UAS Cyber Security and Safety Literature Review

March 23, 2022

v 1.1

NOTICE

LEGAL DISCLAIMER

The information provided herein may include content supplied by third parties. Although the data and information contained herein has been produced or processed from sources believed to be reliable, the Federal Aviation Administration makes no warranty, expressed or implied, regarding the accuracy, adequacy, completeness, legality, reliability or usefulness of any information, conclusions or recommendations provided herein. Distribution of the information contained herein does not constitute an endorsement or warranty of the data or information provided herein by the Federal Aviation Administration or the U.S. Department of Transportation. Neither the Federal Aviation Administration nor the U.S. Department of Transportation shall be held liable for any improper or incorrect use of the information contained herein and assumes no responsibility for anyone's use of the information. The Federal Aviation Administration and U.S. Department of Transportation shall not be liable for any claim for any loss, harm, or other damages arising from access to or use of data or information, including without limitation any direct, indirect, incidental, exemplary, special or consequential damages, even if advised of the possibility of such damages. The Federal Aviation Administration shall not be liable to anyone for any decision made or action taken, or not taken, in reliance on the information contained herein.

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. Enter the report number assigned by the sponsoring agency.	**2. Government Accession No.**	**3. Recipient's Catalog No.**
4. Title and Subtitle UAS Cyber Security and Safety Literature Review		**5. Report Date** November 23, 2021 **6. Performing Organization Code** ASSURE: Oregon State University, New Mexico State University and University of North Dakota.
7. Author(s) Houssam Abbas, PhD; Julie A. Adams, PhD; Rakesh B. Bobba, PhD; Yeongjin Jang, PhD; Henry M. Cathey Jr.; Satyajayant Misra, PhD; Roopa Viswanathan, PhD; Amanda Brandt, PhD; Naima Kaabouch, PhD; Paul Snyder; Joe Vacek, JD; David Hertz; Gay Lenzo; Joseph Millette; Kelly Maynard; Mohammad Ababneh; Jinhong Choi; Matthew Jansen; Sharad Shrestha;		**8. Performing Organization Report No.** Enter any/all unique alphanumeric report numbers assigned by the performing organization, if applicable.
9. Performing Organization Name and Address Oregon State University A312 Kerr Admin Bldg. Corvallis, OR 97331 New Mexico State University P.O. Box 30002 Las Cruces, NM 88003-8002 University of North Dakota, 3980 Campus Rd Grand Rapids, ND 58202		**10. Work Unit No.** **11. Contract or Grant No.** 15-C-UAS
12. Sponsoring Agency Name and Address U.S. Department of Transportation Federal Aviation Administration Air Traffic Organization Operations Planning Office of Aviation Research and Development Washington, DC 20591		**13. Type of Report and Period Covered** Literature Review Report **14. Sponsoring Agency Code**

15. Supplementary Notes

Conducted in cooperation with the U.S. Department of Transportation, Federal Highway Administration. This report can be found at Enter project URL.

Enter DOI

Enter Recommended Citation

Enter information not included elsewhere, such as translation of (or by), report supersedes, old edition number, alternate title (e.g. project name), hypertext links to documents or related information in the form of URLs, PURLs (preferred over URLs - https://purl.org/docs/index.html), DOIs (http://www.doi.org), insertion of QR codes, copyright or disclaimer statements, etc. Edit boilerplate FHWA statement above if needed.

16. Abstract

The ASSURE A38 team conducted a literature review to understand the risk and impact of cybersecurity for UAS and their integration into NAS. This involved searching relevant technical academic and no-academic databases to identify relevant papers and documents form the last 10 years and reviewing them to identify cybersecurity threats to UAS and the risks associated with integrating them into NAS. This project sets the stage for follow-on ASSURE projects to better assess the ease of realizing the threats identified in this work and better estimate their success and likelihood, and consequently provide more concrete guidance on the impact of integrating UAS into the NAS. The findings from this project lay the foundation to streamline and accelerate secure, safe and efficient integration of unmanned aircraft into the National Airspace System (NAS).

17. Key Words	**18. Distribution Statement**
Cybersecurity, Attacks, UAS, NAS	No restrictions.

19. Security Classification (of this report)	**20. Security Classification (of this page)**	**21. No. of Pages**	**22. Price**
Unclassified	Unclassified	84	

Form DOT F 1700.7 (8-72)	Reproduction of completed page authorized

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

TABLE OF ACRONYMS

3G/4G	Third/Fourth Generation
ACM	The Association for Computing Machinery
ADS-B	Automatic Dependent Surveillance-Broadcast
AES	Advanced Encryption Standard
AF	Audible Frequency
AHRS	Attitude and Heading Reference System
AIAA	The American Institute of Aeronautics and Astronautics
AIRT	Airborne International Response Team
API	Application Programming Interface
ASSURE	Alliance for System Safety of UAS through Research Excellence
ATC	Air Traffic Control
AV	Antivirus
BVLOS	Beyond Visual Line of Sight
C2	Command and Control
CISA	Cybersecurity and Infrastructure Security Agency
COTS	Commercial-Off-The-Shelf
CPS	Cyber-Physical System
CWE	Common Weakness Enumeration
DAA	Detect and Avoid
DDoS	Distributed Denial-of-Service
DoS	Denial-of-Service
ECDSA	Elliptic Curve Digital Signature Algorithm
ECIES	Elliptic Curve Integrated Encryption Scheme
EMS	Emergency Medical Services
EUROCONTROL	The European Organisation for the Safety of Air Navigation
FAA	Federal Aviation Administration
FANET	Flying Ad-Hoc Network
FAR	Federal Aviation Regulations
FOD	Foreign Object Debris
FTP	File Transfer Protocol
GCS	Ground Control Station
GHz	Gigahertz
GPS	Global Positioning System
HMAC	Hashed Message Authentication Code
HW	Hardware
IDS	Intrusion Detection System
IEEE	The Institute of Electrical and Electronics Engineers
IFR	Instrument Flight Rules
ILS	Instrument Landing System
IMU	Inertial Measurement Unit
ISG	Interagency Supply Chain Group
IoD	Internet of Drones
IoT	Internet of Things

LTE	Long-Term Evolution
LiDAR	Light Detection and Ranging
MAC	Media Access Control / Message Authentication Code
MAVLink	Micro Air Vehicle Link
MEMS	Micro-Electromechanical System
NAS	National Airspace System
NAVAID	Navigational Aid
NIST	National Institute of Standards and Technology
NMSU	New Mexico State University
NextGen	Next Generation Air Transportation System
OS	Operating System
OrSU	Oregon State University
PKI	Public Key Infrastructure
PitM	Person-in-the-Middle
RC4	Rivest Cipher 4
RC	Remote Controlled
RF	Radio Frequency
ROS	Robot Operating System
RSA	Rivest-Shamir-Adleman (public-key cryptosystem)
SATCOM	Satellite Communications
SDR	Software Defined Radio
SMS	Safety Management System
SP	Special Publication
SSH	Secure Shell
SSID	Service Set Identifier
SW	Software
TCP	Transmission Control Protocol
UAS	Unmanned Aircraft System
UAV	Unmanned Aerial Vehicle
UDP	User Datagram Protocol
UND	University of North Dakota
WPA2	Wi-Fi Protected Access 2
sUAS	Small Unmanned Aerial System

EXECUTIVE SUMMARY

Next generation air traffic control systems, such as NextGen, will rely on digital systems making them vulnerable to rapidly evolving cyber threats from both internal and external sources. Recognizing the need for cybersecurity, the Federal Aviation Administration (FAA) has initiated steps to develop a comprehensive and strategic cybersecurity framework for FAA operations in the National Air Space (NAS). However, there are no agency guidelines or frameworks for dealing with the potential cybersecurity and safety risks from Unmanned Aircraft Systems (UAS) or related systems as they are integrated into the NAS. As the small UAS model fleet is projected to grow to more than 2.4 million over the next few years, a cross-organization UAS cybersecurity risk management to complement FAA's efforts for securing NAS is needed. Before such a framework can be developed, a basic understanding of cybersecurity threats to UAS and the impact of their integration to NAS needs to be developed. This project focuses on conducting a literature review to establish baseline information to inform the FAA's approach to cybersecurity issues for UAS and UAS integration into the NAS.

The research team, comprised of Oregon State University (OrSU), University of North Dakota (UND), and New Mexico State University (NMSU), conducted a literature review to understand the risk and impact of cybersecurity for UAS and their integration into NAS. This involved searching relevant technical academic and non-academic databases to identify relevant papers and documents from the last 10 years and reviewing them to identify cybersecurity threats to UAS and the risks associated with integrating them into NAS. It is noted that FAA review of the findings does not constitute an endorsement by the FAA.

Through a detailed review of nearly 550 academic articles, the team identified 41 potential cybersecurity threats to UAS and categorized them into five groups corresponding to the five main components in a UAS ecosystem, namely, UAS hardware (including sensors), UAS software (includes firmware), Network, Ground Control Station (GCS) and Cloud/Server backend (for Internet connected UAS). While the primary objective of this project was literature review for identifying cyber security threats, the team also reviewed existing UAS platforms and use cases. Building on previous ASSURE activities, the team identified 160 commercially available UAS platforms, and also reviewed major hardware and software components used in UAS build kits. Further, the team identified more than 128 UAS use cases across 22 industries. For assessing the cybersecurity threats to these use cases, the team organized them into eight categories using three attributes: autonomy, operational range, and UAS collaboration. They then identified relevant cyber threats to these use case groups. Finally, the team also took the first steps towards cyber risk assessment by performing a preliminary risk assessment for each phase of UAS operation from the 41 identified potential cybersecurity threats using FAA's Safety Management System (SMS) framework. The project also identified mitigative measures against the identified threats through a preliminary review of NIST standards.

This project sets the stage for follow-on projects to better assess the ease of realizing the threats identified in this work and better estimate their success and likelihood, and consequently provide more concrete guidance on the impact of integrating UAS into the NAS. The findings from this project lay the foundation to streamline and accelerate secure, safe, and efficient integration of unmanned aircraft into the NAS.

1. INTRODUCTION AND BACKGROUND

The FAA manages air traffic control through a complex network of information systems and air traffic control facilities. The FAA is currently modernizing its air traffic control operations through the implementation of the Next Generation Air Transportation System (NextGen) that includes digital communications between controllers and pilots—known as DataComm—and other technologies including satellite-based systems for tracking and managing aircraft. Given this increased reliance on digital systems, rapidly evolving cyber threats from both internal and external sources could threaten the connectivity and operations of an increasingly complex aviation infrastructure. Recognizing the need for a cybersecurity strategy and a plan to address the emerging and evolving cyber threats to NAS, the FAA has initiated steps to develop a comprehensive and strategic cybersecurity framework for FAA's operations and NAS. However, currently, there are no agency guidelines that provide a framework or direction on how to properly assess, identify, and mitigate cybersecurity or safety risks specifically for UAS or related systems as they are integrated into the NAS.

This is a critical gap as the FAA Strategic Plan (2019-2022) forecasts that small UAS (less than 55 lbs) model fleet will more than double in size over the next five years from 1.1 million to over 2.4 million. It also projects that by 2022, small UAS non-model fleet will likely grow to over 450K from the current ~100K units. These increases would lead to a need for significant communication and coordination, and consequently would expose them to significant cyber threat risks. There is a need to develop a guide or framework that will establish cross-organization UAS cybersecurity risk management and complement FAA's efforts for securing NAS. To establish such a framework or guide a basic understanding of cybersecurity threats to UAS and the impact of their integration to NAS needs to be established. This project focuses on conducting a literature review to establish baseline information to inform the FAA's approach to cybersecurity issues for UAS and UAS integration into the NAS.

1.1 Scope

This effort considered the following questions that defined the scope for this literature survey:

> **Question 1:** What are the common use-cases and operations scenarios for small UAS (sUAS) (<55lbs; Group 1 and Group 2)?
>
> **Question 2:** What are the common sUAS platforms? This covers cybersecurity relevant aspects such as computing hardware, software, communication and coordination protocols, actuators etc.
>
> **Question 3:** What are cybersecurity threats and issues related to sUAS; and what is there on NAS?
>
> **Question 4:** What sUAS applications are impacted by cybersecurity threats and what agencies deploy those applications/sUAS fleets?
>
> **Question 5:** What mitigation strategies against the cybersecurity threats have been proposed in the literature?

To ensure that the survey could be completed within the requested time frame, the scope was bounded by the following parameters:

1. The survey, literature review, and analysis were limited to literature published within the last 10 years.
2. The effort focused on small UAS (<55lbs; Group 1 and Group 2), as they are likely to be the most commonly used for commercial purposes. Hereafter sUAS and UAS will be used interchangeably.

1.2 Objectives

The overall objective of this literature survey is to support the establishment of a baseline model to identify and assess cybersecurity related risks of integrating UAS into NAS and undertaking a survey of strategies for managing such risks. Specifically:

- Characterize the cybersecurity threat landscape around sUAS and their integration into the NAS through a survey of relevant academic and non-academic literature
- Survey available strategies for managing evolving cybersecurity risks and their relevance to sUAS

1.3 Sub-Tasks

To meet the objectives, the literature survey is divided into the following subtasks:

1. Review use cases and operations of UAS
2. Survey common UAS platforms
3. Identify and review academic and non-academic literature on cybersecurity issues related to UAS and their impact on NAS
4. Identify UAS applications impacted by cybersecurity risks and the agencies deploying the applications
5. Survey mitigation strategies available to counter the discovered cybersecurity risks

The rest of this report describes how these tasks were carried out and discusses the key findings.

2 UAS USE CASES AND OPERATIONS REVIEW

The goal of this sub-task is to identify and understand common use cases so the impact of cybersecurity threats and risks can be better understood. This sub-task involved surveying and identifying some common use cases and operations of sUAS to understand the scope of their deployments and integration into NAS. This team built on use case lists compiled in previous ASSURE activities, specifically A2, A18, and A19, and expanded them through a brief literature survey. It should be noted that the goal is not the creation of a comprehensive list of use cases, but to capture some common ones to be able to better understand the impact of cybersecurity threats and risks. The team then surveyed literature (including industrial whitepapers) to identify emerging use cases.

2.1 ASSURE Tasks A2, A18, and A19 Use Case Information

Under the ASSURE A2 task, "Small UAS Detect and Avoid Requirements Necessary for Limited Beyond Visual Line of Sight (BVLOS) Operations," an extensive assessment of potential sUAS use cases was developed. These were documented by VanHoudt in a report titled, "FAA Interim Technical Report: Small Unmanned Aircraft Systems Use Cases and Detect and Avoid Approaches" [1] and were later revised, updated, and included in Askelson's project report titled, "Small UAS Detect and Avoid Requirements Necessary for Limited Beyond Visual Line of Sight (BVLOS) Operations" [65]. These two reports captured sUAS use cases as well as approaches for Detect and Avoid (DAA) aircraft in the immediate operating airspace.

The desire was to understand the current sUAS use cases to better grasp the potential BVLOS use cases that can be enabled in the future. The DAA and BVLOS applications are not germane to this current effort, but the categorization of use cases is applicable. Use case information was gathered through public and industry data calls, and review of all 333 exemption docket data (summary information from more than 5,000 exemption holders was included in these reports). Twelve general use areas were identified. Short descriptions of each are as follows:

- **Aerial Data Collection**: Use cases that are either described simply as "Aerial Data Collection" (or having a very similar description) or can most accurately be described as a use involving the collection of data by means of sensors or cameras on-board of the sUAS. Separate from the definitions of "Aerial Surveying / Mapping," "Agriculture," "Inspection," and "Research," the description given of the use case is not necessarily specific as to what data is collected and what purposes the data will be used for.
- **Aerial Photography/Videography**: Use cases that are either described simply as "Aerial Photography/Videography" (or having a very similar description) or can most accurately be described as a use involving the collection of pictures and videos for no other obvious or implied reason than to have the pictures or videos taken in the applications listed below.
- **Aerial Surveying/Mapping**: Use cases that are either described simply as "Aerial Surveying/Mapping" (or having a very similar description) or can most accurately be described as a mapping or surveying operation for various purposes.
- **Agriculture**: Use cases that are either described simply as "Agriculture" (or having a very similar description) or can most accurately be described as a use involving the collection of data for agricultural purposes.
- **Emergency Services**: Use cases which are either described simply as "Emergency Services" (or having a very similar description) or describe a use case that can be described as aiding police

officers, firefighters, medical services, etc. or in the investigation of areas that are too dangerous to put a human being for investigative purposes.

- **Flight Training/Education**: Use cases which are either described simply as "Flight Training," "Education" (or having a very similar description), or describe a use case involving the training employees, students, or other users in the operation of sUAS technology, and/or procedures. Use cases involved in educating individuals on sUAS principles, or in demonstrating concepts in mathematics and sciences which can be demonstrated by sUAS technology.
- **Inspection**: Use cases that are either described simply as "Inspection" (or having a very similar description) or that describe a use case involving the inspection of different kinds of structures or areas for safety, upkeep, maintenance, etc.
- **Marketing**: Use cases that are either described simply as "Marketing" (or having a very similar description) or describe the capture of aerial images and videos for the express purpose of using these images and videos for the marketing of a business, product, or service.
- **Multiple Applications**: Use cases which are either described simply as "Multiple Applications" (or having a very similar description) or have been cleared for more than one general use case.
- **Research**: Use cases which are either described simply as "Research" (or having a very similar description) or describe a use involving imaging and data collection distinctly for scientific research purposes.
- **Search/Rescue**: Use cases that are either described simply as "Search / Rescue," or describe a scenario where a sUAS platform would be used to aid in various search and rescue operations.
- **Surveillance, Monitoring, etc.**: Use cases that are either described simply as "Surveillance," "Monitoring," or having a description that can be categorized in a similar fashion.

Each of these use cases were further broken down into subcategories to allow additional definition. From the data collected, Aerial Photography/Videography had the most use cases by 333-exemption holders, with 13,262 use cases granted between September 2014 and June 29, 2016. The other most common general use cases included Inspection (7596), Aerial Surveying / Mapping (4116), Flight Training/Education (2399), and Search/Rescue (1917). Researchers also collected information related to vehicle types, manufacturer metrics, and any applicable DAA related information.

Under the ASSURE A18 task, the work from A2 was expanded and reported in "Development of an Operational Framework for Small UAS Beyond Visual Line of Sight (BVLOS) Operations—New Use Cases, Industry Focus, and Framework Expansion" [12]. The use cases gathered in A18 were focused on advancements or operations within the previous approximately two years and attempted to capture the growth and expansion stage for the industry. Most use cases had adjusted and abided by the FAA Part 107 rules which limit altitude operations to 400 ft. The previously developed use case taxonomy was used as an initial basis for categorization of the different types of flights/missions. This taxonomy was slightly revised to include twelve distinct categories with a catch all "other" category added.

Specific representative use cases (with references) were detailed in each of these twelve categories and 47 subcategories. It was clear from assessing the user operations that many applications do not fall cleanly within any one set of particular categorization lines. The applications and use cases often cover multiple areas during one mission. The key applications using UAS include survey/mapping, imaging, environmental monitoring, patrol/security, disaster response, precision agriculture, and reconnaissance/surveillance/intelligence. Almost all use multiple elements, and many are being fueled by better detector/sensor systems, improved data handling, and Artificial Intelligence.

