Cyber-Physical Fuzzing Framework for Extracting Butterfly Effect in UAV Systems

Se-Han Lee
SysCore Lab.
(Convergence Engineering for Intelligent Drone)
Sejong University
Seoul, Republic of Korea
sehanlee141@gmail.com

Sang-Hoon Choi
SysCore Lab.
Sejong University
Seoul, Republic of Korea
csh0052@gmail.com

Woohyun Jang
Cyber Electronic Warfare R&D
LIG Nex1
Pangyo, Republic of Korea
woohyun.jang2@lignex1.com

Shinwoo Shim
Cyber Electronic Warfare R&D
LIG Nex1
Pangyo, Republic of Korea
shimshinwoo@lignex1.com

Yeon-Jae Kim
Cyber Electronic Warfare R&D
LIG Nex1
Pangyo, Republic of Korea
yeonjae.kim@lignex1.com

Ki-Woong Park*
Dept. of Computer and Information Security
Sejong University
Seoul, Republic of Korea
woongbak@sejong.ac.kr

***Abstract*— Today, commercial Unmanned Aerial Vehicles (UAVs) are being utilized in various industrial fields due to the emergence and development of various models. However, cyber threats targeting UAVs that exploit this convenience of use are also increasing. Cyber threats such as controlling commercial UAVs are causing human casualties by being used for criminal activities such as war, terrorism, and unauthorized surveillance. Therefore, a cyber threat response plan for UAVs is necessary. In this regard, we propose a new fuzzing framework for UAV systems based on the butterfly effect, where a small cause leads to a large-scale incident in the future. In the proposed fuzzing framework, flight behavior control-related factors that can affect the occurrence of a neutralizing situation in a UAV system are analyzed and collected. Then, the factors (i.e. impact vectors) for finding the impact chain, which is a UAV neutralizing flow that can occur due to a combination of various factors, are identified. In this paper, we delineate the design configuration of the framework, outline the implementation methodology, and introduce a testbed for UAV simulation and flight status data acquisition within the framework. This framework can be utilized in the future to secure cyber security of UAV systems, ensure security, and establish offensive security strategies to respond to cyber threats.**

Keywords—Cyber-Physical System, Unmanned Aerial Vehicle, Fuzzing Framework, Butterfly Effect, Impact Chain

I. INTRODUCTION

The emergence and development of commercial UAVs that can be conveniently used by anyone can be utilized in various fields such as aerial video shooting, aerial logistics delivery, constructing Internet of Things (IoT)-based network infrastructure through aerial mobile base stations, and surveillance to detect unauthorized persons in specific security areas [1, 2]. In addition, the Pixhawk Project (e.g., PX4, Ardupilot), an open source-based UAV system development project, has emerged and is being utilized in many fields[3].

However as these UAV systems have grown, so have cyber threats. For example, malware attacks are now focusing on open source libraries in UAV systems, and Distributed Denial of Service (DDoS) threats are now targeting the network infrastructure that UAV systems need to work [4, 5].

The UAV systems exist in various forms depending on their intended use, and each manufacturer uses different system construction methods, communication protocols, and system languages, creating an environment that is difficult for security experts to analyze [6]. For this reason, various security vulnerability testing methods in existing computing systems cannot be used as they are, and a new approach is needed.

To address this, we looked at what system components all UAV systems have in common, regardless of manufacturer or open source project, and focused on the fact that all UAV systems are subject to the same laws of physics when in flight. And we checked that the same control process principle is performed to control flight within these physical laws [7].

Therefore, in this work, we analyze and collect various components related to the flight behavior control of UAV systems and propose a Cyber-Physical Fuzzing Framework to discover vulnerabilities to neutralize UAV systems. The proposed framework is based on the butterfly effect [8, 9], which states that one small cause can lead to a large result in the future and can be utilized to create a flow in which various flight control impact factors (i.e., impact vectors) of a UAV system are related and to derive a flow (i.e., impact chain) that achieves the goal of neutralizing the UAV system among various flows.

In this paper, we describe the design structure of the framework, the methodology for deriving the impact chain. Also we describe the design and implementation of testbed for the framework for testing UAV system in simulation environment and deriving flight behavior related status data within the framework.

