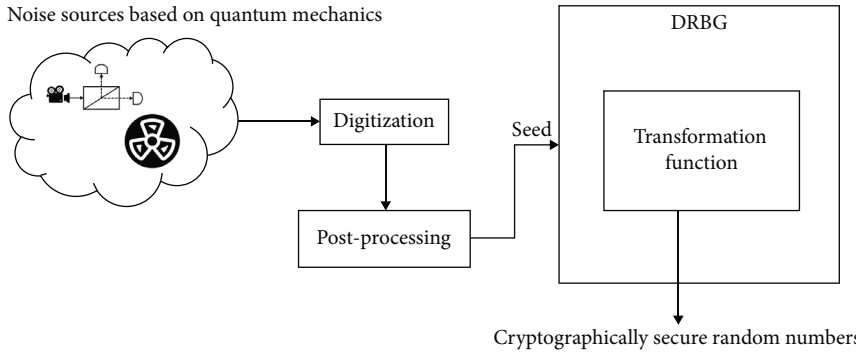

Figure 4: Random number generation process in cryptographic system using quantum mechanics. When configuring the quantum random number generator, the most important aspect is building small parts that generate quantum noise resources. This is directly related to the cost of a quantum random number generator, and accordingly, commercialized quantum random number generators are produced in a compact size compared to general hardware parts. It is well-known how to generate a random sequence using the spacing of random signals generated during the decay of radioactive isotopes or by calculating photons passing through or reflected from beam splitters.

Figure 5 shows the cryptographic mutual authentication structure proposed in [16], and it can be seen that it has an authentication structure based on a certificate. The server uses the oneM2M platform to send and receive messages, and after the authentication process is successfully completed, the shared key is calculated using the random numbers shared with each other. The authentication and key matching process using the certificate proposed in Ref. [16] is widely used in general IT systems as well as in drone systems combined with IoT environments, and the oneM2M platform is a widely used message protocol in IoT environments.

However, the authentication structure proposed in Ref. [16] is performed assuming normal random number generation in the DIM and authentication server. It can be seen that random number generation is essential in the DIM, and accordingly, it is necessary to consider how to generate random numbers considering the software and hardware characteristics constituting the DIM.

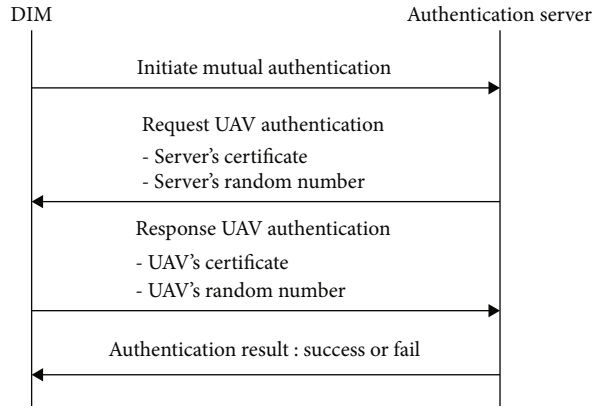

Figure 5: Mutual authentication structure proposed in [16].

3.2. Drone Random Number Generator Based on Sensor Data. In [17], a drone random number generator is proposed using data from sensors installed to control and monitor the state of the drone, which is an essential component in a general drone system, as noise resources. The motivation for Ref. [17] was Ref. [18], which studied the secure version of MAVLink, an open drone control message

protocol. In Ref. [17], the authors claim that the security of the cryptographic key must be ensured for the operation of the cryptographic system in the drone, which was conducted in [18]. For this reason, they designed a random number generator for the secure generation of security parameters, such as the cryptographic key. The drone used for the experiment in this research is equipped with Pixhawk and Raspberry Pi as flight controllers. Pixhawk and Raspberry Pi are currently widely used in DIY drones due to their convenience of development and openness of development-related materials. It is thought that the authors chose these settings to design a random number generator that can be commonly applied to general drone systems. In [17], data collected from an accelerometer, gyroscope, and barometer, which are sensors related to the attitude and speed of the drone, were used as noise sources for the random number generator.