The focus of the A2 and A18 efforts was to gather and assess use cases in relation to potential BVLOS operations and the application of DAA technologies. While DAA and BVLOS are not the focus of this

current work, the breadth of potential sUAS operations and use cases is applicable. Table 1 is an extracted summary from this previous work of the application areas and use case applications.

Of interest toward the data collection process, ASSURE Task A19, "UAS Test Data Collection and Analysis" was to develop a safety case framework that supports UAS integration safety cases. A safety case and an associated data schema were developed. The approach taken was to understand the data for each phase of the framework consists of three primary factors: 1) identify the sources of data, 2) describe the data components of each phase, and 3) define the context for each phase of the framework. The safety case development process was defined in four steps:

1 Operational Context Definition
2 Data Collection
3 Safety Case
4 FAA Approval

While the overall safety case process is not applicable to this cyber and security task, the extensive listing of the metadata [64] collected is important. This provides a full listing of operational parameters that can be collected related to the flight mission. This listing can be reviewed for where vulnerabilities can be exploited by blocking information or in providing incorrect information. This data listing has been provided to the A38 team for review.

Table 1. sUAS Use Cases Extracted from A18.

Application Area	Application
Aerial Data Collection	Aerial Data Collection - Construction/Mining
	Aerial Data Collection - Environmental
	Aerial Data Collection - General
	Aerial Data Collection - Insurance
Aerial Photography/ Videography	Aerial Photography/Videography - Closed-set filming
	Aerial Photography/Videography - Construction
	Aerial Photography/Videography - General
	Aerial Photography/Videography - News-Gathering
	Aerial Photography/Videography - Outdoor Activities
	Aerial Photography/Videography - Real Estate
	Aerial Photography/Videography - Wedding
Aerial Surveying/ Mapping	Aerial Surveying/Mapping - Agriculture/Mining
	Aerial Surveying/Mapping - Construction
	Aerial Surveying/Mapping - Engineering
	Aerial Surveying/Mapping - General
Agriculture	Agriculture - Crop Monitoring
	Agriculture - General
	Agriculture - Precision Agriculture
Emergency Services	Emergency Services - Crisis Response
	Emergency Services - General
	Emergency Services - Investigate Hazardous Regions
Flight Training/ Education	Flight Training/Education - Education
	Flight Training/Education - General
	Flight Training/Education - sUAS Training
Inspection	Inspection - Communications Structures
	Inspection - Construction
	Inspection - General
	Inspection - Insurance
	Inspection - Oil/Pipeline
	Inspection - Power plants
	Inspection - Real Estate
	Inspection - Structure
	Inspection - Wind power
Marketing	Marketing - Aerial Images
	Marketing - General
Multiple Applications	
Research	Research - Academics
	Research - Development
	Research - General
	Research - Market
	Research - Operations
	Research - Product Testing
	Research - Transportation
Search/Rescue	
Surveillance, Monitoring, etc.	Monitoring - Environmental
	Monitoring - General
	Monitoring - Legal
	Monitoring - Safety
	Monitoring - Security
Other UAS Applications	Novel or unique use cases

2.2 Other ASSURE Tasks

There are a number of other ASSURE research tasks that will be exploring and expanding specific use cases. While the expected additions will be more detailed permutations of existing applications, it is worthwhile to note these other ASSURE tasks for potential future exploration.

Task A31, "Safety Risks and Mitigations for UAS Operations On and Around Airports" will define the overall concept and specific use cases for conducting operations on the airport surface. This includes but is not limited to:

- UAS airport inspections (buildings, infrastructure, etc.)
- Ground operations (ex. transitioning across airport grounds)
- Perimeter security
- Foreign Object Debris (FOD) inspections
- Taxiway and Runway inspections
- Emergency response
- Wake Turbulence Separation
- Large UAS takeoff and recovery

This task will consider the airspace class (B, C, D, E, G), towered/non-towered etc. for each applicable representative use case.

Task A28, "Disaster Preparedness and Response" has a vision to develop a safe, effective, and standardized approach to enhance disaster recovery and emergency response using UAS. One of the specific tasks is to "Survey of Experts for Disaster Preparedness and Response Use Case Development." Additional use cases will be captured in this effort. There are other ASSURE research tasks that may also add to the use case listings. The recent FAA Remote ID rule could also have impacts to the cyber security review as well.

2.3 Expanding Use Cases

It is clear that definitive categorization of "all use cases" is impossible due to the evolving nature of the vehicles, sensor/support systems, and potential user applications. It has been stated before that the applications and uses of UAS are limited by the creativity of the proponents. Many of the individually defined use cases are in essence different applications of common use case models. For example, "package delivery" could be for commerce (e.g., purchased goods, warehouse operations, etc.), medical (e.g., testing supplies, test materials, organs for transplant, etc.), test materials, repair parts, transport of specialized payloads (e.g., semen for artificial cow insemination), emergency (e.g., defibrillators, warm clothing, or food/water), and much more. Image and video capture are common across many different use cases from construction site management, disaster response, insurance claim validation, real estate, research, and much more.

The use case listings change over time with a focus on the specific of the applications. Thompson noted and highlighted a few use cases years ago in "25 Commercial Drone Use Cases" [103], which included the following:

1) Insurance Claim Validation
2) Wind Turbine Inspection
3) Construction Site Management
4) Agriculture
5) Live Gas Flare Inspection

6) First Aid
7) Security
8) Flash Flood Warning
9) Organ Transplant Delivery
10) Preventing Shark Attacks
11) Wildlife Conservation
12) Railway Safety
13) Shipping Emission Monitoring
14) Reforestation
15) Cinematography
16) Pipeline Leak Detection
17) Cargo Delivery
18) Journalism
19) Search and Rescue
20) Oil Spill Monitoring

In 2020, Pozner prepared "A Comprehensive List of Commercial Drone Use Cases (128+ And Growing)" [87]. His focus was on commercial use cases. He stated, "In broad terms there are seven (7) buckets of commercial use cases for drones:

1. remove people from dangerous work;
2. reduce the number of people needed;
3. reduce the number of steps in the process;
4. replace more costly methods;
5. access inaccessible (by humans) locations;
6. perform tasks quicker or more efficiently;
7. and, perform functions people do not want to perform / not strong enough labor pool.

For his assessment he uses these areas above as the start of his categorization. "Under these seven buckets we can see a plethora of industries and over 150 use cases," and he explores "the top twenty-two (22) industries that would benefit, in the short term, or could craft a compelling case for commercial drone use." His industry list and use cases are presented below.

Food / Restaurant Industry
1. Food Delivery
2. Convenience Store / Grocery Delivery
3. Food and Beverage Service (i.e at pools, on golf courses)
4. Drone Waiter

Hospitality & Tourism
5. Mobile Hotels
6. Food and Beverage Preparation
7. Entertainment / Activity
8. Security
9. Property Maintenance
10. Visual Marketing
11. Life-guarding
12. Transportation of Materials

Healthcare

13. Medication / Prescription Delivery
14. Blood Donation Delivery
15. Laboratory Sample Collection and Delivery
16. Vaccine Storage and Delivery
17. Organ Transport
18. Ambulance Drone

Emergency Response

19. Search and Rescue (Infrared and Visuals)
20. Equipment Transport
21. Inspect and Explore Disaster Areas (Indoor, Outdoor, and confined spaces)

Humanitarian and Disaster Relief

22. Damage and Infrastructure Assessment
23. Restoration of Vital Services (Power, Phone, Wifi)
24. Predict and Access Natural Disasters and Effected Areas
25. Monitor and Combat Natural Disasters (Forest Fires)
26. Distribute Food and Water
27. Create 3D Models of the aftermath

Disease Control

28. Pest Control / Collection
29. Pollution Monitoring and Control
30. Disease Tracking and Monitoring

Retail

31. Product Delivery
32. Product Organization, Storage, and Inventory

Advertising / Visual / News

33. Cinematography
34. Videography
35. Photography
36. Advertising
37. Promotional Item Delivery
38. News Coverage

Sports and Entertainment

39. Synchronized Light Shows
40. Floating Projection Screens
41. Drone Puppeteers
42. Drone Racing
43. Drone Combat
44. Broadcasting Sports
45. Instant Replay / Officiating Assistance

Agriculture

46. Predict and Analyze Crop Growth
47. Provide Aerial Views
48. Pest Detection and Control
49. Warning and Remedy of Crop Failure
50. Perform Manual Redundant Tasks (i.e. seeding, planting, and spraying)

Weather Forecasting

51. Follow Weather Patterns
52. Explore, Document, and Predict Severe Weather

53. Severe Weather Warnings
54. Gather Data in Inhospitable or Extreme Locations (i.e. ocean depths, high atmosphere)

Conservation

55. Monitor and Track Animals
56. Combat poachers
57. Collect Samples
58. Research Ecosystems

Shipping

59. Safety and Compliance Inspections
60. Detect Emission Infractions and Identify Offenders
61. Navigational Aids
62. Search and Rescue
63. Autonomous Shipping

Construction

64. Monitor Building Progress
65. Topographic Mapping and Analysis
66. Soil Analysis
67. Surveying and Digital Mapping
68. Inspections
69. Physical Construction
70. 3D Renderings

Real Estate

71. Photography and Videography (Exterior and Interior)
72. 3D Renderings
73. Infrared Analysis
74. Property Tours
75. Showcase and Suggestion of Amenities, Additions, or Additional Structures

Insurance

76. Inspection Of Claims
77. Fraud Detection / Prevention
78. Natural Disaster Monitoring and Modeling
79. Drone Insurance

Energy

80. Infrastructure and Compliance Inspection
81. Operate in Contaminated or Hazardous Areas
82. Leakages and Spread Detection
83. Energy Exploration
84. Buildings and Transmission Efficiency Mapping

Mining and Resource Exploration

85. Exploration
86. Surveying and Mapping
87. Safety Inspections
88. Inventory Management
89. Security
90. Mining Operations

Urban Planning

91. Traffic and Population Studies
92. Terrain, Weather, Water, and Resource change Mapping

93. Traffic Direction
94. City Centers Redesign

Telecommunications / Entertainment

95. Infrastructure and Compliance Inspection
96. Radio Planning and Line-of-Sight Mapping
97. Connectivity

Airlines and Airports

98. Search & Rescue
99. Airport Air Security
100. Infrastructure and Airplane Inspections
101. Flight / Navigation System Testing and Verification.
102. Cargo Delivery
103. Pest Control

Manufacturing and Inventory Management

104. Manufacturing
105. Assembly Lines Inspection
106. Raw Materials Discovery
107. Equipment Transport
108. Inventory Location
109. Inventory Measurement
110. Order Compilation and Inspection

Other Drone Use Cases

111. 3D Renderings
112. Fitness
113. Video Games
114. Security
115. Repair Drones
116. Machine learning service
117. Spray Paint
118. Ultrasonic testing (UT)
119. Dry Film Thickness (DFT)
120. Low- or High-Pressure Cleaning Solutions
121. Firefighting
122. Infrared Thermography
123. Home Delivery (i.e., Dry Cleaned Laundry Delivery)
124. Fishing
125. Film: Wedding, Fireworks, Concerts, Parties, etc.
126. Use a spotlight
127. Carry equipment
128. Indoor drone shows

These are additional examples of how use cases are evolving, have similar performance or deliverables, and how they can be viewed through different lenses such as application, industry, or end products. It is also worth noting that with almost every line item, one can take a deeper dive into the details. An example of this using a quantitative assessment balancing the potential health impact and the potential supply chain impact for prioritization is in "UAVs in Global Health: Use Case Prioritization" by the ISG UAS Coordinating Body [40]. This provided a second level of four use cases/clusters:

1) Delivery in response to medical emergencies

2) "Just in Time" resupply to campaigns
3) "Just in Time" resupply to health clinics
4) 2-way transport of diagnostic samples and treatment

This is presented as an example. Almost every top-level use case listed can be broken down into further details, approaches, and prioritizations. There are further detailed levels that one can break any use case down into whatever set of metrics one desires including flight times, payload capacity, power, etc.

2.4 Use Case Categorization in Relation Operations

Almost all "use case" lists center on the functions unique to the application or industry. Descriptions of the applications tend to focus on the uniqueness of the use cases or the desired end product. The listing is function/product focused. From a review in relation to Cyber Security and Safety, the approach needs to be oriented from the opposite perspective. What is the smallest common baseline set of areas that are potentially subject to compromise? The overall use case taxonomy generated is fine for assessing common markets and approaches. From a cyber security standpoint, in an inspection use case, for example, it generally does not matter what is being inspected. It is the planning, operation, command, control, imaging, data, etc. that are all common elements.

Per the proposed plan, the use-cases and operations of UAS were reviewed to understand the scope of their deployments and what these deployments require in terms of integration with the NAS. An attempt to restructure the use cases to be able to capture vulnerabilities to common potential threats was assessed. Regardless of the specific application, all flight operations have similar functional elements. It is valuable to look at the flight operation in terms of the "muscle movements" for each type of use case.

A flight process for all missions and use cases is presented below. This should serve as a starting point to highlight classes of vulnerabilities and points of vulnerability under the broader use case categories. A network attack, firmware attack, sensor attack, or ground station attack can be independent of the specific use case and may be a function of timing within an operation.

The approach was to map common functions to the use cases. This can serve as a starting point to map specific vulnerabilities to each type of operation and when in the operation they might be applicable. The UAS Phases of Operation and "major muscle movements" are as follows:

UAS Phases of Operation

- Pre-Flight / Mission Planning
    - UAS Selection
    - Payload /Sensor Selection
    - Flight Planning (both for manual and autonomous)
    - Programming flight (autonomous only)
- Preparation /System Checks (applicable at almost all phases of mission/flight)
    - Ground station
    - Flight controls
    - Data links
    - GPS
    - Magnetometer
    - Power – battery/fuel
    - Environment
- Launch
    - System checks (similar to those noted above)

    - Altimeter verification
- Flight
    - Manual
    - Autonomous – Flight plan verification
- Mission/Application
    - Ground Station
    - Data Relay – Telemetry
    - "Payload" data
        - Video relay
        - Sensor information
- Return to Land
    - Manual
    - Autonomous
- Post- Flight
    - Ground Station
    - Data Download

One has to assess potential issues based on the phase of the mission. For example, during flight one has to look at the potential issues with GPS, RF, data, video, sensor, etc.

Segregation of systems at all phases is another attribute to consider based on use case. A ground station connection to the internet – pre/during/post operations--can be an opening. This also holds true for the sensors that require connection to the internet for operation or to download data, and any flight item that connects to the UAV's autopilot (ex. for DAA operations sensor pointing, etc.) Segregated ground stations and sensors do not have these same access points.

This categorization has been done in relation to operations and can be applied to the general use case categories developed under ASSURE Tasks A2 and A18. This can also be done for the industry-based applications presented in Section 4. All 128 use cases could be mapped back to the one presented in Section 2. Deeper dives could be made for each specific use case area.

2.5 Summary

UAS use cases from previous ASSURE Tasks A2 and A18 were reviewed for applicability to the A38 UAS cybersecurity literature review task. Additional sets of use cases were also documented. The overall use case taxonomy generated was appropriate for assessing common markets and approaches, but from a cyber security standpoint, it is common elements related to the planning, operation, command, control, imaging, data, etc. that are the best approach for assessment.

The use cases previously generated were broken down into the flight operation in terms of the "muscle movements" for use cases. This should serve as a starting point to highlight classes of vulnerabilities and points of vulnerability under the broader use case categories. This can serve as a starting point to map specific vulnerabilities to each type of operation and when (timing) in the operation it might be applicable.

3 SURVEY OF COMMON UAS PLATFORMS

3.1 Rationale

A survey of common UAS platforms comprising the current commercially available small UAS market was performed to identify common sUAS platforms, covering hardware, software (including firmware, operating systems, middleware etc.) and communication and coordination protocols, as well as commercially available components used for construction of sUAS (including flight controllers, processors, actuators, etc.). The rationale for this sub-task was to determine specific vulnerabilities of common UAS platforms and UAS modules and observe whether any patterns of cybersecurity vulnerability emerge when searching a representative sample. Any patterns that emerged can inform threat landscape in terms of scope of vulnerability and magnitude of risk.

3.2 Methodology

The methodology initially built on outcomes from ASSURE Tasks A2 and A18. Following the results of those activities, the team surveyed the available UAS platforms on the market, and component modules available to build sUAS. The results informed a framework of categories that would both capture and organize all reasonably available Commercial-Off-The-Shelf (COTS) small UAS.

A usable framework of categories that would both capture and organize all reasonably available COTS sUAS was needed. Several iterations of the framework were tested and reviewed, and the final selection was a spreadsheet version. Excerpts from this spreadsheet are located in the results section, with the entire spreadsheet attached as an external accompanying document (A38-UASPlatforms-Table.xlsx). The framework, spreadsheets, and table entries were developed by the team to create a list of UAV manufacturers and distributors (like DJI, E-flite, Aerovironment, etc.) that were commonly known. With the known company names and distributors, the team searched each company/manufacturer website individually to return results for their off the shelf aircraft to be added to the list. For each individual aircraft, the company website was used to gather as much detail in each of the categories in the COTS table as possible. The team also searched each individual aircraft to check for other aircraft specifications that could potentially conflict with or indicate error in the information on the website. There were no cases of conflicting information, so this last verification step indicated good data.

In addition to the COTS list, the team generated a list of search terms that would yield results finding aircraft manufacturers and distributors that weren't already known and that may not be as well known (e.g. multirotor, fixed wing, RC, commercial, precision ag, etc..) which yielded several more companies and distributors (like Teal Drones and Terraview). The team again scoured their websites and all available information to add each aircraft to the list and additional aircraft information under each of the COTS categories. The results were again verified for conflicting information by searching the company and aircraft name using different search engines. There were no cases of conflicting information.

For the modular component tables, the team started with the assumption that only listing COTS aircraft was too limiting from a cyber security standpoint, as aircraft airframes can easily have various critical components changed or substituted. These modular components can be compromised just as those on the aircraft and those that are part of the ground control station. The modular parts of the aircraft critical to cybersecurity are the autopilot/flight controller, telemetry receiver for the autopilot, GPS, and receiver. The modular parts that are part of the ground control station critical to cybersecurity are the physical controller/transmitter and telemetry transmitter for the autopilot. The team created different sheets for each of these modular parts categories to track manufacturers and distributors and different modular part names.

For each sheet, the process was identical to finding aircraft and specifications above. The team Subject Matter Expert's (SME) first generated a list of manufacturers and companies they already knew and used that to search for and fill in the different sheets with component names. However, for modular components, new categories were created to ensure the necessary specifications critical to C2 were added to the list. The SME's first added the categories they knew were critical to each component and cybersecurity and these were verified with the A38 team cybersecurity experts (processors, wifi enabled, frequency, communications protocol, etc.). The next step was generating a list of relevant search terms for each type of modular component to identify manufacturers and companies that were not already known. More were identified and the components and extra specifications were added. All of this information was verified by searching the specific part names in different search engines to search for any conflicting information and again, there was none.

3.3 Findings

The tabulated results can be found in an accompanying external document (A38-UASPlatforms-Table.xlsx), and excerpts of the most relevant COTS and modular units tables are found in Table 2.

Table 2. COTS UAS Descriptions.