This paper is structured as follows. In Section II, we describe the background and related works. In Section III, we describe the framework design structure and the methodology of deriving the impact chain. In Section IV, we describe a design and implementation of testbed for UAV simulation testing and derive the flight status data. In Section V we describe the conclusion and future research plan.

II. PRELIMINARY BACKGROUND

A. General Flight Control in UAV Systems

The UAV system essentially runs a real-time motion control process to perform flight control inputs from the user. Based on this, the UAV system strives to maintain a stable flight attitude and continues until the mission is completed. The process that all UAV systems have in common, especially for flight attitude control, is the PID control process. This is a future-oriented concept that is often discussed in the field of

* Corresponding author

2831-7203/24/$31.00 ©2024 IEEE
DOI 10.1109/AIxDKE63520.2024.00018

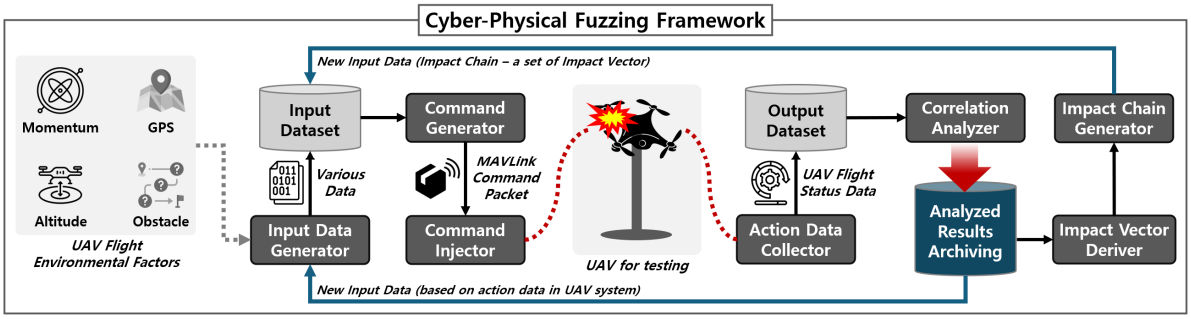

Fig. 1. An overview of Cyber-Physical Fuzzing Framework for UAV systems

control engineering, and it secures the stability of the output result according to the input of the system by using proportional, integral, and differential calculations [7].

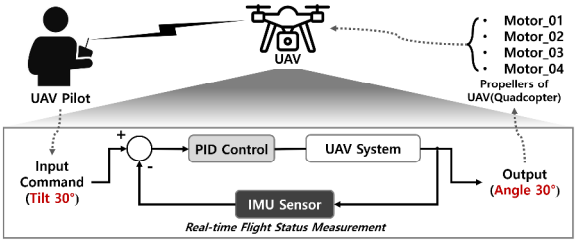

Fig. 2. An overview of flight control process in UAV systems

The Figure 2 shows an overview of flight control process using PID control technique in UAV systems. In order to accurately output the user's desired motion input, the UAV system senses the current attitude information and calculates using PID control to output the same attitude control value as the user's desired motion input value. Based on the calculated value, the output value of each wing motor is adjusted so that the user's desired motion can occur.

B. Related Works on Offensive Research for UAV Systems

As commercial UAV systems emerge and are utilized in various fields, research on vulnerability investigation and analysis of UAV systems is actively being conducted. A brief summary of some of the research results related to this is shown in Table 1.

TABLE I. A PART OF RELATED WORKS ON OFFENSIVE RESEARCH FOR UAV SYSTEMS

Approaches	Methods	Ref.
Cyber	Firmware Fuzzing	[10]
	DUML Protocol Packet Fuzzing	[11]
Physical	EMI Injection	[12]
	Acoustic Wave Injection	[13]
Cyber-Physical	Control Logic Equation & Firmware Code Analysis	[14]

First, the cyber-level approach directly extracts the firmware of the UAV system [10] or conducts an offensive research targeting the DJI Universal Markup Language (DUML) packets used in UAV systems of specific manufacturers such as DJI [11]. This approach has the disadvantage that only SW-related elements can be analyzed and that a lot of information about the target UAV system is required. Next, the physical-level approach can be confirmed to utilize physical elements such as electromagnetic interference (EMI) [12] or acoustic waves [13] that can occur in hardware circuits. This approach has the disadvantage that it is very difficult to obtain vulnerabilities related to software due to the research results being highly dependent on hardware. Lastly, the cyber-level and physical-level approach combines the UAV system control logic equation and firmware code analysis method to conduct an offensive research [14]. This is an approach that can understand the UAV system and consider both hardware and software characteristics.