Figure 6 sequentially shows the process of the experiment performed in [17]. The authors performed the statistical tests recommended in NIST SP 800-22 [19] to check the quality of random numbers output by the random number generator. The results of the NIST 800-22 test for the random number generator output for the drone in Ref. [17] are presented in Table 2 in [17], indicating that the random number generator outputs designed by the authors of Ref. [17] are statistically random. It is worth focusing the test results of the raw sensor data collected when the drone is stationary in the results presented in Ref. [17]. The stationary state expressed by the authors refers to a state in which the drone does not fly and is stationary on the ground with the power turned on. The authors of Ref. [17] performed the NIST 800-22 test suite on raw data collected from the accelerometer, gyroscope, and barometer, and as a result, some NIST 800-22 tests failed. DroneRNG designed dividing and shuffling functions to improve the lack of randomness in raw data. The random numbers generated through the "mix and swap" process showed success on the entire NIST 800-22 test suite. The method of Ref. [17] is meaningful in that it generates random numbers necessary for operating a cryptographic system in a drone using Pixhawk and Raspberry Pi, which are widely used in commercial drone systems.

4. Experiment and Discussion

4.1. Analysis of Noise Resources Available in Existing Drone Systems. Figure 7 shows the structure of an existing drone system operated without the introduction of the UTM concept. Existing drone systems use the Ground Control System (GCS), a software that can control commercialized drones, to configure and perform services. The types of drones used range from ready-to-fly drones manufactured and released by large manufacturers, such as DJI and Parrot, to DIY drones using open hardware platforms, such as Pixhawk and Raspberry Pi. The drone must be equipped with a flight controller, a computer that processes logic related to flight, and an optional companion computer, a computer for further user data processing. The flight controller and companion computer interact with the GCS and user application server, respectively, to control, monitor, and transmit user data. The flight controller is a device that controls the posture and speed of a drone when it flies and allows the UAS operator to monitor the drone's state based on the values of sensors installed in the drone. Because drones' flight capabilities are heavily weighted, flight controllers are generally manufactured with compact specifications, such as with Pixhawk and Naze32. The companion computer is mounted on the drone and installed to process user data, such as video and photos. The companion computer is built to have higher specifications than the flight controller to process high-definition images and operates on rich operating systems, such as Windows and Linux.

The biggest change in drone systems after the introduction of the concept of UTM is that each drone system must participate in UTM in a unified manner. Unlike existing drone systems that allowed drones to fly only with a connection between the flight controller and GCS, the UTM architecture requires DIMs that store standardized RIDs to participate in the system. As explained in Background, the DIM must obtain permission to fly through cryptographic authentication with UTM's Flight Permit Authority Server and must generate random numbers. Accordingly, we select the flight controller as a candidate random number generating device among the elements constituting existing drone systems, and we analyze the limitations of this method using the research contents of [17].

Random number generation follows two steps [10]:

Step 1: construct seeds based on collected noise resources

Step 2: DRBG operation by inputting parameters for random number generation including the seed configured in Step 1

The DRBG operating in Step 2 is a deterministic algorithm that always has the same result for the same input. For this reason, it can be implemented in software and on various types of processors. However, because collecting noise resources used in Step 1 requires resources with nondeterministic characteristics generated by software or hardware, the method of generating noise resources is implemented in various ways depending on the software, hardware types, and characteristics of the encryption system.

We have established two scenarios for collecting the required noise resources in the DIM. The first is to generate noise sources using components of existing drone systems, and the second is to generate noise resources using the DIM itself. However, in the first scenario, we excluded the noise source collection scenario using the companion computer from the analysis to assume the most common situation, as the scenario of collecting noise sources from the optional companion computer component cannot be applied to drones that operate with a flight controller alone.

The flight controller has several types of sensors built in to control the posture and speed and monitor the conditions of the drone, and additional sensors are sometimes attached to improve performance. The data output by these sensors are mixed with natural noise when drones fly to create irregular patterns and have entropy characteristics. In [17], which designed a random number generator using sensor data as a noise source, a random number generator using

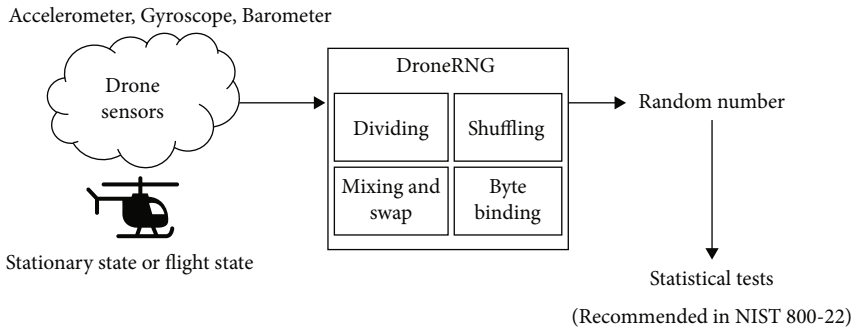

Figure 6: Random number generator and statistical test process proposed in [17].