Manufacturer	DJI	DJI	DJI	Eflite	Eflite	Hobbyzone	Hobbyzone
Model	Phantom 4 Pro	Inspire	Inspire 2	UMX Radian BNF with AS3X, 730mm	UMX Night Vapor BNF Basic	Sport Cub S 2	Carbon Cub S 2 1.3m
Weight (lbs/kg)	1388g	6.27 lbs (2845 g, including propellers and battery, without gimbal and camera) 6.74 lbs (3060 g, including propellers, battery and Zenmuse X3)	Weight: 7.58 lbs (3440 g, including propellers and two batteries, without gimbal and camera) Max Takeoff Weight: 9.37lbs (4250 g)	1.50 oz (43 g) w/150mAh 1S 3.7V 25C	0.9 oz (25g) w/150mAh 1S 3.7V 25C 0.7 oz (21g) w/o battery	2 oz (57g)	2.3 lbs (1075g)
Endurance (mins) (Battery/Flight Conditions as applicable)	Approx 30 minutes	Approx. 27min (with Zenmuse X4S) Approx. 23min (with Zenmuse X7) (Hovering at sea level with no wind.)	Approx. 27min (with Zenmuse X4S) Approx. 23min (with Zenmuse X7) (Hovering at sea level with no wind.)	10 min with 150mAh 1S 3.7V 25C	~5 mins with 150mAh 1S 3.7V 25C	3 mins with 150mAh 1S 3.7V 25C Li-Po	10 mins with 2200mAh 3S 11.1V
Sensor Hardware	EO camera	DJI sensor connection (varies between EO, thermal, high zoom. Swappable	Swappable, Varies	N/A	N/A	N/A	N/A
Controller Frequencies:							
2.400-2.4835 GHz	x	x	x				
5.725-5.850 GHz	x	x	x				
2.4 GHz				x	x	x	x
Protocols (Bluetooth / Wi-Fi / Cellular (2G/3G/4G/5G) / etc)	DJI Light Bridge	DJI Light Bridge	DJI Light Bridge	Spektrum DSM2/DSMX	Spektrum DSM2/DSMX	Spektrum DSM2/DSMX	Spektrum DSM2/DSMX

Table 3. Modular GPS Descriptions.

	Avionics Anonymous GNSS + Compass	Beitain BN-220 GPS	CUAV C-RTK 9P	Hex Here 2	Holybro Micro M8N GPS Module	Holybro Pixhawk 4 GPS Module	simpleRTK2B V1	CUAV NEO 3 GNSS u-blox M9N GPS Module	Hex HerePro Multi-RTK
GNSS System(s):									
GPS+QZSS/SBAS			x				x	x	x
GPS/QZSS	x	x		x	x	x			
GLONASS	x	x	x	x	x	x	x	x	x
Galileo		x	x	x	x	x	x	x	x
BeiDou		x	x	x	x	x	x	x	x
Anti Jamming:									
Active CW detection and removal	x	x	x	x	x	x	x	x	x
Extra onboard SAW band pass filter	x	x		x	x	x		x	x
Onboard band pass filter	x		x				x		
Spoofing Detection	Built-in	Built-in	Advanced anti-spoofing	Built-in	Built-in	Built-in	Advanced anti-spoofing algorithms	Built-in	Advanced anti-spoofing algorithms
Signal Integrity	Signature feature with SHA 256	Signature feature with SHA 256	N/A	Signature feature with SHA 256	Signature feature with SHA 256	Signature feature with SHA 256	N/A	Signature feature with SHA 256	N/A
Protocols:									
NMEA	x	x	x	x	x	x	x		x
NMEA 4.10								x	
UBX	x	x	x	x	x	x	x	x	x
Binary	x	x	x	x	x	x	x	x	x
RTCM	x	x		x	x	x			
RTCM v. 3.3			x				x	x	x
SPARTN v. 1.8			x				x		x

Table 4. Pixhawk & Non-Pixhawk Controllers.

Modular Pixhawk Flight Controllers

Autopilot/Flight controllers	**3DR Pixhawk 1**	**mRo Pixhawk**	**mRobotics-X2.1 Rev 2**	**Drotek Dropix**	**mRo Pixracer Pro**
Main system-on-chip (MSOC)	180 MHz ARM® Cortex® M4 with single-precision FPU	32-bit STM32F427 Cortex® M4 core with FPU	32-bit STM32F427 Cortex M4 core with FPU	32 bit ARM Cortex® M4 Processor running NuttX RTOS	32 bit Cortex M7 RISC core with FPU 460 MHz
FSOC CPU	24 MHz ARM Cortex M3	32 bit STM32F103	32-bit STM32F103	N/A	N/A
Sensors: **Accelerometer**	ST Micro LSM303D 14 bit	ST Micro LSM303D 3-axis 14-bit	Invensense/TDK ICM-20602 (6DOF)	ST Micro LSM303D 14	Invensense/TDK ICM-20602 (6DOF)
Gyroscope	ST Micro L3GD20H 16 bit	ST Micro L3GD20 3-axis 16-bit		ST Micro L3GD20H 16 bit	Invensense/TDK ICM-20948 (9DOF)
Magnetometer	Invensense MPU 6000 3-axis		Invensense/TDK MPU-9250 (9DOF)	ST Micro LSM303D 14 bit	AK09916 inside ICM-20948
Barometer	MEAS MS5611	MEAS MS5611	MEAS MS5611	MEAS MS5611	DPS310

Non Pixhawk based Flight Controllers

Autopilot/Flight controllers	**Omnibus F4 SD**	**Snapdragon Flight**	**OcPoC-Zynq Mini**	**Holybro Kakute F7**	**Holybro Durandal**	**ModalAI Flight Core v1**
Main system-on-chip (MSOC)	STM32F405RGT6	Snapdragon 801	FPGA+ARM System-on-Chip: Xilinx Zynq Z-7010	2x MPU9250 9-DOF 1x MS5611	STM32H743	STM32F765II
FSOC CPU	168 MHz ARM Cortex M4 with single-precision FPU	Quad-core 2.26 GHz Krait	667 MHz Dual-Core ARM A9	216 MHz ARM Cortex M7 with single-precision FPU	32 Bit Arm ® Cortex® -M7 480MHz	32-bit ARM M7
Sensors: **Accelerometer**	BMP280 Baro	Invensense MPU-9250 9-Axis Sensor	2x MPU9250 9-DOF 1x MS5611	BMP280 Baro	ICM-20689	ICM-20602 ICM-42688
Magnetometer					IST8310	
Barometer		Bosch BMP280			MS5611	BMI088 BMP388

4 LITERATURE REVIEW OF UAS CYBERSECURITY AND IMPACT ON NAS

This sub-task formed the core of the literature review effort with the goal of identifying known cybersecurity threats/risks to UAS, their impact on UAS integration into NAS, and any potential defenses that have been discussed in the literature. At a high-level this involved i) identifying and compiling a corpus of relevant literature published within the last 10 years, ii) analyzing the corpus, and finally iii) summarizing the findings. The process used for identifying, collecting, and reviewing the corpus is described next, followed by a discussion of the findings from three different perspectives.

4.1 Corpus Compilation and Review Methodology

The first step in the literature review process was identifying and compiling a list of relevant literature. The research team started out by identifying a list of initial keywords shown in Table 5 to be used for searching the technical databases.

Table 5. Initial List of Keywords.

UAS Terms	Cybersecurity Terms
Unmanned Aircraft System Unmanned Aerial Vehicle Remotely Piloted Vehicle Optionally Piloted Vehicle Urban Air Mobility	Cyber-security Cyber-physical security Cyber attacks

Keyword pairs were created by combining one keyword from the list of UAS terms and one keyword from the list of cybersecurity terms. Both hyphenated (e.g., cyber-security) and unhyphenated (e.g., cybersecurity) versions were used. Keyword pairs were then used to search technical databases for articles that are of potential relevance to this project. Three key technical databases were selected initially, namely, IEEE, ACM, AIAA. Searching these three databases was deemed sufficient to provide a good snapshot of the relevant literature especially since ACM database also holds metadata on technical articles contained in other technical databases.

The team wrote programs to automatically connect to the technical databases and obtain metadata for technical articles that match the search criteria. Note that the entire paper is searched for matching keyword pairs and not just the metadata. The metadata of the matching article (title, authors, venue, etc) was fetched from the database and stored in a shared team database. Matching articles were then reviewed in phases that progressively narrowed the corpus or articles to identify the most relevant articles. At a high-level, team members performed a quick review of the abstracts of the articles fetched from technical databases to ensure that the article was indeed relevant to the project. All articles deemed relevant were then reviewed in detail (i.e., the full technical article) and synthesized in the next phase of the technical review. The review task was undertaken in a distributed manner among the partnering institutions with the technical articles to be reviewed split between the institutions based on the expertise (e.g., network security, platform security, communication security, aviation, UAS platforms, NAS etc.).

4.1.1 *Keyword Expansion & Database Construction*

Additional relevant keywords were discovered after obtaining the search results from technical databases using the initial keyword list. The team added keyword terms to find articles related to legislation, standards,

and policy related to UAS. Terms related to cyber-security were also broken into single word pieces. The final list of keywords is shown in Table 6.

Table 6. Expanded List of Keywords.

UAS Terms	**Cybersecurity Terms**
Drone Unmanned Aerial Piloted Urban Air Mobility National Air System FAA Federal Aviation Administration	Attack Cyberattack Security Cybersecurity Cyberphysical Cyber-physical Safety

Each query for searching the technical databases comprised a pair of keywords created by combining one keyword from UAS terms and one from cybersecurity terms. Each query was written to match articles in the technical databases (IEEE, ACM, AIAA) that contained both the keywords. As a result, 56 queries (8 keywords from UAS terms and 7 keywords from Cybersecurity) were performed for each database. The team wrote programs to automatically connect to the technical databases and obtain metadata for technical articles that match the search criteria. The search was limited to articles published between 2010 and 2020.

- IEEE: Digital library of IEEE, namely IEEE Xplore, provides users with Application Programming Interface (API). The team wrote a program in Python that leveraged the IEEE API to collect the matching articles successfully. More information about IEEE Xplore API and documentation can be found here: https://developer.ieee.org.
- ACM and AIAA: As no API was available for these databases, the team utilized an open-source framework for web-crawling. Programs were written for each database as the web pages listing the results had different user interfaces and design.

Once articles matching with keywords were fetched, they were processed before saving into the database. As the same article can match multiple queries, duplicate articles needed to be identified and merged. If an article matches several different queries, that is, it has multiple keyword pairs, it indicates that it might be more relevant to our survey. Therefore, during the deduplication process, a score was assigned to each article to show the number of different matched queries (or keyword pairs). The team collected 10278 articles from the digital library of ACM, 8117 from AIAA and 6995 from IEEE after deduplication for a total of 25390 articles.

To streamline the research collaboration among the geographically distributed team members from multiple institutions, the team used an open-source reference management software called Zotero. A private database of articles was created in Zotero cloud that is only accessible to the team members. In addition to capturing metadata such as title, author, year etc., Zotero allows tagging each article with custom tags. The team wrote software to automatically tag each collected article with the keyword pairs the article contains and the number of unique keyword pairs it contains when importing the article into the Zotero database.

4.1.2 *Refining and Reviewing the Corpus*

As more than 25,000 articles matched the search queries, the researchers needed an efficient process for identifying the most relevant articles and reducing the corpus to a size manageable within the tight time constraints of this project. Not all articles matching the keyword search may be relevant as some articles may simply be referring to UAS as motivating examples but without being specifically about UAS security issues. The team used a multi-stage iterative review process: i) reviewing the article abstracts to better identify and categorize relevant articles, and ii) performing a full technical review relevant articles from the previous stage. After each round of review, the tag system of Zotero was utilized to add custom information for each article including the categories that the article covers. The list of categories used are listed in Table 7.

4.1.3 *Abstract Review Stage*

Team members performed a quick categorization of the articles based on a review of the abstract. If the article was deemed relevant, the reviewer would accept it and categorize it into relevant topics by attaching category tags. Three kinds of tags could be assigned in Round 1. First, a reviewer tag, including reviewer's name, was assigned at the start of review to let collaborators know the article was being reviewed and who the reviewer was. Second, a result tag was assigned indicating whether the article was being accepted or rejected, based on the review. Lastly, a reviewer assigned one or more category tags to the article. Note that a single article can have multiple category tags. For example, articles introducing attacks usually also discuss potential countermeasures, leading to both attack and defense tags being associated with the article.

Table 7. Category Names and Tags Used in Survey.

Category	**Category Tags**
UAS Use Case Papers	usecase
UAS Attacks	attacks
UAS Security Defenses	defenses
UAS Platforms (HW/SW)	platforms
National Air Space/Flight Operations/Air Traffic Management --- Overview	nas
UAS Standardization	standardization
UAS Regulations	regulations
Standards for UAS security	security-std
Legal/Policy/Property/Ethical Issues	policy
Major Players (companies, platforms, use cases)	players

The team reviewed 6,833 articles in the abstract review stage. Although the team could not review all 25000+ articles, the team prioritized the abstract review using keyword pair match scores so as not to miss potentially more relevant or important articles. As mentioned earlier, the score of an article was based on the number of matched keyword pairs. Team reviewed articles with the highest scores first. Abstract review stage was deemed complete when all articles with keyword pair match score of 4 or higher were reviewed. At the end of the review stage 1,294 articles were accepted for further review in the next stage. The number of accepted articles at the end of the review stage in each category is shown in Table 8 below.

Table 8. Number of Papers Accepted in each Category after Abstract Review Stage.

Category tags	**Accepted in Round 1**
usecase	233
attacks	164
defenses	321
platforms	235
nas	110
standardization	15
regulations	39
security-std	45
policy	28
players	27
no category tags	249

4.1.4 *Technical Review Stage*

Articles selected for further review in the previous stage were reviewed in detail in this stage. A review process similar to the one in previous stage is followed, where the reviewer name tag (which can be different from the reviewer in the previous round) was added at the beginning of the technical review, with an optional result tag added at the end of the review to highlight the relative importance/relevance of the article for the survey. A summary for each article reviewed was created and shared with the rest of the team. The team reviewed 547 articles in this stage. Table 9 shows the number of reviewed articles for each category in this stage. This stage of review was concluded both due to diminishing return in terms of new information (i.e., new attack categories or defense techniques) and due to the tight timeline of the project. A listing of the papers in each category is available in an accompanying document (A38-LiteratureReview-Library.xlsx).

Table 9. Number of Articles Reviewed in Detail by Category.

Category Tags	**Round 2**

usecase	76
attacks	124
defenses	294
platforms	67
nas	24
standardization	15
regulations	37
security-std	15
policy	28
players	8
no bucket tags	24

4.2 UAS Components

The findings from the survey regarding the threat landscape facing UAS and their integration into NAS are organized and presented in this report from two perspectives: i) organized by UAS components, and ii) organized by UAS operational phases. The team discusses the key UAS components (see Figure 1) before discussing the threats and threat vectors impacting these components. Component wise organization of threats to UAS presents a useful way to understand the threat landscape and the potential impact of such threats and has also been used in multiple prior works. Please note that the researchers use the term Unmanned Aerial Vehicle (UAV) to emphasize that the scope of hardware and software is within UAV, not entire UAS.

4.2.1 *UAV Hardware*

The hardware components of UAV include physical components of the UAV, such as body, propellers, sensors (e.g., GPS, IMU), actuators (e.g., motor), etc. as listed below.

- GPS Transceiver
- Inertial Measurement Unit (IMU): gyroscope, accelerometer, magnetometer
- Optical: Camera, LiDAR, radar
- Processing: processor and memory
- Body
- Actuator
- Payload
- Others: ADS-B transponder, Remote ID module

While the researchers do not focus on physical threats in this work, cyber threats to other hardware components such as the sensors, actuators, and processing elements would be very relevant.

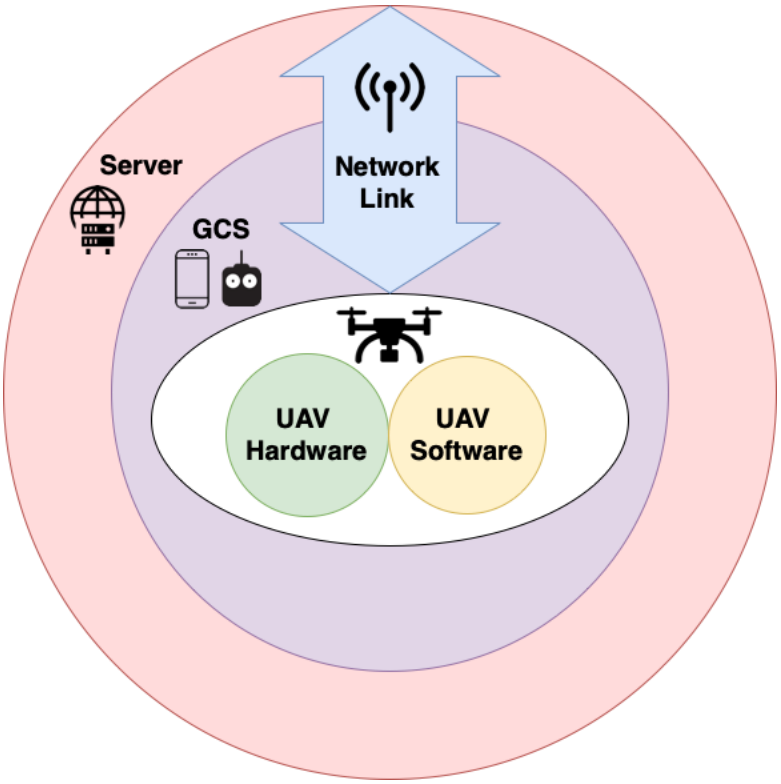

Figure 1. Components of UAS.

4.2.2 *UAV Software*

The software components of UAV include software programs installed in the UAV, such as the firmware, operating system, and application programs that are either pre-installed or possibly installed by the user (e.g., to control a special payload).

4.2.3 *Ground Control Station (GCS)*

Ground Control Station includes the system that controls the UAV remotely from the ground. It consists of hardware and software of the remote site/controller and the operator (e.g., human pilot).

4.2.4 *Network/Communication Link*

This includes communication networks and channels, and protocols used in UAS. This mainly refers to drone-to-GCS communication, but it also includes drone-to-drone, drone/GCS-to-server communications, and communications to services such as GPS. Such communications may take place over Wi-Fi, Ad-hoc Networks, cellular, or other networks depending on the application and environment context.

4.2.5 *Server/Cloud*

Server/Cloud refers to remotely located cloud servers or services that store information regarding UAS such as flight logs and registration information.

4.3 Threat Landscape Organized by UAS Components

This section covers a wide range of attacks against UAS that were identified in the literature. The attacks are categorized by the targeted components of the UAS. As shown in section 4.2, we have 5 categories for UAS components. Potential threats against each category of UAS component are discussed below. Common Weakness Enumeration (CWE) is the community developed list of software and hardware weakness types, maintained by MITRE [100]. Existing CWEs relevant to each threat are also listed.

4.3.1 *UAV Hardware*

A UAV contains various hardware components. Hardware attacks encompass any attacks against hardware components to make them malfunction or be manipulated by the attackers. Attacks in this category mostly focus on the parts whose role is to perceive the environment: sensors. In addition to the physical hardware, attacks can target related algorithm, which is responsible for processing the raw data. Two major attacks targeting hardware are jamming and spoofing, and they are labeled with the different kinds of hardware components.