The framework proposed in this work is also a fuzzing method that can discover vulnerabilities by considering both the cyber-level and physical-level. However, the difference from previous work is that rather than directly analyzing firmware code to find correlations, we derive correlations between hardware and software based on flight state data that can commonly occur in all UAV systems.

III. CYBER-PHYSICAL FUZZING FRAMEWORK

In this section, we describe a Cyber-Physical Fuzzing Framework for the purpose of analyzing and collecting various components associated with the flight behavior control of a UAV system. We also describe a method to find impact chains by utilizing the proposed framework.

A. Design and Configuration of Fuzzing Framework

In this paper, we propose a fuzzing framework to identify common flight control behavior related components of various UAV systems and to collect and analyze them for future derivation of impact chains. The configuration diagram of the proposed framework is as shown in Figure 1.

First, to create input data related to flight behavior control, an input dataset (command dataset for the UAV system) is created based on the flight related factors of the UAV system. The method for creating the input dataset is described in Section IV. After that, it is injected into the UAV system to be tested, the resulting data is collected, and the correlation between the resulting data and flight control factors is analyzed to derive the impact vectors. Using the derived impact vectors, various impact chain are constructed and sent back to the input dataset to verify whether it is an impact chain close to neutralizing the UAV. If there is not enough impact vector to create an impact chain, various action data from the result data archive can be passed to the Input Data Generator to create a new input dataset. By doing this process over and over, different flight behavior control components can be gathered and analyzed. Different input-output data pairs, including intended and unintended motion and malfunction due to an internal system error or crash, can be found by looking at how the UAV system worked with the collected components data.

B. A Methodology for Derivation of Impact Chains

Using the proposed fuzzing framework, we can analyze, identify, and collect various components related to the flight behavior control of the UAV system. Based on the collected various components, we planned a method to find the impact chain to discover vulnerabilities that may occur in the future to paralyze the UAV system.

This is a method inspired by the butterfly effect [8, 9], which states that small causes can lead to large results in the future. Among the various components collected through the framework proposed in this paper, there are elements that cannot be modified within the system, but there are also elements whose values can be modified by the user directly inputting them. We performed tests utilizing all components whose values can be modified, and proposed a fuzzing framework to find various process flows within the UAV system based on this.

An example of the process by which one component among various components of a UAV system creates an impact chain that leads to a future paralyzed situation is shown in Figure 3.

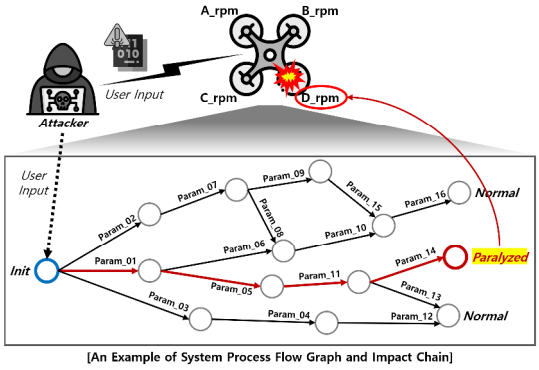

Fig. 3. An example of impact chain in UAV system

The Figure 3 is an example of a situation where an attacker finds a component where they can directly input values and input malicious data and eventually causes problems in the flight operation control of the UAV system, thereby paralyzing the system. In this way, the proposed fuzzing framework can be utilized to secure various components of the UAV system, explore the process flow based on them, and derive the impact chain.

IV. Design and Implementation of Testbed

In this section, we describe a testbed for proposed fuzzing framework. In addition, we introduce an implementing of testbed for testing UAV systems in simulation environment that can perform similar to an actual UAV system. For this purpose, we utilize the MAVLink Protocol for configuring the command and control communication environment with the convenient and accessible PX4 Autopilot and Ardupilot based UAV firmware [15].

A. Design and Configuration of Testbed

In order to use the framework proposed in this work, it is a priority to secure a variety of input command sets for testing. To secure the input command dataset, we designed and configured a testbed to generate various commands that can be applied to the UAV system based on the mission dataset with NED-coordinates, as shown in Figure 4.