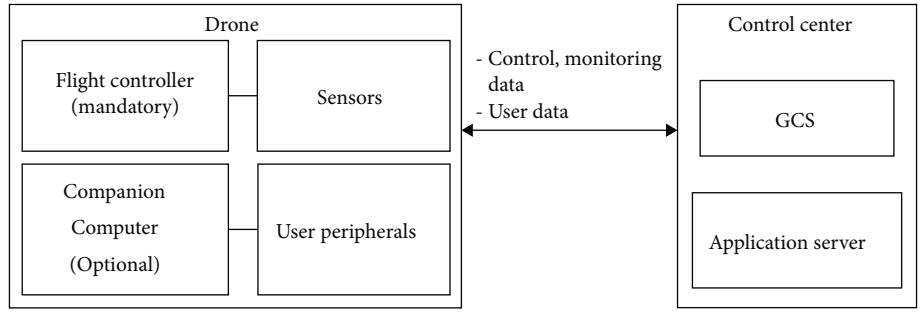

Figure 7: General drone system structure. This figure shows the general structure of existing drone systems. There are visual line of sight (VLOS) and beyond visual line of sight (BVLOS) methods for controlling drones. In the VLOS method, the pilot uses a pilot controller to control the drone within visible distance, and the BLOS method means that the drone is controlled outside the pilot's field of view. The structure of the figure represents an existing drone system using the BVLOS method. ConOps v2.0 [3] describes a UTM architecture that considers both VLOS and BLLOS drone control.

the accelerometer, barometer, and gyroscope mounted on a drone were designed using Pixhawk and Raspberry Pi as noise resources.

However, because these contents use the principle of generating noise resources using irregular pattern in air resistance, pilot manual operation, and sensor values due to obstacle avoidance, the conditioning process cannot guarantee entropy of noise resources. In addition, Ref. [17] claims that the conditioning process was performed to improve the entropy of the raw data of the sensors, and the results of the statistical random number test were presented through the results on the NIST 800-22 test suite. However, the conditioning process can only increase the entropy ratio by removing the bias of the noise resource, not the absolute entropy of the noise resource, and the NIST 800-90B test suite [11] must be performed to determine how much entropy has improved. In other words, the results presented in Ref. [17] cannot be seen as evidence for entropy improvement. The evaluation method of entropy and random number should be different, but a method for statistically evaluating random numbers was used to evaluate the collected entropy. Therefore, from a strict point of view, Ref. [17]'s experimental analysis is wrong. In order for the experimental analysis of Ref. [17] to be done properly, the authors should have measured the entropy obtained from the sensor data through statistical tests such as SP 800-90B and presented how much improvement the entropy obtained from the pure sensor data was made when the designed method was designed.

The values output by the sensors of the drone change within a small margin of error when the drone is not flying. For this reason, it is obvious that the entropy value of sensor data represents a small value compared to when flying. More entropy may be measured in places where natural noise is severe, but otherwise, it is difficult to supply the entropy required for the drone at an appropriate time.

Therefore, to generate entropy that is not affected by the state of the drone and always provides the same strength of security, a noise resource that is not affected by the state of flight is required, and the NIST 800-90B test suite must measure the level of noise entropy.

In Figure 8, the drone is in an "unlicensed" state and cannot fly under UTM. The drone will perform mutual authentication with the UTM Flight Permit Authority to acquire flight rights. In this process, the drone and Flight Permit Authority perform cryptographic protocols, and random number generation is required. Because the drone landed on the ground and received only the power needed for communication and is not in flight, the values of the sensors installed in the drone will only show small changes in values due to natural noise and will not show any significant uncertainty. Based on the above analysis results, it can be seen that noise-generating devices, such as sensors installed in the drone, depend on the flight of the drone,