Table 10. Threats to UAV Hardware.

Attack Reference	Method of Attack	Description	Relevant CWEs
HW-S/GPS	Spoofing - GPS	Synthesizing and transmitting a false GPS signals to deceive a target GPS receiver's location; Meaconing refers to capturing legitimate GPS signal and rebroadcasting with a delay, affecting the timing estimation and ultimately the GPS receiver's location.	- CWE-346: Origin Validation Error - CWE-940: Improper Verification of Source of a Communication Channel
HW-S/OS	Spoofing - Other Sensors	Compromising a computer-controlled sensor by reporting false data collected by the sensor instead of the actual data.	- CWE-346: Origin Validation Error
HW-S/ADSB-ID	Spoofing - ADS-B, Remote ID	Broadcasting illegitimate or modifying legitimate broadcast messages such as ADS-B, Remote ID	- CWE-346: Origin Validation Error - CWE-940: Improper Verification of Source of a Communication Channel
HW-S/ADSB-ID	Spoofing - Actuator	Sending signals (e.g electronmagnetic) in an attempt to spoof signal going from the controller to the actuators.	- CWE-346: Origin Validation Error - CWE-940: Improper Verification of Source of a Communication Channel
HW-J/GPS	**Jamming - GPS**	Transmitting signals to impede reception of GPS signal (i.e., impacting GPS availability).	- CWE-400: Uncontrolled Resource Consumption - CWE-406: Insufficient Control of Network Message Volume (Network Amplification)
HW-J/OS	**Jamming - Other Sensors**	Spamming signals towards a drone's mounted sensors in an attempt to output unreliable/unstable environmental readings.	- CWE-400: Uncontrolled Resource Consumption
HW-J/ADSB-ID	**Jamming - ADS-B, Remote ID**	Fill ADS-B frequency band with noisy signal, or spamming high volumes of ADS-B broadcasts in the hopes of jamming a target UAS's receiver/transceiver	- CWE-400: Uncontrolled Resource Consumption - CWE-406: Insufficient Control of Network Message Volume (Network Amplification)
HW-J/A	**Jamming - Actuator**	Spamming signals (e.g electronmagnetic) in an attempt to perform a denial of service on the link between the controller and actuators.	- CWE-400: Uncontrolled Resource Consumption - CWE-406: Insufficient Control of Network Message Volume (Network

			Amplification)
HW-FF	**Firmware Flashing**	Physically flashing the firmware, and replacing it with a modified, potentially malicious version.	- CWE-693: Protection Mechanism Failure - CWE-1191: Exposed Chip Debug and Test Interface With Insufficient or Missing Authorization
HW-SCA	**Supply Chain Attack**	Gaining control of suppliers to modify hardware components	- CWE-506: Embedded Malicious Code - CWE-507: Trojan Horse

Jamming: If the attackers can add the noisy signal into the medium (e.g. radio signal, sound, etc.), the receiver may not be able to differentiate the correct signal from the noise, resulting in the state of denial-of-service.

GPS: Attackers can jam the GPS receiver by filling the radio frequency band with noisy signals. As it is known that GPS signal for civilian use is neither encrypted nor authenticated, the receiver can be jammed when the noise signal is as strong as legitimate signals from satellites at the victim UAV. Many recent UAS have a fail-safe mode when GPS signal is not available (e.g., the signal is too weak) or not recognizable. A known course of actions triggered by fail-safe mode might be exploited. Jamming GPS can be achieved without just applying noise signals. Moser et al. conducted experiments to show that GPS signals can be canceled [74]. They could craft and apply a signal that appears identical in shape to the legitimate signal but is actually out-of-phase. This makes the signals create a destructive interface.

ADS-B/Remote ID: Similar to GPS, radio frequency band is used for ADS-B protocol and also expected to be used for Remote ID. If those frequency bands are filled with noise, it is possible that they cannot recognize the legitimate signal either.

Other sensors: Other than transceivers mentioned above, UAS may include other sensors such as Electro-Optical (EO) system, LiDAR, radar, etc. Deceptive jamming, for example, is that the attacker sends pulses to the target radar which has the same frequency and similar power as a typical reflected pulse from actual objects, thus resulting in false objects from radar's view. It is possible that any other sensor could be jammed if the medium that signal is being carried with is filled with similar but noisy signals; although it is not as easy as the case for radio frequency. Son et al. showed that Micro-Electro-Mechanical Systems (MEMS) gyroscopes, part of Inertial Measurement Unit (IMU), can fail when a strong signal, whose frequency is the same as the resonant frequency of the gyroscope, is applied [95]. As they found that many commercially available gyroscopes have resonant frequency in audible frequency (AF) range, they could demonstrate the attack by placing a speaker close to the MEMS gyroscope.

Actuators: Actuators, such as motors and payload, are controlled by the flight controller. If the attacker has access to the controller or the connection between two, they can attempt jamming by spamming noisy signals or sending messages not executable by the actuators. This may result in halting or even cause permanent damage. It requires the attacker to take control of the flight controller or connection prior to jamming, thus it adds a higher obstacle from the attackers' perspective.

Spoofing: Beyond the jamming attack, where noise signals interfere with the legitimate signal, the goal of a spoofing attack is that the target recognizes the signal from the attacker as a legitimate input.

GPS: When attackers can generate and apply their own signals to the GPS receiver, the victim can be misled about location. As creating and synthesizing radio signals becomes easier with Software-Defined Radio (SDR), spoofing the GPS could be accessible to more potential attackers. Noh et al. shows that GPS spoofing can be used as a defense to deny a UAS from the designated area by spoofing their GPS [82].

ADS-B/Remote ID: Because ADS-B and Remote ID protocols assume users are compliant with the rules, their message formats are public, and messages are not encrypted, the bar against the attacker is not high enough to prevent spoofing attacks against ADS-B and Remote ID transponders. Attackers can forge and broadcast the messages with fake information. Manesh et al. did experiments upon a simulation platform that they could inject false ADS-B message to create ghost UAS, causing nearby UAS to deviate abruptly to maintain well-clear zone [62].

Other sensors: Sensors process the raw data and output the measurement by applying relevant physics. If attackers can override the legitimate input with the maliciously crafted raw data, sensors can be deceived and output incorrect measurements. Nashimoto et al. showed that a known attitude-heading reference system (AHRS) algorithm used for inclination measurement can be spoofed by manipulating noise level in sensor input [78]. Sensor fusion, where a measurement output is decided by multiple sensors, can be deemed as a defense strategy. However, Dash et al. demonstrated that even a sensor fusion algorithm protected by a certain intrusion detection system (Control Invariant by Choi et al. [15]) can be spoofed if the attacker uses a crafted data set for sensor input [20]. Beyond jamming using resonant frequencies demonstrated previous works, Trippel et al. showed that resonant frequency can be used to control the MEMS accelerometer [104]. Optical sensors such as cameras are also used to determine movement of system. Davidson et al. demonstrated that optical flow system, which is downward-facing camera to the ground, can be spoofed by projecting forged image using projector or laser. Because the algorithm in the UAS in their experiments assume the ground image is stationary, the victim UAS is drifted if the algorithm recognizes the projected image as legitimate input and that image is intentionally drifted [22].

Firmware Flashing: Firmware flashing includes that the attacker replaces the firmware of any hardware components with a malicious version via physical access. Modifying the firmware remotely via the chain of vulnerabilities in software will be discussed in the software section.

Supply chain attack: Supply chain attack in hardware includes external attackers, as well as malicious suppliers, gain the access to the manufacturing process for a certain hardware component, resulting in producing physically flawed products. If those components are not examined and are crucial for maneuvering or executing missions, a propeller for example, it might cause the failure in completing the mission. The cases when the components containing malicious software are supplied are classified as software attacks.

4.3.2 *UAV Software*

The operating system would take a major part of UAV software, but there would be other firmware on microcontrollers for sensors, motors, communications, etc. Control and application software are also another major category.

Injection: Without modifying software, attackers can inject malicious code via legitimate I/O channels of software. In addition, attackers can put erroneous or disguise data in the database. If the software counts it as normal data, it can trigger malicious behavior when it is read and executed.

Buffer overflow: Buffer overflow is a well-known vulnerability category in cyber security, caused by poor memory management. Using buffer overflow, attackers can write over the memory of the application and

break the execution path. Although various defense strategies are available, each of them has overhead and it might be critical for applications in real-time. Habibi et al. targeted a specific platform (Ardupilot Mega 2.5) and exploited buffer overflow vulnerability in the running code [36].

Malware: Attackers can infect the software with malwares when the UAS is connected, via physical ports (e.g., USB) or wireless channels. After being infected, malwares can actively cause issues, or passively infect other software and gathering information, or stay inactive then be activated only if specific conditions are met to maximize damage (e.g., on a mission, take-off/landing)

Firmware modification: One of the ways to modify firmware is mentioned in the previous section through physical tampering. Another way to achieve this goal remotely is using the firmware update process. Without proper authenticity and integrity check, attackers can upload a modified version after they analyze the official firmware by disassembling and reverse-engineering.

Battery draining: If an attacker succeeds in gaining the privilege to execute commands, they can simply load a heavy process to keep the processing unit working with the highest clock speed. In another scenario, attackers would try to prevent the transition to "energy saving mode" or "sleep mode" by waking the target UAS up by sending an input whenever the victim is about to sleep. This will cause abnormal increase in energy consumption and in turn, the duration of operation may deteriorate.

Supply chain attack: As introduced in the hardware section, similar attack surface exists in the supply chain of software. Attackers, including malicious suppliers, can infiltrate the repository for software to be delivered to drone manufacturer, or firmware to be installed in the part they supply. Malware such as backdoor, worm, etc. can be installed. Unlike supply chain attack in hardware, the same attack in software regime would be more difficult to detect because the inspection is limited for software.

Table 11. Threats to UAV Software.

Attack Reference	Method of Attack	Description	Relevant CWE
SW-CI	**Code Injection**	Introducing additional instructions with malicious intent (i.e., Sensor Parsing, Control Algorithm Adjustment, Memory Leaks, and Structured Query Language injection).	- CWE-77: Improper Neutralization of Special Elements used in a Command ('Command Injection')
SW-DI	**Database Injection**	Exploiting database vulnerabilities, typically by adding erroneous data	- CWE-943: Improper Neutralization of Special Elements in Data Query Logic
SW-FM	**Firmware Modification**	Modifying the firmware to get to the ultimate target. Requires acquiring samples of an official firmware update, then analyzing, disassembling, and attempting to infer the method used by the device to validate updates.	
SW-BD	**Battery Draining**	Causing the system to rapidly exhaust its battery by forcing it to never sleep or to execute jobs with high computation power	

SW-BO	**Buffer Overflow**	Caused by overwriting the memory of application to change the execution path of the program that exposes private information	- CWE-119: Improper Restriction of Operations within the Bounds of a Memory Buffer
SW-MI	**Malware Infection**	Infecting software of the system with deliberate harmful intent by exploiting vulnerabilities in software that are unknown or not fixed yet	
SW-SCA	**Supply Chain Attack**	Gaining access to supplier computers and modifying the firmware, e.g., pre-installing back doors, malicious code, etc.	- CWE-506: Embedded Malicious Code - CWE-507: Trojan Horse

4.3.3 *Ground Control Station (GCS)*

GCS can have various forms - from single human pilot with a remote device or smartphone as a remote, to a large ground facility with multiple operators that manage a fleet of drones. As GCS consists of human operator(s) and control station, attack vectors would be divided into two from attackers' view respectively. Even if a UAS is fully autonomous and no human pilot is needed, human operators who plan and manage the ground station can still be targeted. The second part, control station, can be viewed as a standalone cyber-physical system, with its own hardware and software, recursively. Further categorization by 'hardware' and 'software' in GCS, however, might cause confusion in the taxonomy. To prevent this, the team considers GCS as a whole and connects methods of attack directly with it.

Because GCS communicates with both UAV and server via network link, attackers might leverage this connectivity to attack GCS remotely. If attackers have a remote access to the remote device, they can utilize known or zero-day vulnerabilities to perform various kinds of attacks on GCS. It can result in attackers having access with higher privilege (e.g., root), cutting connection with the UAV by forcefully quitting running applications, and gathering information stored in the device. Smartphones, which is currently of great interest for attackers, can be in danger if security updates are not performed timely, because the cycle for new attacks and responding patches occurs faster than other areas. Furthermore, the platform where the application for controlling a UAV is distributed is well known and accessible to attackers, therefore they can download the application, analyze it, and find security vulnerabilities in them.

Human factors have been a main subject of cyber-attacks - scam, phishing, wrong choices for passwords, etc. Many traditional and existing techniques can be used to draw human errors, including leaking passwords, installing malware, and even causing incorrect maneuvering by human pilots.

Table 12. Threats to GCS.

Attack Reference	Method of Attack	Description	Relevant CWEs
GCS-RA	**Remote access**	Infecting GCS with remote access tool, allowing attackers to take remote access of drone	- CWE-506: Embedded Malicious Code - CWE-507: Trojan Horse
GCS-FQA	**Forced quitting application**	Crashing GCS application, losing link with drone	- CWE-506: Embedded Malicious Code - CWE-507: Trojan Horse - CWE-511: Logic/Time Bomb

GCS-DE	**Data exfiltration**	Extracting potentially sensitive information stored in GCS, either relating to or not relating to drone operations (e.g., data streams, sensor measurements/location, passwords, etc.)	- CWE-506: Embedded Malicious Code - CWE-507: Trojan Horse - CWE-512: Spyware
GCS-PB	**Password Breaking**	Recovering password from data that has been stored in the device	- CWE-328: Reversible One-Way Hash - CWE-261: Weak Encoding for Password - CWE-693: Protection Mechanism Failure
GCS-RE	**Reverse Engineering GCS Application/ Software**	Reverse engineering can be performed on the GCS application to find hardcoded authentication tokens, or other potentially sensitive information	- CWE-318: Cleartext Storage of Sensitive Information in Executable - CWE-656: Reliance on Security Through Obscurity - CWE-615: Inclusion of Sensitive Information in Source Code Comments
GCS-SE	**Social Engineering**	Manipulating technique to exploit human error to gain private information/ access	- CWE-359: Exposure of Private Personal Information to an Unauthorized Actor - CWE-640: Weak Password Recovery Mechanism for Forgotten Password

4.3.4 *Network/Communication Link*

Network links are wireless communication channels used in UAS, which attackers would actively search for any vulnerabilities to break in. Because both command and data are transferred via network links, compromised networks would lead to information leakage and even losing control of UAS. Technical detail for a network link attack would be greatly dependent on the protocol used for communication. However, attacks against network links are mostly categorized based on what capabilities that the attacker has upon the communication between legitimate sender and receiver as it will help understand the new attacks in the future regardless of protocols.

Table 13. Threats to Network/Communication Links.

Attack Reference	Method of Attack	Description	Relevant CWEs
NL-BH/GH	**Black Hole/Gray Hole**	Black hole attacks involve a malicious/compromised node within a network to become a central routing point, and then to begin dropping all packets sent to the node. A gray hole attack is similar, although it selectively drops packets, instead of dropping all packets.	- CWE-284: Improper Access Control - CWE-290: Authentication Bypass by Spoofing - CWE-400: Uncontrolled Resource Consumption - CWE-406: Insufficient Control of Network Message Volume (Network Amplification)

NL-W	**Wormhole**	A wormhole attack involves two or more malicious/compromised nodes, and entails one node tunneling packets to another node, instead of taking the broadcasted route.	- CWE-284: Improper Access Control - CWE-290: Authentication Bypass by Spoofing - CWE-300: Channel Accessible by Non-Endpoint
NL-Syb	**Sybil**	An adversary registers many fake identities in an ad-hoc network. Has the potential to impact voting outcomes in FANET routing protocols	- CWE-284: Improper Access Control - CWE-290: Authentication Bypass by Spoofing - CWE-300: Channel Accessible by Non-Endpoint - CWE-694: Use of Multiple Resources with Duplicate Identifier
NL-Sink	**Sinkhole**	Adversary advertises itself as best route in network - has the potential to modify, drop or delay packets.	- CWE-284: Improper Access Control - CWE-290: Authentication Bypass by Spoofing - CWE-300: Channel Accessible by Non-Endpoint
NL-RFJam	**Radio Frequency (RF)-based Jamming**	Intentional physical interference with the reception of a required signal; the adversary needs to be in vicinity of nodes to use a strong enough signal to jam the wireless channel.	- CWE-400: Uncontrolled Resource Consumption - CWE-406: Insufficient Control of Network Message Volume (Network Amplification)
NL-PBJam	**Protocol-based Jamming (Message Flooding)**	Intentionally flooding host's network interface with protocol messages, includes ping floods, TCP handshake flooding, etc. to result in denial-of-service in network	- CWE-400: Uncontrolled Resource Consumption - CWE-406: Insufficient Control of Network Message Volume (Network Amplification)
NL-D	**De-authentication**	Sending network protocol messages to de-authenticate legitimate GCS, cutting the link between UAV and GCS	- CWE-284: Improper Access Control - CWE-290: Authentication Bypass by Spoofing - CWE-276: Incorrect Default Permissions
NL-PS/A	**Packet Sniffing/Analysis**	Listening to network communication to gain access to private information and analyzing patterns to deduce information	- CWE-300: Channel Accessible by Non-Endpoint - CWE-319: Cleartext Transmission of Sensitive Information - CWE-497: Exposure of Sensitive System Information to an Unauthorized Control Sphere - CWE-523: Unprotected Transport of Credentials
NL-PB	Password Breaking	Guessing or otherwise determining a password in documentation or brute force attack.	- CWE-259: Use of Hard-coded Password - CWE-327: Use of a Broken or Risky Cryptographic Algorithm - CWE-522: Insufficiently Protected Credentials

			- CWE-759: Use of a One-Way Hash without a Salt - CWE-760: Use of a One-Way Hash with a Predictable Salt - CWE-261: Weak Encoding for Password - CWE-328: Reversible One-Way Hash - CWE-521: Weak Password Requirements
NL-PitM	**Person-In-The-Middle**	Connecting independently to two computers that are part of the system with the purpose of eavesdropping/manipulating messages	- CWE-284: Improper Access Control - CWE-290: Authentication Bypass by Spoofing - CWE-300: Channel Accessible by Non-Endpoint - CWE-940: Improper Verification of Source of a Communication Channel
NL-CJ	**Command Injection**	Accessing a target control unit or network to execute a command with malicious intent.	- CWE-77: Improper Neutralization of Special Elements used in a Command ('Command Injection') - CWE-78: Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
NL-M	**Masquerading**	Malicious node pretending as a legitimate node	- CWE-290: Authentication Bypass by Spoofing - CWE-923: Improper Restriction of Communication Channel to Intended Endpoints - CWE-266: Incorrect Privilege Assignment - CWE-287: Improper Authentication
NL-ReplayA	**Replay Attack**	Observing and recording a communication sequence to replay it later to spoof the system	- CWE-294: Authentication Bypass by Capture-replay - CWE-290: Authentication Bypass by Spoofing
NL-RelayA	**Relay Attack**	Capturing a communications signal and relaying it through a longer-range communication	- CWE-294: Authentication Bypass by Capture-replay - CWE-290: Authentication Bypass by Spoofing
NL-F	**Fuzzing**	Gaining network access and bombarding the target with messages to observe which one has a physical effect	- CWE-119: Improper Restriction of Operations within the Bounds of a Memory Buffer - CWE-682: Incorrect Calculation - CWE-665: Improper Initialization - CWE-707: Improper Neutralization - CWE-691: Insufficient Control Flow Management

Routing: The attackers can change the route of communication traffic in order to hinder messages being delivered or broadcasted properly. Although contents of messages may not be revealed, there can be serious threats to UAS on the mission if command messages do not reach out to the UAS. This kind of attack includes black/gray hole, wormhole, sybil, and sinkhole attacks. To launch these attacks, it typically starts with placing one or more malicious nodes in the network, which are pretending to be legitimate nodes. If they succeed in attracting the traffic, they can drop, delay, or redirect the acquired packets.