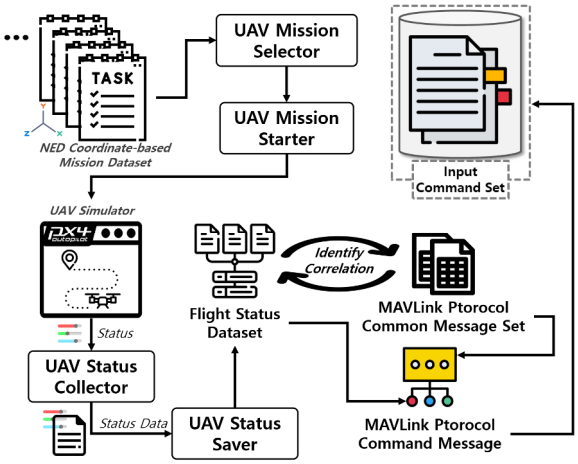

Fig. 4. A design and configuration of testbed for input command set

The process of generating an input command set through this testbed is as follows: First, the UAV simulator is utilized with various mission data based on NED-coordinates to collect status information that occurs during mission execution. Then, the collected status information is utilized to generate a MAVLink Protocol command message, which is then used as an input command set to be used in the framework.

B. Implementaton of Testbed

In this work, we implement a testbed, called archiving system, to obtain UAV testing environment and various flight control related data required by the proposed fuzzing framework. The environments of the implemented system are shown in Table 2.

TABLE II. The Environments of Testbed

Type	Environment
Operating System	Ubuntu Linux 20.04.6 LTS
Platform	Docker PX4-Autopilot jMAVSim (UAV Simulation)
Programming Language	Python 3.8
Library	MAVSDK-Python Pymavlink
Database	MySQL-Server

First, in order to configure the simulation environment, an independent operating environment was configured using the Docker platform within the Ubuntu Linux operating system. This is to simultaneously utilize Software-in-the-Loop Simulation (SITL) and Hardware-in-the-Loop Simulation (HILS) methods [16] and to operate parallel flight mission execution simulation. In addition, the jMAVSIM simulation platform was installed to enable the use of a virtual quadcopter system in order to see how the virtual UAV system operates.

And then, in order to obtain the various components related to flight control, we identified the need for a platform to generate and perform various flight missions, for which we utilized the MAVSDK-Python library and implemented it using the Python language. The implemented flight mission generator utilizes a flight mission dataset, which consists of NED-coordinate based data.

Finally, a database was built using MySQL-Server to collect various flight behavior control related data that occurs during the flight mission, and the data was implemented to be

collected using the Pymavlink library. An example of the operation of the testbed (i.e. archiving system) implemented for testing UAV system and collecting various flight behavior related data is shown in Figure 6.

Fig. 5. An example of an implemented testbed

The Figure 6 is explained as follows: ① shows the UAV system operating environment in the SITL simulation environment, and ② shows the UAV system operating environment in the HILS simulation environment. ③ and ④ shows the platform for creating and executing flight missions utilizing the MAVSDK-Python library, which works with the same code in SITL and HILS. Finally, ⑤ and ⑥ shows the platform that utilizes the Pymavlink library to collect data related to the control of various flight behaviors that occur during flight missions, which is stored in a database built using MySQL-Server.

Some of the flight status data obtained using the testbed built in this work is shown in Table 3.

TABLE III. A PART OF FLIGHT STATUS DATA BY TESTBED

Type of Flight Status Data	Details
ATTITUDE	Information about the pose of the aerial frame
LOCAL_POSITION_NED	Local position information for filtered flight vehicles
ATTITUDE_TARGET	Current commanded attitude information for the flight vehicle specified by the autopilot
GPS_RAW_INT	Global Position information returned by the Global Positioning System (GPS)
SERVO_OUTPUT_RAW	RAW output value information of the flight vehicle's wing motors

By utilizing the various flight state data that can be secured through the testbed constructed in this work, it is possible to generate various input command sets required for the proposed fuzzing framework and use them to derive an impact chain that can be compared with an actual UAV system.