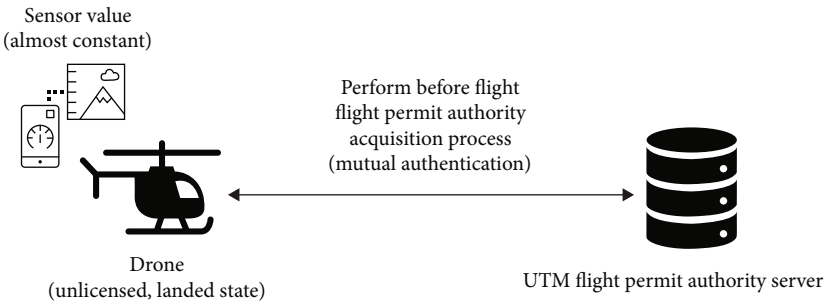

Figure 8: Status of drones and sensors when performing cryptographic authentication with UTM Flight Permit Authority Server. This figure shows the state of the sensor value installed in the drone when the drone and UTM Flight Permit Authority perform cryptographic authentication. Drones in the unlicensed state cannot fly until they obtain flight permission.

and there is a limit to cover scenarios in which the flight of the drone is not performed, such as Flight Permit Authorization. Through this, a method for collecting noise resources with the same level of entropy before and after the drone flight is performed is needed.

4.2. DIM Structure Using Quantum Noise Source. A way to overcome the limitations of the noise resource collection scenario using existing drone systems is to mount the noise resource generating device on the DIM itself, which is the second set scenario. However, to mount the noise resource generating device on the DIM, it is necessary to have a compact size and always provide the same level of entropy without being affected by the flight of the drone.

We propose DIM using QRNG as a way to satisfy these conditions. The noise resource generation device using quantum mechanics used in QRNG can be implemented at a compact size, and as long as the normal operation of the device is guaranteed, it can supply the noise resources required for the DIM independently of the drone's flight.

4.2.1. Compact Size of QRNG. Availability is an essential information security function that must be satisfied along with confidentiality and integrity when composing a cryptographic system. In the case of drones, their weight and size can lead to reduced ability to fly, and smaller drones are more sensitive to this. Therefore, the weight and size of the hardware installed on the drone to configure the cryptographic system are very important. In this analysis, we compare the noise generator candidates for existing drone systems with the physical specifications of the noise generators used in QRNG to analyze whether they have sufficiently compact size to be mounted on the DIM. What should be noted in the analysis results below is the overhead of the weight and area that occurs when additionally installed in a drone to generate entropy. A mini computer such as Raspberry Pi is a device designed to run an operating system with a built-in multichip that handles multiple functions. Sensors can include chip that can digitally convert analog signals and the weight and area of the case surrounding them. Finally, the devices that generate the quantum noise source proposed in this paper are hardware composed of single chip. These devices operate through system communication with the hardware that makes up the device, and their area is also very small compared to the two aforementioned hardware.

Table 1: Prototype DIM dimensions [20, 21]. In the case of SoC chip, it means an SoC with microcontroller mainly used in the mobile and IoT environments, and the general size is not indicated because each manufacturer has different specifications. However, compared to the flight controller and companion computer used in existing drone systems, it is manufactured with a compact size.

Hardware	General size (width × length × height; mm)
SD card	$32.0 \times 24.0 \times 2.1$
Mini SD card	$21.5 \times 20.0 \times 1.4$
Micro SD card	$15.0 \times 11.0 \times 1.0$
Full-size SIM	$85.6 \times 53.9 \times 0.76$
Mini SIM	$25.0 \times 15.0 \times 0.76$
Micro SIM	$15.0 \times 12.0 \times 0.76$
Nano SIM	$12.3 \times 8.8 \times 0.67$
SoC chip	Varies

In general, the weight of the hardware increases in proportion to the area of the hardware. The hardware that generates the chip-type quantum noise source proposed in this paper has an area of up to 5.0×5.0 mm, so it is much smaller than the comparison Raspberry Pi and sensor. Therefore, the weight is significantly lighter than the two hardware. For this reason, the comparison of weights is omitted and the overhead in the area is sufficient to analyze the possibility of installation in the DIM prototype.

Table 1 lists the physical specifications of SD cards and SIM cards that are being studied as DIM prototypes. Table 2 lists the devices that can collect noise resources for generating random numbers in existing drone systems and is presented to compare the difference in specifications with the DIM prototype. Although companion computer was excluded from the noise generation method using existing drone systems, the size of the Compute Module, the smallest version of Raspberry Pi, is presented in the table to show the difference between the physical specifications of the existing drone system and DIM.