Jamming: The other way to obstruct the communication traffic is jamming. Two jamming methods are introduced here. First, RF-based jamming is the same method as the one introduced in hardware attacks. If the communication protocol and its radio frequency band are known to the attackers such as Wi-Fi, 4G/5G cellular network, mmWave, as well as control and command (C2) communication via C-Band, or , they can interfere by sending strong noise signals to jam the channel. Secondly, protocol-based jamming is similar to well-known distributed denial-of-service (DDoS) cyberattack to web servers. The attacker sends a flood of messages or access requests to the network, resulting in the legitimate messages not being processed while the host handles the false messages and denies them.

De-authentication: When the connection is not properly secured, attackers can perform a de-authentication attack by sending protocol-compliant messages to the host. If it succeeds, the existing connection is cut, and the host makes a new connection with the attacker. Pleban et al. showed that UAS receives forged messages that are not from currently connected GCS if the connection is established upon User Datagram Protocol (UDP), by adjusting the internal sequence counter embedded in the message [43].

Eavesdropping: Without disrupting the connection, attackers might passively listen to the communication and record the signals. With accumulated data, they extract private information or deduce information by analyzing patterns even if the messages are encrypted. The paper by Nassi et al. showed that when flickering illumination is applied to the object and if it is filmed and streamed by UAS, the information (flickering) can be checked from encrypted video stream sent by UAS without decryption [79]. In addition, attackers can utilize the captured messages because they are written by a legitimate sender, to use it later (replay attack) or to send to a distant entity (relay attack).

Modification and fabrication: The attacker is connected to both legitimate sender and receiver independently for Person-in-the-Middle attack. They can relay messages between them then victims believe that they are communicating directly to each other. Moreover, the attacker can even forge or modify messages to confuse the victim. When the attacker has a capability of sending their own fabricated messages, the threat to UAS becomes most serious and imminent as the attacker can perform command injection attack, meaning the attacker takes full control of the UAS.

Masquerading: As explained previously, pretending to be a legitimate node in order to draw connections from victims can be a base to launch other attacks. If this masquerading attack is successful, it means that the victim trusts the attacker, which can ultimately lead to handing over sensitive information to attackers.

Fuzzing: Different from other attacks, fuzzing is when the attacker repeatedly generates messages and sees whether the forged message affects the victim. The attacker can perform the fuzzing attack without prior knowledge of the protocol or security defenses.

4.3.5 *Server*

The information stored in the remote server or cloud can be of interest to attackers. It spans data collected during flight such as flight logs, video footage, and private information about operators. Attacking servers connected to the Internet is a classic subject of cyber-attacks and existing techniques will also apply to servers for UAS. Attacks on servers can occur anytime regardless of UAS operation phases. Although the rules related with remote servers are not included in the final rule on Remote ID regulation, it is possible in the future that servers can serve a role to broadcast the location of a UAS. If so, successful attacks on servers will impact UAS on the flight in real-time.

Table 14. Threats to Server/Cloud.

Attack Reference	Method of Attack	Description	Relevant CWE
SRV-DL	**Data leakage**	Attacker is able to exfiltrate video feeds, live camera feeds, or other potentially sensitive information from the cloud/third-party server.	- CWE-284: Improper Access Control - CWE-922: Insecure Storage of Sensitive Information
SRV-PIL	**Pilot identity leakage**	Attacker is able to leak the identity of the pilot, or other personal sensitive information related to the UAS pilot.	- CWE-284: Improper Access Control - CWE-922: Insecure Storage of Sensitive Information
SRV-LL	**Location leakage**	Attacker is able to leak the current (or past) location(s) of a drone.	- CWE-284: Improper Access Control - CWE-922: Insecure Storage of Sensitive Information

4.4 UAS Operation Phases

In the preceding cyber threats to UAS components were considered. The team organized the threats around different operational phases of a UAS introduced in Section 2.4. A pictorial depiction is shown in Figure 2 below.

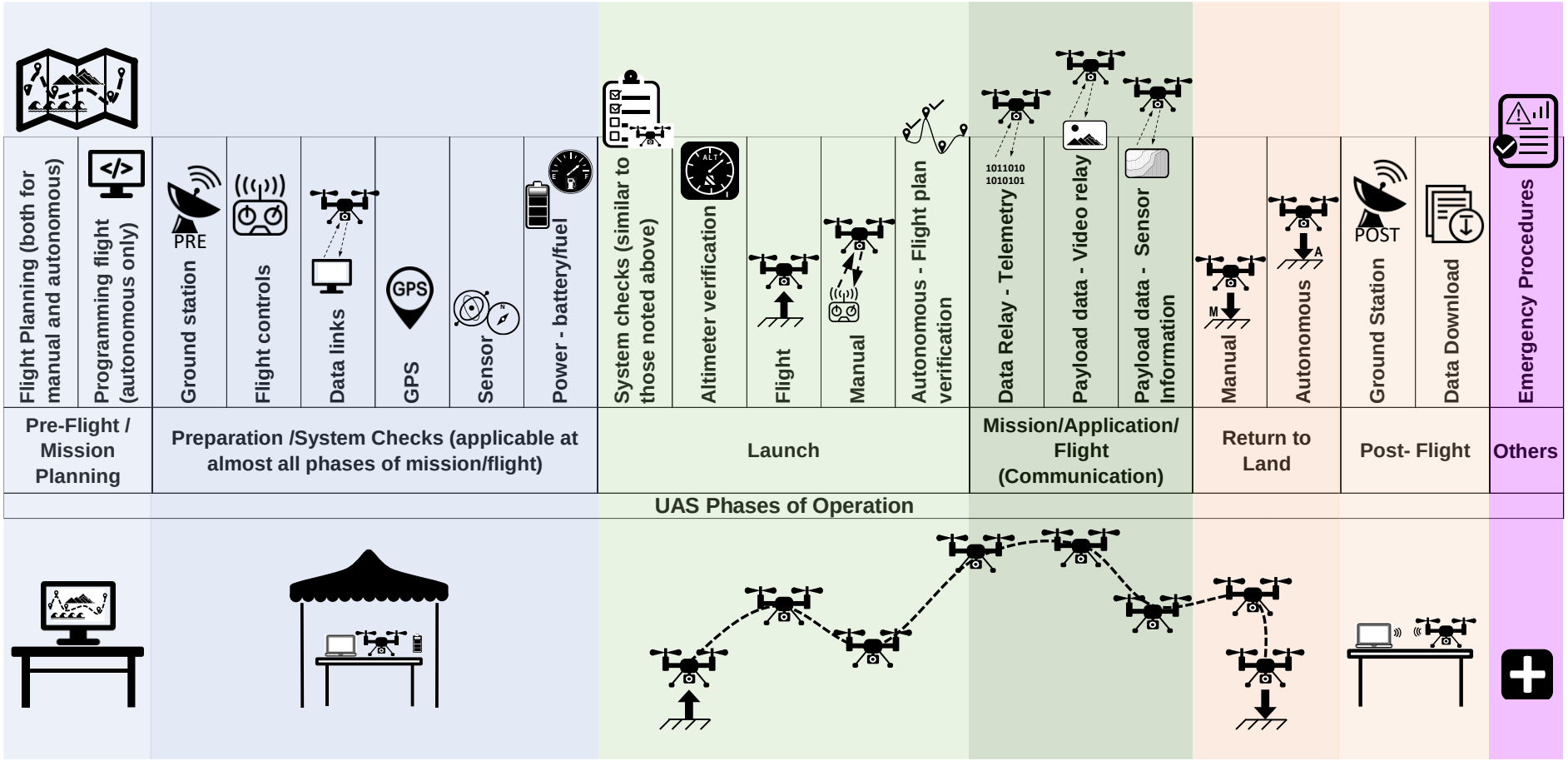

Figure 2: UAS Operational Phases

4.5 Threat Landscape from the Lens of UAS Operations

Threats to UAV were considered in all UAS phases of operation. Table 15 summarizes the UAS phases of operation that we considered along with its brief description.

Table 15. UAS Phases of Operation.

UAS Phases of Operation		Description
Pre-Flight/ Mission Planning	Flight Planning (both for manual and autonomous)	Check for flight plan, navigation plan, receive all clearances that are required. Familiarize with all relevant information
	Programming flight (autonomous only)	Establish communication between UAS and GCS.
Preparation /System Checks (applicable at almost all phases of mission/flight)	Ground station	Complete the flight report.
	Flight controls	Flight controls allow the UAV to be controlled by either a human pilot or automatically via a computer.
	Data links	Establish data link communication between GCS and UAV.
	GPS	Check GPS devices and verify that it could operate error free.
	Sensor	Check for all other sensors: barometer, altimeter, compass, camera.
	Power - battery/fuel	UAV should operate with sufficient battery/fuel to complete the flight/mission and be properly mounted.
Launch	System checks (similar to those noted above)	Check for every component/ value from the UAV system components.
	Altimeter verification	Check for UAV's altitude above sea level
	Flight	When UAV is on air.
	Manual	Manually control UAV on launch.
	Autonomous - Flight plan verification	Verify actual flight path with the one that was planned,
Mission/Application/Flight (Communication)	Data Relay - Telemetry	Record and relay reading of instruments
	Payload data - Video relay	Transfer video feed from UAV to GCS.

	Payload data - Sensor Information	Communicate information/data about the payload.
Return to Land	Manual	Land the UAV safely by a human.
	Autonomous	Safely land the UAV without human intervention.
Post- Flight	Ground Station	Fill up flight report including altitude flown, mission overview, frequencies used in communication, flight time
	Data Download	Download the flight data from UAV
Others	Emergency Procedures	In case of emergency concerning safety of person or property UAV performs a set of procedures relating to UAV, equipment and weather minimums to the extent required to meet the emergency.

Risk is determined as per severity and likelihood (or probability) of the outcome. There is a risk in each phase of UAV operations. These are defined as per current version of FAA Order 8000.369, Safety Management System (SMS).

Likelihood matrix: Likelihood is defined as the estimated probability in quantitative or qualitative terms, of a hazard's effect or outcome. It defines the occurrence rate per operation / flight hour/ operational hour[3].

Table 16. Expected Occurrence Rate Probabilities.

	Operations: Expected Occurrence Rate (per operation / flight hour / operational hour[3])
	Quantitative (ATC / Flight Procedures / Systems Engineering)
Frequent (A)	(Probability) ≥ 1 per 1000
Probable (B)	1 per 1000 > (Probability) ≥ 1 per 100,000
Remote (C)	1 per 100,000 > (Probability) ≥ 1 per 10,000,000
Extremely Remote (D)	1 per 10,000,000 > (Probability) ≥ 1 per 1,000,000,000
Extremely Improbable (E)	1 per 1,000,000,000 > (Probability) ≥ 1 per 10^{14}

Severity: Severity is the consequence or impact of a hazard's effect or outcome in terms of degree of loss or harm.

Severity Legend:

- 5: Minimal: Discomfort to those on ground.
- 4: Minor: Non-serious injury to < 3 indicators fail.
- 3: Major: Non-serious injury to >3 indicators fail.
- 2: Hazardous: Proximity of less than 500 ft to manned aircraft. Serious injury to individuals other than operators.
- 1: Catastrophic: Collision with manned aircraft or fatal injury to non-operators. Fatality or fatal injury.

Risk: Risk is the composite of predicted severity and likelihood of the potential effect of a hazard. Hazards are categorized into three levels: high risk, medium risk, and low risk. Risk levels are determined using a risk matrix. Risk on each phase of UAS helps to prioritize the mitigation strategy.

A risk matrix was used to assess the risk based on likelihood and severity of attack. Safety risk management policy 'FAA order 8040.4' was used as a baseline to calculate the risk.

Severity / Likelihood	Minimal 5	Minor 4	Major 3	Hazardous 2	Catastrophic 1
Frequent A	Low	Medium	High	High	High
Probable B	Low	Medium	High	High	High
Remote C	Low	Medium	Medium	High	High
Extremely Remote D	Low	Low	Medium	Medium	High
Extremely Improbable E	Low	Low	Low	Medium	High* / Medium

Figure 3. Severity vs. Likelihood Matrix

Table 17. Attack Type and Likelihood by Phases of Flight.

		UAS Phases of Operation																					
		Pre-Flight / Mission Planning		Preparation /System Checks (applicable at almost all phases of mission/flight)						Launch					Mission/Application/Flight (Communication)			Return to Land		Post- Flight		Others
Attack Reference Designation	**Likelihood Legend:** Frequent (A) (Probability) ≥ 1 per 1,000 Probable (B) 1 per 1000 > (Probability) ≥ 1 per 100,000 Remote (C) 1 per 100,000 > (Probability) ≥ 1 per 10,000,000 Extremely Remote (D) 1 per 10,000,000 > (Probability) ≥ 1 per 1,000,000,000 Extremely Improbable (E) 1 per 1,000,000,000 > (Probability) ≥ 1 per 10^14	Flight Planning (both for manual and autonomous)	Programming flight (autonomous only)	Ground station	Flight controls	Data links	GPS	Sensor	Power - battery/fuel	System checks (similar to those noted above)	Altimeter verification	Flight	Manual	Autonomous - Flight plan verification	Data Relay - Telemetry	Payload data - Video relay	Payload data - Sensor Information	Manual	Autonomous	Ground Station	Data Download	Emergency Procedures
HW-ID	**UAV Hardware Attack**																						
HW-S/GPS	Spoofing - GPS	E	E	D	A	A	A	A	C	D	A	A	B	B	A	A	A	A	A	D	D	A	
HW-S/OS	Spoofing - Other Sensors	E	E	D	A	A	A	A	C	D	A	A	B	B	A	A	A	A	A	D	D	A	
HW-S/ADSB-ID	Spoofing - ADS-B, Remote ID	D	D	C	C	C	C	C	C	A	A	A	A	A	A	A	A	B	B	C	C	A	
HW-S/A	Spoofing - Actuator	E	E	D	A	A	A	A	C	D	A	A	B	B	A	A	A	A	A	D	D	A	
HW-J/GPS	Jamming - GPS	E	E	E	A	A	A	A	C	D	A	A	A	B	A	A	A	A	A	E	E	A	
HW-J/OS	Jamming - Other Sensors	E	E	E	A	A	A	A	C	D	A	A	A	B	A	A	A	A	A	E	E	A	
HW-J/ADSB-ID	Jamming - ADS-B, Remote ID	D	D	C	C	C	C	C	C	A	A	A	A	A	A	A	A	B	B	C	C	A	
HW-J/A	Jamming - Actuator	E	E	E	A	A	A	A	C	D	A	A	A	B	A	A	A	A	A	E	E	A	
HW-FF	Firmware Flashing	B	A	B	B	B	B	B	B	C	C	C	C	C	C	C	C	C	C	C	C	C	
HW-SCA	Supply Chain Attack	E	E	A	B	C	A	A	A	B	A	A	A	A	C	C	D	A	A	C	D	A	
SW-ID	**UAV Software Attack**																						
SW-CI	Code Injection	A	A	A	A	A	A	A	A	D	E	E	E	E	E	E	E	E	E	E	E	E	
SW-DI	Database Injection	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	
SW-FM	Firmware Modification	A	A	A	A	A	A	A	A	D	D	D	D	D	D	D	D	D	D	A	A	D	
SW-BD	Battery Draining	E	E	E	E	E	E	E	E	E	D	D	D	D	D	D	D	D	D	E	E	D	
SW-BO	Buffer Overflow	E	E	E	E	E	E	E	E	A	D	D	D	D	D	B	B	D	D	A	A	E	
SW-MI	Malware infection	A	A	A	A	A	A	A	B	B	A	A	A	A	A	A	B	A	A	A	A	A	
SW-SCA	Supply Chain Attack	E	E	A	B	C	A	A	A	B	A	A	A	A	C	C	D	A	A	C	D	A	
GCS-ID	**Ground Control System (GCS) Attack**																						
GCS-RA	Remote access	C	C	D	C	B	C	C	D	C	C	C	C	C	B	B	B	A	A	D	D	A	
GCS-FQA	Forced quitting application	C	C	D	C	B	C	C	D	C	C	C	C	C	B	B	B	A	A	D	D	A	
GCS-DE	Data exfiltration	D	D	B	B	B	D	D	E	A	B	B	B	B	A	A	A	A	A	A	A	A	
GCS-PB	Password Breaking	D	D	A	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	A	D	A	
GCS-RE	Reverse Engineering GCS Application/Software	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	C	C	D	
GCS-SE	Social Engineering	A	A	A	A	A	A	A	A	D	D	D	A	B	A	A	A	D	D	E	E	B	
NL-ID	**Network Link Attack**																						
NL-BH/GH	Black Hole/Gray Hole	E	E	E	E	E	E	E	E	B	C	C	C	C	C	C	C	C	C	E	E	B	
NL-W	Wormhole	E	E	E	E	E	E	E	E	D	D	D	D	D	C	C	C	C	C	E	E	B	
NL-Syb	Sybil	E	E	E	E	E	E	E	E	C	D	C	D	D	D	D	D	D	D	E	E	D	
NL-Sink	Sinkhole	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	
NL-RFJam	Radio Frequency (RF)-based Jamming	E	E	E	E	E	E	E	E	D	D	D	D	D	C	C	C	A	A	E	E	A	
NL-PBJam	Protocol-based Jamming (Message Flooding)	E	E	E	E	E	E	E	E	C	C	C	C	C	A	A	A	C	C	E	E	A	
NL-D	Deauthentication	E	E	E	E	E	E	E	E	A	A	A	B	A	A	A	A	B	A	A	A	B	
NL-PS/A	Packet Sniffing/Analysis	E	E	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	
NL-PB	Password Breaking	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	
NL-PitM	Person-In-The-Middle	E	E	E	E	E	E	E	E	E	E	C	C	D	A	A	A	A	A	E	E	A	
NL-CJ	Command Injection	E	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	C	A	A	
NL-M	Masquerading	E	E	E	E	E	E	E	E	E	E	E	E	E	B	B	B	C	C	B	B	B	
NL-ReplayA	Replay Attack	E	E	E	E	E	E	E	E	A	A	A	B	A	A	A	A	E	E	E	E	A	
NL-RelayA	Relay Attack	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	
NL-F	Fuzzing	E	E	E	E	E	E	E	E	A	A	A	B	A	A	A	A	B	A	A	A	A	
SRV-ID	**Server Attack**																						
SRV-DL	Data leakage	C	B	A	A	A	C	C	D	B	B	B	B	B	A	A	A	A	A	A	A	A	
SRV-PIL	Pilot identity leakage	A	A	A	E	E	E	E	E	E	E	C	E	E	E	E	E	E	E	A	B	D	
SRV-LL	Location leakage	A	A	A	E	E	E	E	E	D	E	B	D	B	E	E	E	A	A	A	C	C	

Table 18. Attack Type and Severity by Phases of Flight.