V. CONCLUSION

In this paper, we propose a Cyber-Physical Fuzzing Framework that can be utilized to discover vulnerabilities to neutralize UAV systems in the future. However, in order to derive an impact chain using this framework, it is necessary to accurately extract a valid impact vector and secure various data sets to conduct comparative analysis. In addition, it is necessary to additionally derive various system process elements existing in the UAV system as well as flight motion control related elements to secure a specific data set. We need to validate the framework by generating more diverse data to generate impact vectors and analyzing and evaluating the correlation with UAV systems. To address this, in the future, we plan to fully implement the testbed for proposed framework, secure various UAV system components dataset, and derive valid impact vectors. In addition, we plan to analyze and verify the influence of impact vectors and derive an impact chains suitable for UAV system neutralization situations.

ACKNOWLEDGMENT

This work was supported by Korea Research Institute for Defense Technology Planning and Advancement (KRIT) - Grant funded by Defense Acquisition Program Administration (DAPA) (KRIT-CT-22-051).

REFERENCES

[1] M. Ghamari, P. Rangel, M. Mehrubeoglu, G. S. Tewolde and R. S. Sherratt, "Unmanned Aerial Vehicle Communications for Civil Applications: A Review," IEEE Access, vol. 10, pp. 102492-102531, September 2022.

[2] S. A. H. Mohsan, M. A. Khan, F. Noor, I. Ullah and M. H. Alsharif, "Towards the Unmanned Aerial Vehicles (UAVs): A Comprehensive Review," Drones, vol. 6, no. 6, article no. 147, June 2022.

[3] Pixhawk Project, https://pixhawk.org

[4] R. H. Jacobsen and A. Marandi, "Security Threats Analysis of the Unmanned Aerial Vehicle System," IEEE Military Communications Conference (MILCOM), San Diego, CA, USA, 2021, pp. 316-322.

[5] Z. Yu et al., "Cybersecurity of Unmanned Aerial Vehicles: A Survey," IEEE Aerospace and Electronic Systems Magazin, vol. 39, no. 9, pp. 182-215, September 2023.

[6] Y. Alghamdi, A. Munir and H. M. La, "Architecture, Classification, and Applications of Contemporary Unmanned Aerial Vehicles," IEEE Consumer Electronics Magazine, vol. 10, no. 6, pp. 9-20, March 2021.

[7] V. R. S. Ezhil et al., "Investigation on PID controller usage on Unmanned Aerial Vehicle for stability control," Materials Today: Proceedigs, vol. 66, no. 3, pp. 1313-1318, May 2022.

[8] E. N. Lorenz, "Deterministic Nonperiodic Flow," Journal of the Atmospheric Sciences, vol. 20, no. 2, pp. 130-141, March 1963.

[9] E. Ferrara, "The Butterfly Effect in artificial intelligence systems: Implications for AI bias and fairness," Machine Learning with Applications, vol. 15, article no. 100525, March 2024.

[10] Y. Kim, K. Cho and S. Kim, "Challenges in Dynamic Analysis of Drone Firmware and Its Solutions," IEEE Access, vol. 12, pp. 106593-106604, July 2024.

[11] N. Schiller et al., "Drone Security and the Mysterious Case of DJI's DroneID," The Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, 2023.

[12] J. Jang, M. Cho, J. Kim, D. Kim and Y. Kim, "Paralyzing Drones via EMI Signal Injection on Sensory Communication Channels," The Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, 2023.

[13] J. Jeong, D. Kim, J. Jang, J. Noh, C. Song and Y. Kim, "Un-Rocking Drones: Foundations of Acoustic Injection Attacks and Recovery Thereof," The Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, 2023.

[14] T. Kim et al., "From Control Model to Program: Investigating Robotic Aerial Vehicle Accidents with MAYDAY," 29th USENIX Security Symposium, Online, 2020.

[15] A. Allouch et al., "MAVSec: Securing the MAVLink Protocol for Ardupilot/PX4 Unmanned Aerial Systems," 15th International Wireless Communications & Mobile Computing Conference (IWCMC), Tangier, Morocco, 2019, pp. 621-628.

[16] K. D. Nguyen and C. Ha, "Development of hardware-in-the-loop simulation based on gazebo and pixhawk for unmanned aerial vehicles," International Journal of Aeronautical and Space Sciences, vol. 19, pp. 238-249, no. 1, March 2018.