Table 2: Size of Raspberry Pi Compute module and Pixhawk's sensor. The thickness of Raspberry Pi indicates only the height of the PCB excluding the height of components. In the case of sensor, it shows the specifications of the sensor used in Pixhawk, which is widely used in general drone systems. MPU 6000 is a sensor used as the main accelerometer and gyroscope of Pixhawk, and the MS5611 series is a sensor used as the main barometer of Pixhawk. The reference link to the specification for each product is as follows: Compute Module 3+: https://www.raspberrypi.com/products/compute-module-3-plus/, Compute Module 4: https://www.raspberrypi.com/products/compute-module-4/, MPU 6000: https://invensense. http://tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf, and MEAS: https://datasheetspdf.com/pdf/921406/measurement/MS5611-01BA03/1.

Manufacturer	Product	Size (width × length × height; mm)	Dependent on drone flight
RPi	Compute Module 3+	67.6 × 31.0 × 1.1	N
RPi	Compute Module 4	55.0 × 40.0 × 1.3	N
InvenSense	MPU 6000	4.0 × 4.0 × 0.9	Y
MEAS	MS5611-01BA	5.0 × 3.0 × 1.0	Y

Table 3: Size of noise-generating devices used in commercial QRNG. This table shows the sizes of noise generators used for quantum noise generation in QRNG. The environment that can generate quantum mechanical phenomena is configured in the noise generator. The noise generator manufactured by EYL uses radioactive isotope decay, and the noise generator manufactured by IDQ uses photons to generate noise. The reference link to the specification for each product is as follows: EYL: https://www.eylpartners.com/index.php/quantum-entropy-chip/ and IDQ: https://www.idquantique.com/random-number-generation/products/quantis-qrng-chip/.

Manufacturer	Product	Size (mm)	Dependent on drone flight
EYL	QEC 1.0	5.0 × 5.0	N
EYL	QEC 2.0	3.0 × 3.0	N
EYL	QEC 3.0	3.0 × 3.0	N
IDQ	IDQ6MC1	4.2 × 5.0 × 1.1	N
IDQ	IDQ20MC1	4.2 × 5.0 × 1.1	N
IDQ	IDQ250C2	2.5 × 2.5 × 0.84	N

Compared to the physical specifications of the DIM prototype presented in Table 2, the physical specifications of MPU 6000 and MS5611-01BA in Table 3 are smaller than those of the DIM prototype. However, as confirmed in the previous analysis of the noise resource collection limit in existing drone systems, the sensor data cannot provide a constant entropy independent of whether the drone is flying, so it is not suitable to be mounted as a noise-generating device in the DIM. In addition, considering only the physical specifications, the physical specifications of the Compute Module, the smallest version of Raspberry Pi, are larger than the specifications of most DIM prototypes. Therefore, this is also not suitable for mounting as a noise-generating device in the DIM.

Table 3 lists the sizes of noise generators used in commercialized QRNG. The devices presented in Table 3 have compact specifications and can be installed mostly under the standard specifications of the prototype of the DIM shown in Table 1. The reason for this is that quantum mechanical phenomena can be fabricated with very small hardware. Quantum noise-generating devices can generate a noise source that always has the same level of entropy independently of the flight of the drone if it is assumed that the hardware in which the quantum mechanical phenomenon is implemented is preserved. In addition, in the case

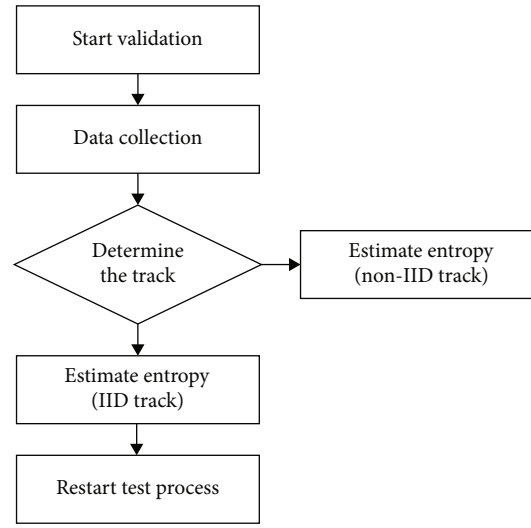

Figure 9: NIST SP 800-90B entropy estimation strategy [11]. This figure shows the entropy measurement process described in NIST SP 800-90B. The entropy evaluation process is performed in a statistical manner. If a dataset has IID characteristics, IID track is performed. Otherwise, statistical entropy measurement is performed on non-IID track. In this experiment, the restart test process was not performed.