		UAS Phases of Operation																						
		Pre-Flight / Mission Planning		Preparation /System Checks (applicable at almost all phases of mission/flight)						Launch					Mission/Application/Flight (Communication)			Return to Land		Post- Flight		Others
Attack Reference Number	**Severity Legend:** 5: Minimal 4: Minor 3: Major 2: Hazardous 1: Catastrophic	Flight Planning (both for manual and autonomous)	Programming flight (autonomous only)	Ground station	Flight controls	Data links	GPS	Sensor	Power - battery/fuel	System checks (similar to those noted above)	Altimeter verification	Flight	Manual	Autonomous - Flight plan verification	Data Relay - Telemetry	Payload data - Video relay	Payload data - Sensor Information	Manual	Autonomous	Ground Station	Data Download	Emergency Procedures
HW-ID	**UAV Hardware Attack**																					
HW-S/GPS	Spoofing - GPS	5	5	5	5	5	5	5	5	2	1	1	1	1	5	5	5	1	1	5	5	1
HW-S/OS	Spoofing - Other Sensors	5	5	5	5	5	5	5	5	2	1	1	1	1	5	5	5	1	1	5	5	1
HW-S/ADSB-ID	Spoofing - ADS-B, Remote ID	5	5	5	5	5	5	5	5	4	4	4	4	4	5	5	5	4	4	4	4	4
HW-S/A	Spoofing - Actuator	5	5	5	5	5	5	5	5	2	1	1	1	1	5	5	5	1	1	5	5	1
HW-J/GPS	Jamming - GPS	5	5	5	5	5	5	5	5	1	1	1	1	1	5	5	5	1	1	5	5	1
HW-J/OS	Jamming - Other Sensors	5	5	5	5	5	5	5	5	1	1	1	1	1	5	5	5	1	1	5	5	1
HW-J/ADSB-ID	Jamming - ADS-B, Remote ID	5	5	5	5	5	5	5	5	4	4	4	4	4	5	5	5	4	4	4	4	4
HW-J/A	Jamming - Actuator	5	5	5	5	5	5	5	5	1	1	1	1	1	5	5	5	1	1	5	5	1
HW-FF	Firmware Flashing	4	4	4	4	4	4	3	2	2	3	3	3	3	4	4	4	1	1	3	4	1
HW-SCA	Supply Chain Attack	5	5	5	5	5	5	5	5	5	5	1	1	1	5	5	5	1	1	5	5	2
SW-ID	**UAV Software Attack**																					
SW-CI	Code Injection	5	5	5	5	5	5	5	5	1	1	1	1	1	5	5	5	1	1	5	5	1
SW-DI	Database Injection	5	5	5	5	5	5	5	5	1	1	1	1	1	5	5	5	1	1	5	5	1
SW-FM	Firmware Modification	5	5	5	5	5	5	5	5	2	3	1	1	1	5	5	5	1	1	5	5	1
SW-BD	Battery Draining	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	4
SW-BO	Buffer Overflow	5	5	5	5	5	5	5	5	5	5	1	1	1	5	5	5	1	1	5	5	2
SW-MI	Malware infection	4	4	4	4	4	4	3	2	2	3	3	3	3	4	4	4	1	1	3	4	1
SW-SCA	Supply Chain Attack	5	5	5	5	5	5	5	5	5	5	1	1	1	5	5	5	1	1	5	5	2
GCS-ID	**Ground Control System (GCS) Attack**																					
GCS-RA	Remote access	5	5	5	5	5	5	5	5	5	5	3	3	3	5	5	5	3	3	5	5	3
GCS-FQA	Forced quitting application	5	5	2	2	2	3	2	2	1	1	1	1	1	5	5	3	1	1	5	5	1
GCS-DE	Data exfiltration	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
GCS-PB	Password Breaking	5	5	5	5	5	5	5	5	5	5	3	3	3	5	5	5	3	3	5	5	1
GCS-RE	Reverse Engineering GCS Application/Software	5	5	5	5	5	5	5	5	5	5	3	3	3	5	5	5	3	3	5	5	1
GCS-SE	Social Engineering	5	5	5	5	5	5	5	5	5	5	3	3	3	5	5	5	3	3	5	5	1
NL-ID	**Network Link Attack**																					
NL-BH/GH	Black Hole/Gray Hole	5	5	5	5	5	5	5	5	5	4	1	1	1	5	5	5	1	1	5	5	1
NL-W	Wormhole	5	5	5	5	5	5	5	5	5	3	1	1	1	5	5	5	1	1	5	5	1
NL-Syb	Sybil	5	5	5	5	5	5	5	5	5	3	3	3	3	5	5	5	2	2	5	5	2
NL-Sink	Sinkhole	5	5	5	5	5	5	5	5	2	2	1	1	1	4	3	3	1	1	3	3	1
NL-RFJam	Radio Frequency (RF)-based Jamming	5	5	5	5	5	5	5	5	5	5	1	1	1	5	5	5	1	1	5	5	1
NL-PBJam	Protocol-based Jamming (Message Flooding)	5	5	5	5	5	5	5	5	5	5	1	1	1	5	5	5	1	1	5	5	1
NL-D	Deauthentication	5	5	4	4	4	4	4	5	4	4	3	3	3	5	5	5	1	1	5	5	1
NL-PS/A	Packet Sniffing/Analysis	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
NL-PB	Password Breaking	5	5	1	5	5	5	5	5	3	5	1	1	1	5	5	5	1	1	5	5	1
NL-PitM	Person-In-The-Middle	5	5	5	5	5	5	5	5	5	5	1	1	1	5	5	5	1	1	5	5	5
NL-CJ	Command Injection	5	5	5	5	5	5	5	5	3	5	1	1	1	5	5	5	1	1	5	5	1
NL-M	Masquerading	5	5	4	4	4	4	4	5	4	4	3	3	3	5	5	5	1	1	5	5	2
NL-ReplayA	Replay Attack	5	5	5	5	5	5	5	5	5	5	1	1	1	5	5	5	1	1	5	5	1
NL-RelayA	Relay Attack	5	5	5	5	5	5	5	5	5	5	1	1	1	5	5	5	1	1	5	5	1
NL-F	Fuzzing	5	5	5	5	5	5	5	5	5	5	1	1	1	5	5	5	1	1	5	5	1
SRV-ID	**Server Attack**																					
SRV-DL	Data leakage	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
SRV-PIL	Pilot identity leakage	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
SRV-LL	Location leakage	5	5	5	5	5	3	3	4	3	3	3	3	3	5	5	5	3	3	5	5	3

Table 19. Attack Type and Likelihood vs. Severity by Phases of Flight.

Attack Reference Number	Severity vs. Likelihood Legend: Low - Green; Medium - Yellow; Medium/High - Pink; High - Red	UAS Phases of Operation: Pre-Flight / Mission Planning		Preparation /System Checks (applicable at almost all phases of mission/flight)						Launch					Mission/Application/Flight (Communication)			Return to Land		Post- Flight		Others
		Flight Planning (both for manual and autonomous)	Programming flight (autonomous only)	Ground station	Flight controls	Data links	GPS	Sensor	Power - battery/fuel	System checks (similar to those noted above)	Altimeter verification	Flight	Manual	Autonomous - Flight plan verification	Data Relay - Telemetry	Payload data - Video relay	Payload data - Sensor Information	Manual	Autonomous	Ground Station	Data Download	Emergency Procedures
HW-ID	**UAV Hardware Attack**																					
HW-S/GPS	Spoofing - GPS	L	L	L	L	L	L	L	L	M	H	H	H	H	L	L	L	H	H	L	L	H
HW-S/OS	Spoofing - Other Sensors	L	L	L	L	L	L	L	L	M	H	H	H	H	L	L	L	H	H	L	L	H
HW-S/ADSB-ID	Spoofing - ADS-B, Remote ID	L	L	L	L	L	L	L	L	M	M	M	M	M	L	L	L	M	M	M	M	M
HW-S/A	Spoofing - Actuator	L	L	L	L	L	L	L	L	M	H	H	H	H	L	L	L	H	H	L	L	H
HW-J/GPS	Jamming - GPS	L	L	L	L	L	L	L	L	H	H	H	H	H	L	L	L	H	H	L	L	H
HW-J/OS	Jamming - Other Sensors	L	L	L	L	L	L	L	L	H	H	H	H	H	L	L	L	H	H	L	L	H
HW-J/ADSB-ID	Jamming - ADS-B, Remote ID	L	L	L	L	L	L	L	L	M	M	M	M	M	L	L	L	M	M	M	M	M
HW-J/A	Jamming - Actuator	L	L	L	L	L	L	L	L	H	H	H	H	H	L	L	L	H	H	L	L	H
HW-FF	Firmware Flashing	M	M	M	M	M	M	H	H	H	M	M	M	M	M	M	M	H	H	M	M	H
HW-SCA	Supply Chain Attack	L	L	L	L	L	L	L	L	L	L	H	H	H	L	L	L	H	H	L	L	H
SW-ID	**UAV Software Attack**																					
SW-CI	Code Injection	L	L	L	L	L	L	L	L	H	*H/M	*H/M	*H/M	*H/M	L	L	L	*H/M	*H/M	L	L	*H/M
SW-DI	Database Injection	L	L	L	L	L	L	L	L	H	H	H	H	H	L	L	L	H	H	L	L	H
SW-FM	Firmware Modification	L	L	L	L	L	L	L	L	M	M	H	H	H	L	L	L	H	H	L	L	H
SW-BD	Battery Draining	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L
SW-BO	Buffer Overflow	L	L	L	L	L	L	L	L	L	L	H	H	H	L	L	L	H	H	L	L	M
SW-MI	Malware infection	M	M	M	M	M	M	H	H	H	H	H	H	H	M	M	M	H	H	H	M	H
SW-SCA	Supply Chain Attack	L	L	L	L	L	L	L	L	L	L	H	H	H	L	L	L	H	H	L	L	H
GCS-ID	**Ground Control System (GCS) Attack**																					
GCS-RA	Remote access	L	L	L	L	L	L	L	L	L	L	M	M	M	L	L	L	H	H	L	L	H
GCS-FQA	Forced quitting application	L	L	M	H	H	M	H	M	H	H	H	H	H	L	L	H	H	H	L	L	H
GCS-DE	Data exfiltration	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L
GCS-PB	Password Breaking	L	L	L	L	L	L	L	L	L	L	M	M	M	L	L	L	M	M	L	L	H
GCS-RE	Reverse Engineering GCS Application/Software	L	L	L	L	L	L	L	L	L	L	H	H	H	L	L	L	H	H	L	L	H
GCS-SE	Social Engineering	L	L	L	L	L	L	L	L	L	L	M	H	H	L	L	L	M	M	L	L	H
NL-ID	**Network Link Attack**																					
NL-BH/GH	Black Hole/Gray Hole	L	L	L	L	L	L	L	L	L	M	H	H	H	L	L	L	H	H	L	L	H
NL-W	Wormhole	L	L	L	L	L	L	L	L	L	M	H	H	H	L	L	L	H	H	L	L	H
NL-Syb	Sybil	L	L	L	L	L	L	L	L	L	M	M	M	M	L	L	L	M	M	L	L	M
NL-Sink	Sinkhole	L	L	L	L	L	L	L	L	M	M	*H/M	*H/M	*H/M	L	L	L	*H/M	*H/M	L	L	*H/M
NL-RFJam	Radio Frequency (RF)-based Jamming	L	L	L	L	L	L	L	L	L	L	H	H	H	L	L	L	H	H	L	L	H
NL-PBJam	Protocol-based Jamming (Message Flooding)	L	L	L	L	L	L	L	L	L	L	H	H	H	L	L	L	H	H	L	L	H
NL-D	Deauthentication	L	L	L	L	L	L	L	L	M	M	H	H	H	L	L	L	H	H	L	L	H
NL-PS/A	Packet Sniffing/Analysis	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L
NL-PB	Password Breaking	L	L	H	L	L	L	L	L	H	L	H	H	H	L	L	L	H	H	L	L	H
NL-PitM	Person-In-The-Middle	L	L	L	L	L	L	L	L	L	L	H	H	H	L	L	L	H	H	L	L	L
NL-CJ	Command Injection	L	L	L	L	L	L	L	L	M	L	H	H	H	L	L	L	H	H	L	L	H
NL-M	Masquerading	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	H	H	L	L	H
NL-ReplayA	Replay Attack	L	L	L	L	L	L	L	L	L	L	H	H	H	L	L	L	*H/M	*H/M	L	L	H
NL-RelayA	Relay Attack	L	L	L	L	L	L	L	L	L	L	*H/M	*H/M	*H/M	L	L	L	*H/M	*H/M	L	L	*H/M
NL-F	Fuzzing	L	L	L	L	L	L	L	L	L	L	H	H	H	L	L	L	H	H	L	L	H
SRV-ID	**Server Attack**																					
SRV-DL	Data leakage	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L
SRV-PIL	Pilot identity leakage	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L
SRV-LL	Location leakage	L	L	L	L	L	L	L	L	M	L	H	M	H	L	L	L	H	H	L	L	M

It can be inferred from the hazard table that risk is high after the UAV is launched for operation. Code and command injection, password cracking, and false data injection in sensor and database are high risk factors for every phase of UAV's mode of operation.

4.6 Impact of Cybersecurity Threats to UAS on NAS

To better understand the potential negative ramifications of incorporating UASs into the NAS, the impact of cyber threats against UASs and their respective effect on NAS operations must be assessed. To accomplish this, the research team first constructed a broad overview of the threat landscape against the NAS. Next, the cyber and physical impacts of cyber threats against UASs was analyzed, along with how they may pose a threat to the NAS. Finally, the transitivity between cyber threats to UAS and general threats to NAS was discussed, showing how cyber threats to UAS may pose a risk to NAS.

4.6.1 Background & Threats to NAS

The NAS is defined by the FAA as "a network of both controlled and uncontrolled airspace, both domestic and oceanic," and includes "air navigation facilities, equipment and services; airports and landing areas; aeronautical charts," and much more [27]. The purpose of the NAS is to bring safety and efficiency to air travel within U.S airspace, for both commercial and military uses. To determine what the threat landscape for the NAS looks like, the components which make up the NAS must be enumerated. In a joint report from MITRE and the FAA, the major components utilized by the NAS to ensure safe and efficient air travel are discussed [42]. One important aspect of these components is the critical nature of their operations with respect to the workflow of the NAS. If any of these services were fully taken offline, it could seriously jeopardize the safe operating conditions of the NAS. Therefore, availability of these services is of the highest priority, and thus attacks which seek to damage the availability of these services would have the largest impact. A majority of the components listed in the report, along with their associated definitions, are listed below.

- Air Traffic Control (ATC) Towers: "... provide safe, orderly and expeditious flow of traffic on and in the vicinity of an airport… towers also provide for the separation of IFR aircraft in the terminal areas" [30].
- Control Centers: "... provide air traffic service to aircraft operating on IFR flight plans within controlled airspace, and principally during the en route phase of flight" [30].
- Airport Weather Stations: Provide weather information, such as wind, visibility, weather phenomena, etc. for a specific airport [26].
- Ground/Satellite-Based Navigation: Navigational aids, also referred to as NAVAIDs, assist pilots with navigating from point A to point B. These include both ground-based NAVAIDS such as the ILS, and satellite-based NAVAIDs such as GPS [70].
- Satellite Surveillance (ADS-B): Satellite-based surveillance, such as ADS-B, is used to perform surveillance using satellite signals [28].
- Landing Systems: These include ground-based NAVAIDs such as ILS, and assist pilots in safely landing their aircraft [29].
- Terminal Radar: Radar surveillance can be used to find and show the position of aircraft in a given area. In addition to this, radar facilities commissioned by the FAA can provide other services such as safety alerts and traffic advisories [30].

- Flight Service Stations: "... provide pilot briefings, flight plan processing, enroute flight advisories, search and rescue services, and assistance to lost aircraft and aircraft in emergency situations" [30].
- Airline Dispatchers: Plans flight paths by taking into account aircraft performance and loading, enroute winds, weather information, airspace restrictions, and airport conditions [108].

Another important element of the NAS landscape is determining which entities might attempt to target and attack the NAS. In a letter released by the White House, the National Strategy for Aviation Security was revised, and included in this revision was a list of potential "originators of threats" to the NAS [101]. These threats include terrorists, hostile nation states, criminals, insiders, foreign intelligence activities, and the spread of infectious disease via air travel. Moving forward, these entities will be considered as potential adversaries which may attempt to attack the NAS.

With the NAS components and potential threat actor categories enumerated, attacks which may take place by utilizing UASs will be categorized. In November 2020, CISA released a document which advised government agencies how to protect themselves against the threat of UASs [19]. Although the CISA report proceeded with the assumption of the UAV pilot having malicious intent, which is not an assumption considered here, the categorization of threats posed by UASs are still valid. These threats are listed below.

- Hostile Surveillance: An adversary uses UAS to collect information about federal government operations, security measures, or law enforcement operations.
- Smuggling or Contraband Delivery: An adversary uses UAS to bypass security measures to deliver illegal or prohibited items onto federal property.
- Disruption of Government Business: An adversary uses UAS to interfere with federal government operations through the presence of the UAS, use of on-board cyber-capabilities, or by using the UAS to distribute propaganda onto federal property.
- Weaponization: An adversary mounts a firearm, explosive, chemical, or biological agent on a UAV or deliberately crashes the UAV in an attack.

With the information that's been collected, all possible threats to the NAS can be generated by incorporating a specific (or combination of) category of threat actors, attack methods, and NAS components. For example, one potential attack scenario would entail the targeting of an airport's terminal radar systems by a criminal group, by weaponizing UASs to deliberately crash into the physical infrastructure associated with the radar systems. To correctly associate this threat landscape to the scope of this document, the way in which UASs are utilized to attack NAS components needs to be specified further to only include threats derived from UAS cyber-attacks. In other words, the means for which a threat actor category may use UASs to attack NAS components can only be the result of the threat actor performing a cyber-attack against the UAS. To accomplish this reduction, the physical and cyber outcomes of a UAS cyber-attack must be discussed.

4.6.2 Impact to the NAS

For each threat category listed in CISA's report, methods for how they may be realized through UAS cyber-attacks will be examined. Keep in mind that for each of these threat categories, it's assumed that one method of performing these attacks would be for a UAV operator with malicious

intent to utilize their own purchased/constructed UAS for the malicious operation in question. This subsection seeks to expand these categories to include how they might be performed through UAS cyber-attacks.

Hostile Surveillance: In this scenario, the UAS is abusing the NAS by performing surveillance without authorization or permission. One scenario where this can occur is when a malicious actor hijacks a UAS, which is being used for a separate benign mission, and controls the UAS to an unauthorized area to perform surveillance. In this scenario, the collected information is sent to the malicious actor through their controlling device, or to a remote server if the UAS is connected to the internet. Another scenario entails a UAS being used for authorized surveillance of a secured area being hijacked, resulting in the collected information being sent to the malicious actors controlling device, or to a remote server. In the previous scenario, the adversary could abstain from hijacking the UAS, and instead focus on exploiting the remote server where the authorized surveillance information is being held, and exfiltrating the footage.

Smuggling or Contraband Delivery: This use case is illegal in nature, and as the UAS would likely need special equipment to mount contraband, it's assumed that there are no surrounding UASs that an adversary could hijack to perform this operation. Hijacking a benign UAS to perform this mission is still a potential threat, although it may be considered less likely to occur than other malicious operations.

Disruption of Government Business: In this category, interfering with government operations through the presence of an unauthorized UAS or through utilizing cyber-capabilities is considered. In the first instance, the presence of a UAS can disrupt operations at secured government facilities, such as airports or military bases. This can be performed by an adversary hijacking a UAS and sending it to a location which may interrupt government operations. In the second instance, a UAS may be hijacked and uploaded with cyber exploitation toolkits, or even brought back to the adversary to have other malicious tools (e.g., hardware WiFi jammer) attached as payloads. This can lead to the potential compromise of networks or hosts within proximity to the drone, or to the interruption of communications via an attached jamming device. For example, a UAS may infiltrate a secured location and perform WiFi attacks on the location's wireless network, or travel to an airport and perform Radio Frequency (RF) jamming on nearby NAS infrastructure.

Weaponization: In this scenario, a benign UAS may be hijacked and brought back to an adversary to have weapons (e.g., firearms, explosives, or chemical/biological agents) mounted to it. Another scenario could include a UAS hijacking leading to the adversary physically crashing into people, locations or things. This has the potential for a large impact to NAS, as NAS-critical infrastructure could be damaged and need repaired. The hijacked UAS could also collide with other UASs or aircraft in mid-air, causing hazardous NAS conditions and even the risk for loss of life. A legitimate UAS could also be hit by an adversary with a denial-of-service attack, causing the UAS to lose control and crash into people, locations, NAS-infrastructure, other aircraft, or other critical infrastructure such as electrical grids, oil and gas pumping stations.

In summary, with the understanding of cyber threat landscape against UAS and with an understanding of threats to NAS, scenarios can be generated to gain insight into potential threats to the NAS through cyber-attacks on UAS as they are integrated into the NAS.

5 AGENCIES USING UAS FLEETS AND POTENTIAL CYBERSECURITY RISKS

This section will explore U.S agencies at the local, state, and federal level which have been integrating UAS into their missions. To accomplish this, logical groupings of agency types will be defined. For each type of agency, the potential use cases which may be utilized by the agencies in question will be explored. These use cases will be derived from literature surveying UAS use cases for public safety agencies over the past several years. Finally, the potential for cybersecurity risks being introduced as part of UAS integration will be discussed.

5.1 U.S Agency Use Cases

In the United States, UASs have been utilized at the national, state, and local levels for a variety of public safety related use cases. In GRA Inc.'s report, Use of Drones in Public Safety, the FAA gave GRA access to their Federal Aviation Regulations (FAR) Part 107 database of UAS registrations. Using this database, GRA analyzed the number of UAS registrations, the types of agencies which were registering UASs, and the annual rate for which UASs were being registered. By doing this, an analysis of UAS adoption rates by public safety agencies could be derived, including projections for the next several years [34].

In GRA's findings, 89% of public safety agencies operating UAS (as of 2020) do so under FAA FAR Part 107 regulatory framework. Within this database, 2,399 organizations were deemed likely to use UAS for public safety purposes. The number of UASs being registered does not seem to be slowing down, over 4,600 new UAS were registered by public safety agencies in 2019 alone. As a low estimate, the GRA projects there to be an increase to the size of the public safety UAS fleet over the next few years, with an estimation of 15,600 active UAS by 2025. For a high estimate, the GRA projects there to be an increase to 41,000 active UAS by 2025.

The GRA report also cited findings from a survey performed by the Airborne International Response Team (AIRT) in the Spring of 2020 [16]. In this survey, over 500 public safety UAS users were asked questions regarding their UAS usage. 51.77% of respondents were a municipal-level agency, 30.85% were county/parish-level, 11.35% were state-level, 4.96% were federal, and 0.35% were indigenous/tribal. In the same survey, participants were asked which types of public safety UAS missions has their organization flown to date in 2020. The results of this query from 248 agencies are shown in Table 20.

Table 20. Percentage of Agencies using UASs for Public Safety Missions.

Public Safety UAS Missions	**Responses**
Crime Scene Investigation / Forensic Analysis	47.98% (199)
COVID-19 Support	23.39% (58)
Damage Assessment	45.97% (114)

Incident Command and Control (Live Streaming)	52.42% (130)
Hazardous Materials (HAZMAT) Response	20.97% (52)
Mapping (non-forensic related)	45.97% (114)
Public Information	41.13% (102)
Target Search (including Search and Rescue)	56.05% (139)
Security Overwatch (Surveillance)	41.53% (103)
Structure Fire Response	39.52% (98)
Special Event Planning	32.66% (81)
SWAT-related	36.69% (91)
Swift Water Rescue	12.10% (30)
Training / Exercises	82.26% (204)
Transport of Cargo / Equipment	2.42% (6)
Wildfire Response	18.95% (47)
Other	14.92% (37)

The AIRT survey data describes the trends of public safety UAS missions across all levels of public safety agencies in the U.S. The GRA report takes this information a step further and breaks down these trends across individual levels of government, including federal and tribal, state-level, county and municipal agencies through their own analysis of the Part 107 database. In Table 21, a summarized version of the GRA's breakdown between public safety agency jurisdiction, agency type, and registrations of drones to the Part 107 database is shown. This table accounts for Part 107 database registrations between the first half of 2016 and the first half of 2020.

Table 21. Public Safety Agency Jurisdiction, Type, and UAS Usage.

Jurisdiction	**Type**	**Total # of Agencies**	**% of Agencies**

County	Emergency Mgmt.	212	2%
County	Law Enforcement	86	1%
County	Sheriff	1,170	12%
County	Fire	72	1%
County	Not Specified	941	9%
County	Total	2,481	24%
City (Municipal)	Emergency Mgmt.	53	1%
City (Municipal)	Police	1,200	12%
City (Municipal)	Fire	448	4%
City (Municipal)	Not Specified	1,573	15%
City (Municipal)	Total	3,274	32%
State	Emergency Mgmt.	146	1%
State	Law Enforcement	801	8%
State	Not Specified	534	5%
State	Total	1,481	15%
Federal	Law Enforcement	110	1%
Federal	Not Specified	2,742	27%
Tribal	-	68	1%
Federal & Tribal	Total	2,920	29%

-	Law Enforcement	3,367	33%
-	Emergency Mgmt.	411	4%
-	Fire	520	5%
-	Not Specified	5,858	58%
-	Total	10,156	100%

As seen from the results, the GRA categorized public safety agencies at all jurisdictions into the following buckets: law enforcement, urban and wilderness firefighting, emergency management, and "Not specified". Public safety agencies which are marked as "Not specified" could not be categorized into any of the previously mentioned agency types, as the registration information did not indicate a specific purpose for the UAS. The GRA analysis reports seem to align with results from the AIRT survey, as most drone registrations from public safety agencies seem to derive from county-level or municipal-level agencies. Additionally, for specified (i.e. not including agencies marked as "Not specified") public safety agencies within a particular jurisdiction, law enforcement appears to be the most-utilized use case for UASs.

Now that the distribution of UASs among public safety agency types and jurisdictions are known, the use cases which are utilized for each agency type will be explored. Many of the use cases mentioned are derived from the GRA report and the AIRT survey.

Law Enforcement: Approximately 43% of agencies surveyed by AIRT were self-identified as law enforcement agencies. Similarly, GRA's analysis of the Part 107 database found that of the 2,399 agencies which could be identified as public safety agencies, 34% were identified as law enforcement agencies. GRA's report also outlined several potential use cases for law enforcement agencies. One of these use cases included enhancing the situational awareness of a particular location, which can entail identifying access or escape points of a particular area, animals, tripping hazards, suspects, and more. UASs can also be used by law enforcement to assist in search and rescue efforts, expanding the reach and visibility of law enforcement when searching for a missing person. Agencies which the GRA report outlined as UAS users include the Federal Bureau of Investigation, the Department of Homeland Security, Immigrations and Customs Enforcement and Customs and Border Protection for public safety efforts. In addition to these use cases for federal agencies, many more state, county, and municipal-level law enforcement agencies use UASs for crime scene documentation, enabling faster response from first responders, and other public safety efforts.

Urban & Wilderness Firefighting: "Fire rescue" described 37.23% of the public safety agencies that participated in the AIRT survey and accounted for around 5% of public safety agencies considered in the GRA report. Federal agencies that fit this description include the U.S Department of Interior, which utilizes UASs to fight and monitor wildfires. Municipal and county fire departments also use UASs to track and monitor wildfires, and are extremely useful when

considering when evacuation notices should be delivered. In addition, UASs can be used to help those trapped by a fire and first responders find each other, greatly improving evacuation efforts.

Emergency Management: Finally, emergency management was approximately 4% of public safety agencies within the GRA report, and approximately 14% of organization types within the AIRT survey (results were combined between "Emergency Management" and "EMS/Healthcare"). Use cases within this realm include natural disaster response, such as tracking hurricanes, assessing damage from flooding, creating high quality maps for areas impacted by natural disasters, supporting rescue and relief operations for hurricane victims, and much more. With respect to EMS services, this also includes deploying UAS for emergency medical services, such as delivering medicine and blood to/from hospitals and delivering medical devices such as defibrillators. The Civil Air Patrol is a national U.S agency which uses UAS in many of these categories, including search and rescue operations and aerial disaster imaging. The Department of the Interior (including the National Park Services) also utilizes UAS for search and rescue operations, as well as many state and local public safety agencies.

GRA's analysis of the Part 107 database only accounted for organizations which could be identified as a public safety government agency. As roughly 89% of public safety agencies register their UAS using the Part 107 framework, GRA's analysis of the Part 107 database is relevant to public safety agencies alone. However, these 2,399 public safety agencies only account for approximately 24% of Part 107 registrations. From this restriction, we consider a further examination of the Part 107 database which categorizes and analyzes a larger percentage of the database, beyond just public safety agencies, to be a potential future research topic.

5.2 Potential Cybersecurity Risks

Now that a portion of use cases relating to U.S agencies are understood, the cybersecurity risks involved with their execution can be examined. To accomplish this, concerns relating to the vulnerabilities of UASs and to the security of flight records among UAS users among U.S agencies will be studied. Next, methods for which cyber-attacks on UASs can introduce risks to U.S agency use cases will be discussed. To solidify understanding, a few examples of UASs introducing cybersecurity risks to U.S agency use cases will be considered.

In the GRA report, security risks associated with UASs, or UAS components, imported from China were noted especially from the Department of the Interior. The GRA report noted that from the Part 107 database, 20% of federally registered UASs and over 86% of non-federally registered UASs were manufactured by Chinese company DJI. Additionally, they noted that most western-brand UAS are either manufactured in China or are compiled using Chinese-made components. UASs which are developed in foreign countries may have a higher risk of having malware pre-installed on the hardware or firmware/software. This can also be the case for components of UASs such as actuators or sensors, where malicious chips can be installed. These malicious chips can introduce the ability for foreign entities to remotely access UAS flight information or information related to the pilot, or in a potential worst-case scenario, allow remote access to the UAS.

In addition to concern from federal agencies, there is also unease from state and local public safety agencies regarding UAS cybersecurity risks. In the AIRT survey, which is comprised of approximately 95% non-federal public safety agencies, participants were asked "How concerned are you about the security of your drone data and any potential security vulnerabilities within the

UAS or related software that might allow a foreign company or government to receive sensitive information surrounding your domestic flight operations?" In response to this question, around 80% of respondents noted at least slight concern over access to their flight data by foreign governments or companies. Concerns over unauthorized access to UAS flight data emphasize the importance of incorporating data security into UAS use cases and ensuring access control policies are in place for sensitive UAS flight records.

In the previous section, many of the cyber-attack vectors found in literature were enumerated and discussed, showing potentially how large the attack surface for a UAS can be. These attack vectors can lead to numerous outcomes during the execution of a UAS mission, including partial or full loss of control over the UAS, sensitive information being exfiltrated, etc. These attacks may or may not have direct impact on the completion of a mission, and may even have impacts extending beyond the mission itself. For example, a UAS which has lost its communication link may crash into a nearby person, potentially injuring them in addition to the incompletion of a mission. All potential impacts that can result from a cyber-attack on a UAS are important to consider when gauging risks to the completion of U.S agency use cases.

To counteract these risks, policies and procedures should be made in the pre-planning phase of a mission to ensure mission completion even in the event of the UAS becoming unavailable. In the case that a UAS loses its link to the GCS, a policy should be put in place to ensure the safety of all nearby people, as well as the safety of all nearby sensitive objects or buildings. Additionally, although UASs bring convenience and extend the capability of users in certain contexts, this should not enable reliance on UASs to complete certain jobs. In other words, contingency plans should be immediately available in the event of the UAS becoming unavailable. Furthermore, sufficient considerations regarding the safety of the UAS payload should be made.

Some example use cases for UASs in public safety agencies within the U.S, as well as the importance of contingency plans in each use case are as follows: Crime scene documentation via aerial imaging can be very useful for law enforcement agencies but may contaminate evidence if the UAS crashes. Therefore, policies should be put in place to prevent contamination when loss of control over the UAS is noticed. Wildfire monitoring is another important use case for UAS by federal and local firefighting agencies. However, quick transitions to human-based monitoring should be at-hand in the event of UAS unavailability. If a UAS becomes unavailable during blood or organ delivery to/from hospitals, contingency plans should be in place to determine the recoverability of the payload.

6 SURVEY OF POTENTIAL MITIGATION STRATEGIES

6.1 Mitigation Strategies Found in Literature

This section will cover mitigation strategies that may be implemented to thwart attacks mentioned in the previous subsections. Similar to the attacks against UAS mentioned earlier, the countermeasures presented will first be categorized by the UAS component which is being protected. An overview of these defense strategies, categorized by UAS component, can be seen in Figure 4.

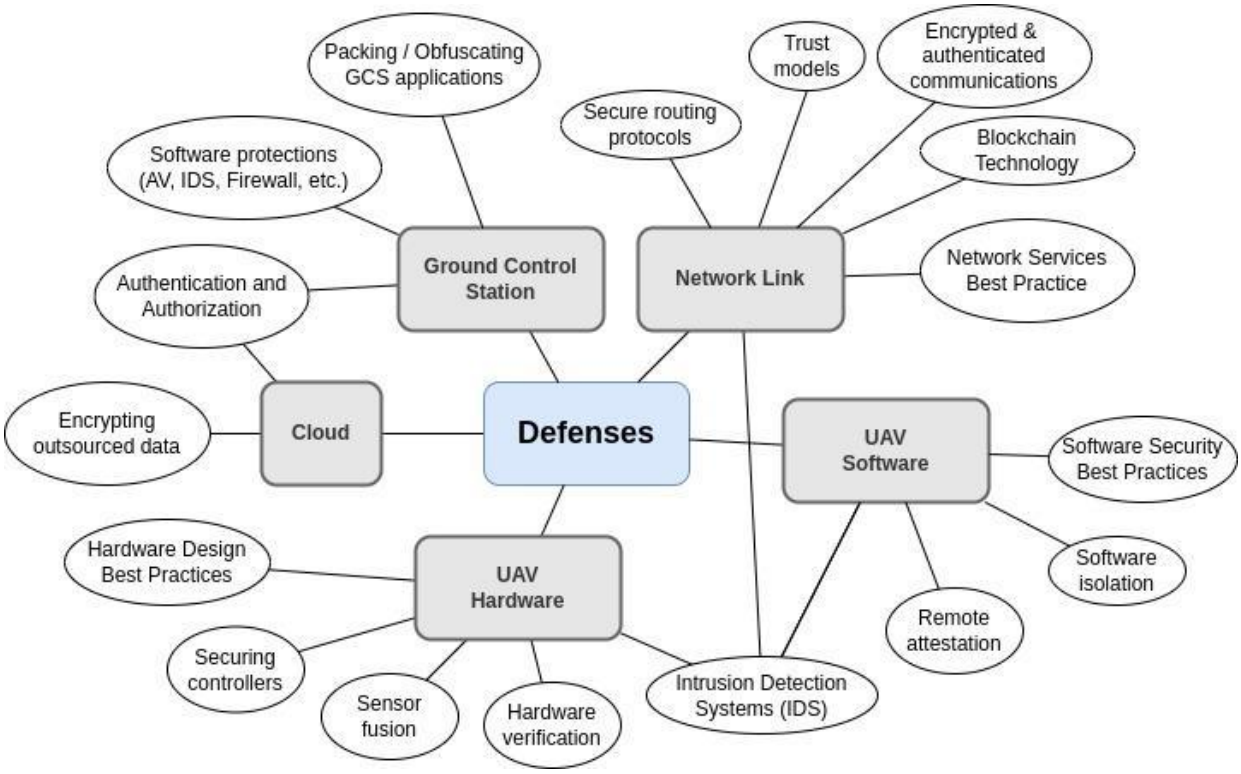

Figure 4. Enumeration of Defense Strategies for Unmanned Aerial Systems.

Before diving into the countermeasure categories for each UAS component, some terms which are frequently seen among each countermeasure category will be defined. First, a *cyber-physical system* (also denoted as a *CPS*) can be defined as a system which is controlled by a piece of software (cyber), and physically interacts with the environment in some way (physical). Additionally, an *intrusion detection system* (denoted as an *IDS*) is a piece of software designed to detect some form of malicious activity and raise alerts when they occur. There are several different design approaches for IDS, although Mitchell and Chen classified IDS for CPS by detection type and audit material [73]. Knowledge-based and behavior-based detection both observe runtime features of a system, although the former raises alerts when the features match a specific pattern or misbehavior, while the latter raises alerts when the system is behaving out of the ordinary. The final detection type is behavior-specification-based detection, and it occurs when a system's behavior falls outside a formally defined model.

It should also be noted that some countermeasures may span across more than one component category – for example, depending on the scope and focus of an IDS proposed in literature, the IDS may detect hardware, software, or network attacks, or some combination of those three.

6.1.1 UAV Hardware

The countermeasures mentioned in this subsection all relate to protecting the hardware components embedded within a UAS. There are many different types of hardware that can be embedded or attached to a drone, including cameras, RF transceivers, etc. For the purpose of generalization, these hardware components can be placed into one of three categories: controllers, actuators, and sensors. In the CPS plant model described by Giraldo et al., these components are

essential to the operation of a feedback control system [32]. A controller takes input from one or more sensors and sends commands to one or more actuators. The actuators perform some physical action or process on the drone (e.g., decrease rotor power), meanwhile the sensors observe environmental data and feed it to the controller.

With this background, the following defense strategies are aimed at protecting all three of these components in various ways. These hardware-based countermeasure strategies include intrusion detection systems, securing controllers, sensor fusion, hardware verification, and hardware design best practices.