Table 4: Specification of QRNG. This table shows the details of the QRNG used to collect the dataset for the experiment. We used QRNG based on the radioactive decay phenomenon to collect quantum noise sources, and the experiment was performed with the QRNG's microcontroller and the noise generator configured for communication.

QRNG	Description
Noise type	Radioactive decay
Sampling	60-100
Size (noise)	3.0 × 3.0
Size (QRNG)	20.0 × 20.0

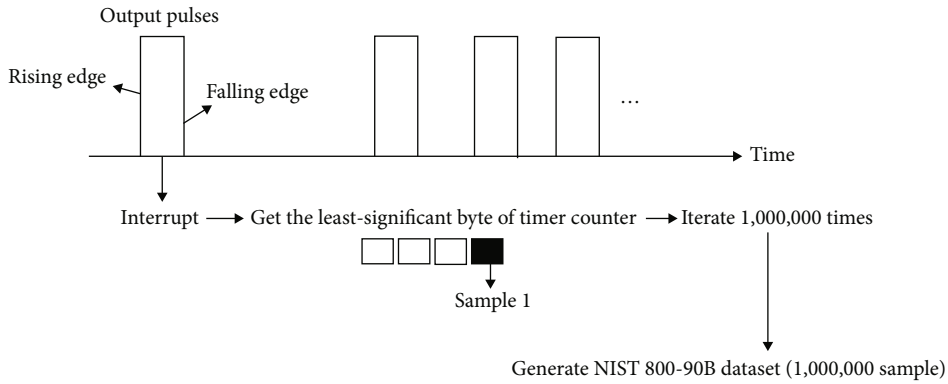

Figure 10: Process of sampling the quantum noise resource used in the experiment. The QRNG used in the experiment consists of a microcontroller in which the DRBG algorithm that generates random numbers by processing entropy is implemented with a device that generates quantum noise. We sampled the noise through communication with a microcontroller and noise generator.

of radioactive isotope decay, there is an additional advantage that the drone can perform its functions without being affected by the surrounding environment when flying because the effect of heat or pressure is insignificant [22].

4.2.2. Experiment and Analysis. To present the entropy value that can be expected when using the DIM equipped with the quantum noise source device proposed in this paper, entropy resources are collected from the device generating the quantum noise sources, the dataset is configured, and the entropy measurement test of NIST 800-90B presents the results.

The method of measuring entropy is based on several statistical tests. In other words, there are several ways to measure entropy, and the measurement result varies depending on which entropy measurement method is used. In the NIST 800-90B test performed in this study, entropy per collected single sample was measured based on Min-Entropy. This is a method of measuring the entropy of a sample from the worst distribution among the distributions obtained through statistical testing of a dataset that collects noise resource samples. Figure 9 shows the entropy evaluation process for entropy sources recommended by NIST 800-90B. Before the entropy evaluation process, permutation testing is performed on the dataset of the collected samples, and based on the results, it is determined whether the noise resources have independent and identically distributed (IID) characteristics. In this result, bit strings with IID characteristics are evaluated for entropy by a statistical method in the IID track, and for non-IID tracks, entropy is evaluated by a statistical method in the non-IID track. In this study, we used a C++-based tool distributed by NIST through GitHub for entropy evaluation of QRNG resources. Using this tool, it is possible to measure the entropy that can be obtained per sample of the noise resource collected from the QRNG constituting the collected dataset [18]. This can then be used as a basis for quantifying the amount of noise resources to sample to satisfy the required security strength when constructing a cryptographic system using QRNG. Table 4 lists the specifications of the QRNG and entropy collection device used for this experiment, and Figure 10 shows the process of collecting samples to perform the tests recommended by NIST 800-90B.

The QRNG used in the experiment consists of a chip that generates a noise resource and a microcontroller that receives the generated noise resource and executes DRBG, and it is configured to process the signal when noise occurs in the microcontroller. The type of noise resource used in QRNG utilizes the decay of radioactive isotopes, and 60–100 samples can be collected per second using the noise. We used the microcontroller's timer counter to collect samples of the noise source needed for the experiment. When the main function is executed, it is programmed to collect the lower 8 bits of the timer counter when the pulse generated by the noise generator is detected. The timer counter consists of an integer of 4 bytes and repeats the initialization process when the maximum value is reached. We judged that entropy characteristics can be expected in the lower 8 bits of the timer counter according to the generation of quantum noise. To collect 1,000,000 samples, which is the amount of data required by NIST SP 800-90B, the above process was iterated 1,000,000 times, and the configured dataset was input in binary format to the entropy measuring tool distributed by NIST.

The minimum entropy measured in the non-IID track was measured to be approximately 6.64 bits per sample of 8-bit size. This means that entropy can be obtained with an efficiency of approximately 83% per sample, and approximately 20 samplings are required to generate a security parameter with 128-bit security strength. The experimental results failed in the permutation test to determine whether it was IID, and it was determined that the dependency occurred between samples due to the repetition of the lower 8 bits of the timer counter, the sample responsible for this phenomenon.

Figure 11 shows the operation structure of the DIM using quantum noise resources. We previously demonstrated that they could be mounted on the DIM through the analysis of the physical specifications of quantum noise generators, and the results of the NIST SP 800-90B experiment indicate how much sampling is required when used in the DIM. As shown in Figure 11, the random number

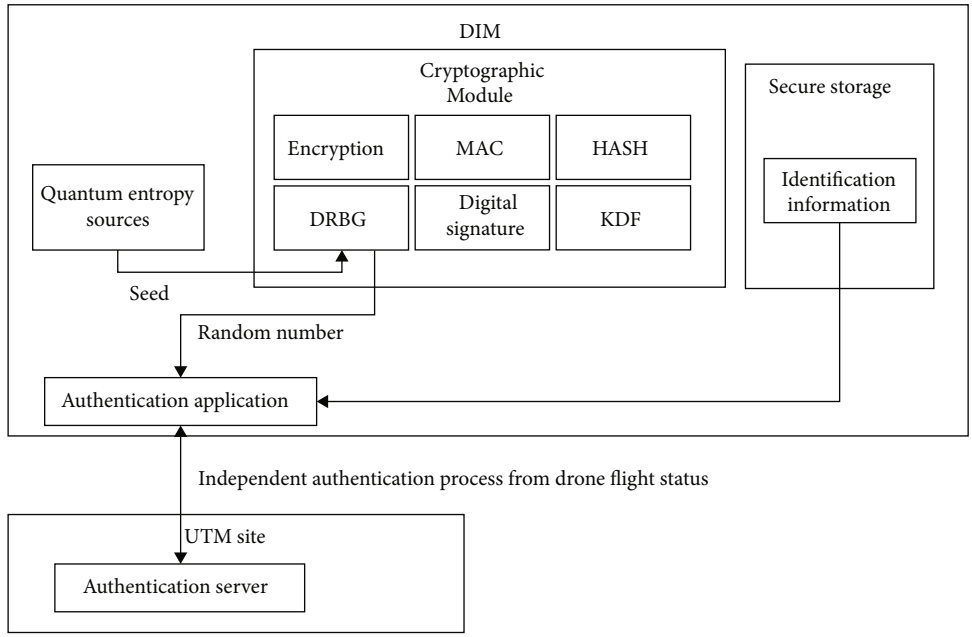

Figure 11: Operational structure of the DIM equipped with QRNG device. This figure shows the structure of a DIM equipped with a quantum noise resource generating device for constructing a cryptographic system. The DIM operating with quantum noise-generating devices can configure a seed to operate DRBG using a quantum noise resource and can perform the necessary cryptographic authentication in UTM using the random number output by DRBG. The important point here is that if the DIM structure is designed in this manner, independent cryptographic operation that is not affected by the flight status of the drone is possible.

generation method of the DIM using quantum noise resources can always satisfy the level of cryptographic random number security required by UTM. This advantage allows the Flight Permit Authority Server, which is required before flying a drone, to perform cryptographic operations independently of the drone's state in all scenarios of DIM in UTM, including cryptographic authentication, thereby securely performing the information security functions required by UTM.

5. Conclusion

In this study, we analyzed the noise resource generation method using the existing drone system and derived the limitations of the method based on the results. In addition, a method for generating random numbers in the DIM using quantum noise resources, which is a method to solve this limitation, was presented.

In the NIST SP 800-90B test performed in this study, entropy measurement through the IID track could not be performed due to permutation failure. In the entropy measurement process using a statistical method, the entropy measurement result may be different from the intended result depending on the noise resource postprocessing method. The microcontroller timer counter used in the entropy measurement experiment in this study has a size of 4 bytes, and the lower 8 bits are extracted and used to collect 8-bit samples. 8-bit has a short cycle period, which may have created a dependency between samples. In future work, we believe that entropy measurement in the IID track should also be performed by improving the postprocessing method.

UTM will become a major national infrastructure, and accordingly, information security devices to be used in UTM must perform data security using a cryptographic module whose safety has been proven through the Cryptographic Module Validation Process (CMVP). Therefore, the DIM is an information security device used in UTM and should be developed as a cryptographic module that can safely store identification information and perform independent cryptographic operation.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (Ministry of National Defense) (2022-0-00701, Development of Security Technology for Interworking between M-BcN and 5G Commercial Network).

References

[1] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communication*, vol. 38, no. 5, 2020.

[2] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *39th IEEE International Conference on Distributed Computing Systems*, Dallas, Texas, USA, 2019.

[3] Federal Aviation Administration, *Concept of Operations V2.0 – Unmanned Aircraft System (UAS) Traffic Management (UTM)*, 2020.

[4] UAS, *Traffic Management (UTM) – Part 8: Remote Identification*, ISO/CD 23629-8, Internation Organization for Standardization, Switzerland, 2022.

[5] ISO, *License and Drone Identity Module for Drone (Ultralight Vehicle or Unmanned aircraft system) – Part 2: Drone identity module (DIM)*, ISO/IEC AWI 22460-2, Internation Organization for Standardization, Switzerland, 2019.

[6] International Organization for Standardization, *Categorization and Classification of Civil Unmanned Aircraft Systems*, ISO 21895, Internation Organization for Standardization, Switzerland, 2020.

[7] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, USA, 5th edition, 2001.

[8] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*, CRC Press, USA, 2nd edition, 2014.

[9] International Organization for Standardization, *Information Technology - Security Techniques–Security Requirements for Cryptographic Modules*, ISO/IEC 19790, Internation Organization for Standardization, Switzerland, 2012.

[10] National Institute of Standards and Technology, *Recommendation for Random Number Generation Using Deterministic Random Bit Generators*, NIST SP800-90A Revision 1, National Institute of Standards and Technology, USA, 2015.

[11] National Institute of Standards and Technology, *Recommendation for the Entropy Sources Used for Random Bit Generation*, NIST SP800-90B, National Institute of Standards and Technology, USA, 2018.

[12] B. Schoenmakers, *Lecture Notes Cryptographic Protocols*, Department of Mathematics and Computer Science, Technical University of Eindhoven, Netherland, 2022.

[13] J. Park, S. Cho, T. Lim, and M. Tehranipoor, "QEC: a quantum entropy chip and its applications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 6, pp. 1471–1484, 2020.

[14] T. Jennewein, U. Achleitner, G. Weihs, H. Weinfurter, and A. Zeilinger, "A fast and compact quantum random number generator," *Review of Scientific Instruments.*, vol. 71, no. 4, pp. 1675–1680, 2000.

[15] M. Rohe, *RANDy - A True-Random Generator Based on Radioactive Decay*, 2003.

[16] K. Kim and Y. Kang, "Implentation of UAS identification and authentication on one M2M IoT platform," in *2019 International Conference on Information and Communication Technology Convergenece*, Jeju-si, Jesu-do, South Korea, 2019.

[17] S.-M. Cho, E. Hong, and S.-H. Seo, "Random number generator using sensors for drone," *IEEE Access*, vol. 8, pp. 30343–30354, 2020.

[18] A. Allouch, O. Cheikhrouhou, A. Koubaa, M. Khalgui, and T. Abbes, "MAVSec: securing the MAVLink protocol for Ardupilot/PX4 unmanned aeiral systems," http://arxiv.org/abs/1905.00265.

[19] International Organization for Standardization, *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*, NIST SP800-22, National Institute of Standards and Technology, USA, 2010.

[20] International Organization for Standardization, *IIdentification Cards-Physical Characteristics*, ISO/IEC 7810, Internation Organization for Standardization, Switzerland, 2019.

[21] European Telecommunications Standards Institute, *Smart Cards; UICC-Terminal interface; Physical and logical characteristics*, ETSI TS 102 221, European Telecommunications Standards Institute, France, 2013.

[22] E. D. Flakenberg, "Radioactive decay caused by neutrinos?," *Apeiron*, vol. 8, no. 2, 2001.