Intrusion Detection Systems (IDS): Based on the findings from the literature review, there is a lack of knowledge-based detection mechanisms being proposed. This is likely due to the disadvantages of knowledge-based detection, which include needing to keep an updated attack-pattern dictionary, and having reduced capabilities in detecting new attack vectors (e.g., zero-day attacks). The bulk of literature revolves around behavioral-based and behavior-specification-based detection methods, as they have the potential to detect new attack vectors, although each method of detection has its limitations.

One example of behavioral-based intrusion detection for hardware components of mobile robots comes from Guo et al., where the researchers proposed an anomaly detection system for detecting sensor and actuator misbehaviors [35]. Specifically, the focus was to detect alterations in authentic sensor readings received by control units, and alterations in control commands executed by the robot actuators. The researchers accomplished this by developing a state estimation algorithm and modeling their anomaly detection algorithm to classify large differences between estimated states and actuator/sensor inputs as misbehaviors. By implementing this on two mobile robots, the researchers were able to detect signal interference, sensor spoofing, logic bombs and physical jamming attacks with little detection delay.

Other examples of behavior-based detection include work from Elnaggar and Bezzo, where they proposed a Bayesian Inverse Reinforcement Learning technique to detect sensor spoofing attacks [24]. Their technique leverages the history of sensor readings and control inputs on the CPS to predict the goal of sensor spoofing attacks. As an outcome, their method is able to determine which sensors are compromised and recover the system. Additionally, Manesh et al. elaborated that a UAS system is vulnerable to different cybersecurity attacks like a GPS spoofing attack where the attacker sends fake messages to the GPS receiver to mislead a UAV [61]. To address this security challenge, the authors proposed an efficient neural network-based method to detect GPS spoofing messages. The authors used several features such as satellite number, carrier phase, pseudo-range, Doppler shift, and Signal-to-Noise Ratio (SNR) to maximize the accuracy and probability of detection, while minimizing the probability of false alarms from occurring.

In an example of behavior-specification-based detection, Mitchell and Chen propose BRUIDS, an IDS for securing the sensors and actuators within a UAS [72]. This is accomplished by creating a series of behavioral rules for the UAS and transforming those rules into a state machine. Each state corresponds to either an unsafe state (i.e., attack state), or a safe state. A neighboring UAS or remote node monitors the UAS for compliance against the derived state machine. By adjusting variables during the testing phase, the researchers were able to successfully design tradeoffs between false positive and detection rates when the UAS was placed under attack. Another

example of behavior-specification-based detection is BRIoT, which is proposed by Sharma et al. This tool allows a user to specify an operational profile for an Internet of Things (IoT) device, generate a set of security requirements and behavioral rules, and convert the rules into a state machine. The state machine is then used for misbehavior detection [93].

Securing Controllers: The security of hardware controllers has also been an active area of research. In work by Etigowni et al., the researchers presented a control flow predictor for the formal verification of drone controllers [25]. In their scheme, a control flow predictor monitors the execution state of the flight controller and deploys pilot-designed countermeasures if the UAS is approaching an unsafe state. A data driven model using Kalman filters was utilized to perform future state prediction. The researchers tested their control flow predictor by utilizing malware to perform a series of controller-based attacks involving the injection or modification of controller data. The UAS was able to remain in a safe state and inform the operator of the safety violation in all test cases that were mentioned.

Choi et al. proposed Control Invariants (CI) that uses a checking framework for detecting external physical attacks [15]. The authors do not check traditional program-based invariants, but rather control invariants that models both the control and physical properties of the vehicle. CI are determined by a combination of physical attributes and underlying control algorithms, as well as the laws of physics. CI leverages a control system engineering methodology called System Identification (SI). SI takes a control invariant template and large set of vehicle profiling measurement data as input and instantiates the templates coefficient so that the resulted equation provides the best fit for the measurement data. These equations are used at runtime to predict behaviors for the vehicle based on input and states. The authors CI framework involves instrumenting vehicles control program binary to insert a piece of CI checking code into the vehicle's main control. At runtime, code periodically observes the current system state and independently computes expected state using CI equations. If discrepancies are observed, then an alarm is raised. Additionally, DeMarinis et al. proposed a redundancy board capable of automatically switching between two builds in flight controller [23]. In their implementation, if the current primary flight controller fails, the secondary flight controller begins operating. Incorporating redundancy into flight controllers can mitigate attacks in the event of a controller compromise.

Furthermore, Huang et al. presented a controller synthesis algorithm which solves a formulation of the "reach-avoid" problem in the presence of adversaries [39]. The researchers were able to formulate sensor, actuator, and controller attacks in such a way that a secure controller can be synthesized using a Satisfiability Modulo Theories solver. Attacks considered in this paper included partial controller software compromise, packet injection, actuator signal tampering, and sensor spoofing. The resulting controller can follow a behavioral model which ensures safety in an adverse environment.

Within the realm of secure control theory for CPS, another countermeasure that has been proposed is known as dynamic watermarking. This technique involves having actuators inject private data into a CPS and observing how the sensors connected to the CPS react to the injection. Through observing how this private data is handled by the actuators and read by the sensors of the CPS, sensor and/or actuator compromises may be detected. Dynamic watermarking is explored by Satchidanandan and Kumar, where dynamic watermarking is defined as the injection of patterns

into a medium to detect misbehaviors in sensors and actuators [92]. The authors explain how the issue of ensuring secure control over a physical plant can be addressed by performing dynamic watermarking to detect compromises on the sensors and actuators within the plant. Beyond this paper, the theory of dynamic watermarking can be extended to address the secure control of other CPS, such as UASs.

Sensor Fusion: This hardware-based countermeasure entails the use of multiple sensors, which collect data in a similar fashion, such that sensors within the same grouping may validate the data which other sensors are taking as input. Through this validation process, attacks on sensors and actuators can be detected.

Ivanov et al. designed an algorithm which incorporates sensor fusion to provide attack-resiliency into safety-critical cyber-physical systems [41]. Though developing an attack model which focuses on altering sensor measurements to decrease the certainty of the sensor fusion algorithm on a CPS, the authors showed how an attacker's capabilities are affected by sensor communication intervals. To increase the precision of their sensor fusion algorithm, the authors incorporated communication schedules between sensors, and included previous sensor readings into their approach. By utilizing these techniques and implementing their approach onto a ground robot, the researchers were able to show how the impact of compromised sensors can be reduced, as compared to not using communication schedules or incorporating previous sensor readings. Additionally, Nashimoto et al. described two attacks which possess the ability to bypass sensor fusion algorithms utilizing Kalman filters by tricking certain sensors into dominating the output of the sensor fusion algorithm [78]. Through experimentation, the researchers described how these attacks can allow adversaries to have partial or full control over the output of sensor fusion algorithms. They also presented a countermeasure for the two previously mentioned attacks, whereby analyzing the errors found in gravity and geomagnetic measurements from the sensors, these attacks against sensor fusion algorithms may be detected.

Hardware Verification: This area entails verifying that hardware components are configured correctly against a pre-defined baseline. Additionally, hardware verification can be used to identify and recover from the compromise of any hardware component. In that regard, remote attestation can be used to remotely verify the hardware or software configuration of a device. Kohnhäuser et al. proposed a protocol for remote attestation which can detect compromised hardware configurations, in addition to compromised software configurations in a simple and efficient manner [50]. The authors were also able to show that their protocol is robust to noisy and dynamic networks.

Hardware Design Best Practices: The topic of best practices regarding designing UAV hardware components focuses on implementing security at the earliest stages of hardware component creation. This subject can span a wider net than the previously covered topics, as secure hardware design goes beyond UAS use cases. One related topic that has been touched on in recent literature includes the creation of secure design methodologies for CPS. Faruque et al. proposed a framework for cross layer analysis of platform effects on security properties of control algorithms using lightweight cryptographic approaches that would minimize resource usage [5]. This approach exploits the knowledge of the system's dynamic state estimation and identification in the presence of sensor and actuator attacks and attacks on control resources. Integer Programming (IP) was used to obtain state of controlled physical process when attacker compromise system sensor and

actuators. Gomes et al. describes how an interconnected system architecture can be used to ensure the integrity of a UAS [33]. The authors described how in their architecture, each UAS comprises eleven systems which are interconnected. These systems include communications, sensing, weather report, power, maintenance and diagnostics systems, obstacle avoidance, flight management system, position determination, manual override, control unit, path planning system, and emergency response. Their architecture includes capabilities for sensing potential threats and deploying an emergency response system if an alarm is raised. Additionally in an article by Abdulhadi et al., the authors proposed a lightweight hardware solution to assure the confidentiality and integrity of both command data sent by the ground station and data transmitted by the UAV [94]. The authors use a Field Programmable Gate Array (FPGA) module to run the security functions of the UAV. Here, a cryptographic engine is the central part of the FPGA hardware architecture. The authentication and encryption keys (for use in AES encryptions/decryptions) are stored in the registers inside the FPGA during the UAV setup to protect the keys from leaks. The cryptographic engine can only read the authentication keys and the encryption keys and has no read interface to the outside.

Additional hardware security topics include protecting hardware from being reverse engineered and protecting from hardware-specific attacks. In Quadir et al., several techniques used for protecting chips, boards, and firmware from being reverse engineered are explored. Some of these tactics include obfuscating or camouflaging designs, utilizing external keys to prevent piracy, using unmarked chips, and creating tamper-proof fittings [88]. Additionally in Mead et al., the researchers were able to implement hardware-based sandboxing to non-trusted system-on-chip components to secure UAV hardware [67]. Their approach included passing input signals through a property checker, which asserts the legitimacy of the signals processed by the RF receiver. Through simulated testing, they were able to show how their design could detect and prevent RF-based jamming attacks. Furthermore, Thiha et al. proposed an efficient method to detect jamming attacks [102]. In the authors' jamming attack model, the attacker tries to detect the synchronization header (SHR) of a data frame structure defined in wireless standards. When the attacker detects an SHR, it transmits interference signals to break down the communication channel between the sender and receiver. In the proposed countermeasure, the authors introduce a timing channel that is used to prevent the detection of the actual data packet. This is accomplished by having the sender transmit a short dummy packet followed by the actual packet, such that the attacker will intercept the dumpy packet and not the relevant data.

6.1.2 UAV Software

In this subsection, strategies for protecting the software or firmware running on a UAS will be explored, including applications and processes running on the UAS and the underlying operating system. These countermeasures include software-based intrusion detection systems, remote attestation, trust models, software isolation, and software security best practices.

Intrusion Detection Systems (IDS): In comparison to IDSs which protect the hardware components and the network link of the UAS, there seem to be far fewer proposals in literature which focus on detecting intrusions that target software applications, operating systems, or firmware. Another way of viewing this gap is that there is more focus placed on the data being captured and processed by the UAS (e.g., RF signals or network packets), rather than the software already running on the UAS. Of the software-based IDSs proposed in literature, there is a lack of

knowledge-based and behavior-specification-based detection mechanisms. The lack of knowledge-based detection in literature is due to not having the capacity to detect new attack vectors, and requiring an updated attack dictionary, as mentioned earlier. Additionally, the downsides of behavior-specification-based detection mechanisms is that they require expert knowledge of the entity being specified, and significant effort to create said specification. The remaining detection mechanism considered, behavior-based detection, was observed with respect to software-based UAS protections.

One behavior-based IDS found in literature that defends against software attacks was proposed by Stracquodaine et al., which detailed a system for detecting malware placed on the UAS, in addition to hardware failure, communication channel corruption, and sensor spoofing [96]. To detect malicious software, one of the components of their IDS monitors the control flow of the UASs operating system and autopilot software and feeds it into an event processor that compares it to a normal profile (which is derived offline) to detect anomalies in real-time. Through simulation, the authors used an exploit to gain access to a UAS and alter the control flow of the autopilot and showed how their implementation is able to detect the anomalous flow. The methodology depicted in this work fits well with software-level attacks that may occur on UASs, as control flow hijacking is a common trait among many forms of malware.

Additionally, Vuong et al. proposed an IDS for robotic vehicles to detect DoS, command injection, and malware attacks [106]. In their approach, the researchers utilized the processes already running on the system to collect logs pertaining to the cyber and physical operations of the device. Features were extracted from this data and were used to train a lightweight decision tree (machine learning) algorithm. Their evaluation depicted moderate to high accuracy in detecting the previously mentioned attacks, and additionally had low latency due to the lightweight nature of their decision tree implementation. The authors placed emphasis on low-latency attack detection, due to the mobility and energy requirements of mobile cyber-physical systems. Due to the similarities between robotic ground vehicles and UASs, it's believed that a similar implementation could be adapted to perform software-based detection on a UAS.

In an article by Lu et al, the authors proposed a timing data driven malware detection approach using a novel "normal timing method" [59]. It uses several anomaly detection methods: range based, distance based, and support vector machine classification. System behavior model separates system timing into subcomponents instead of utilizing a lumped timing model. Data driven malware detection also utilizes an extended threat model, which incorporates an extensive set of real-world malware attacks. Additionally, their approach implements efficient and non-intrusive hardware detectors for range and distance-based detection, as well as SVM anomaly detection.

Remote Attestation: Software-level attacks on a UAS, such as the introduction of malware to the system, may attempt to change the software configuration of the UAS for the purpose of establishing persistence, increasing privilege levels, or for other malicious motives. Having the capability to verify the software configuration of a remote node (such as a UAS) would have the potential of mitigating software-level attacks and detecting when the node is compromised. Remote attestation serves this exact purpose, allowing a remote node to have its software or hardware configuration verified by a trusted source node.

Kohnhäuser et al. describe a remote attestation protocol for detecting both software and hardware compromises [50]. In their protocol, a "leader" node constantly generates and propagates session keys to other networked devices, for which the devices must mutually authenticate themselves prior to key propagation. Through this method, both physical tampering and software compromises can be detected, and through evaluation this protocol was shown to be efficient and robust to failures or DoS attacks. Asokan et al. proposed an attestation scheme for remotely verifying the configurations of an entire device swarm (e.g., a swarm of robotic vehicles), which is based on assumptions derived from their proposed security model for device swarms [10]. The researchers also presented two working prototypes of their attestation scheme, and their evaluations depicted scalability of their scheme for device swarms up to one million nodes.

Additionally, Ambrosin et al. proposed a practical and secure collective remote attestation protocol for highly dynamic swarms to ensure that all UAV configurations are up to date [7]. In their protocol, the UAV starts a local attestation to check whether its software is running corresponds to a known good configuration. Then, the UAV exchanges their produced attestation and shares their knowledge through a consensus algorithm, enriching their knowledge on the nodes of the network after each step. To guarantee the correctness of the consensus messages, each UAV includes a Trusted Execution Environment (TEE). Each TEE stores the network status and signs the messages exchanged with other UAVs, also adding a timestamp. To obtain knowledge about the network, a verifier can query any device in the network, and if the UAV is not compromised, it will return the consensus state representing its knowledge about each node.

Software Isolation: If a UAS is compromised, having the capability of isolating untrusted software from performing malicious actions on the device is beneficial to mitigating software-level attacks. Liu and Srivastava proposed a mechanism that accomplishes this by defining trusted and untrusted computing blocks and enforcing user-defined access controls to dictate peripheral access, as well as secure communication channels between computing blocks using encryption and signature schemes [56]. Yoon et al. presented a similar framework based on virtualization, which switches to a trusted control state in the event of a violated safety condition [111]. Countermeasures which enable software isolation can serve as a second layer of defense in the event that compromises aren’t detected through IDSs or remote attestation.

Additionally, Jiyang et al. explained that Denial-of-service (DoS) attacks aim to exhaust system resources which can cause system overload and prevent some or all legitimate requests from being fulfilled by a UAS [13]. To address this security challenge, they proposed a software framework that offers DoS attack-resilient control for real-time UAV systems using containers. Their ContainerDrone framework provides defense mechanisms for three critical system resources: CPU, memory, and the communication channel. For protecting CPU resources, they utilize the Linux kernel feature control group (cgroup) and Docker’s built-in mechanism. For memory protection, MemGuard – a Linux kernel module for implementing rate limiting of CPU accesses to memory - was used to protect their system from memory bandwidth DoS attacks.

Software Security Best Practices: Integrating software security best practices while building UAS software has strong potential to mitigate vulnerabilities which may appear during the development process. Similar to hardware design, best practices for software security is a wide-ranging topic that spans far beyond UAS-specific software development, although there exists literature which apply directly to UAS software. Specifically, the literature that was found during

our review include the use of tools to analyze software binaries for vulnerabilities, and other methods for ensuring real-time software security best practices.

One strategy for finding software vulnerabilities is fuzzing, which entails an automated assessment of test cases, where the test cases can derive from (or just partially include) randomly generated data. An example of fuzzing for vulnerabilities in robotic vehicles includes work from Kim et al., where the researchers created a policy-guided fuzzer which ensures a robotic vehicle "... adheres to identified safety and functional policies that cover user commands, configuration parameters, and physical states" [49]. The fuzzer proposed by the authors takes as input a policy given in English and translates it using Metric Temporal Logic (MTL), mutates the metrics to guide the fuzzer and detect when a test case violates the given policy, and presents the violating test case after a post-processing phase by removing irrelevant inputs. Their fuzzer was evaluated against three robotic vehicle controller programs, including the ArduPilot UAS controller software, and were able to find over 150 previously unknown software bugs. Through integrating smart fuzzing, such as creating test cases derived from certain policies, software vulnerabilities which can lead to specific unsafe scenarios can be addressed and mitigated.

Another software-level defense mechanism applicable to UASs mentioned in literature entails reverse engineering embedded binaries to search for vulnerabilities. Sun et al. proposed a tool, which can be attached to reverse engineering software, to extract semantic information from the binary executable and use that to perform vulnerability assessment and binary patching [98]. The tool presented by the authors extracts the control flow graph of a specified function and derives a symbolic expression of the function via symbolic execution, which is then compared to the abstract syntax tree of the algorithm being implemented through the binary. Their tool was evaluated on over 2,000 firmware binaries (including UAS firmware binaries) and was able to assist in identifying a zero-day vulnerability in a Linux kernel controller. As this is a plugin for the IDA Pro decompiler, this goes to show that acquiring the source code for a given UAS firmware is not necessary to perform vulnerability assessments against the firmware binary.

One method for ensuring real-time software security was presented by Abdi et al., where they proposed that frequent restarts and diversification for embedded controllers can increase the difficulty of launching attacks [2]. The proposed software restoration action is composed of restarting the system and reloading the uncompromised image of controller software. This is better than detection as a perfect intrusion detection mechanism does not exist. Frequent restarts and diversification for embedded controllers increase the difficulty of launching attacks. However, restarting the entire system or its components in runtime is not novel and may cause a problem of software aging. This method is suitable if the restart time is very small relative to the speed of physical system dynamics, and during the case where damage to plant has more severe consequences than reduced control performance due to proactive restarts.

Cho et al. proposed another method for software security – a UAV specific random number generator, DroneRNG [14]. Random number generators used in UAVs are not tailored for UAVs specifically as they use random sources generated on a desktop, not a UAV. All the UAVs use open-source cryptographic libraries such as OpenSSL or standard C random function to generate random numbers. The open-source cryptographic libraries collect random sources from user input resources available on PCs, such as user/external peripherals like a desktop PC’s mouse or keyboard, interrupt request time, and disk reading and writing time. However, it is not suitable for

UAVs because the UAVs have no such peripherals, and some UAVs have no operating systems. DroneRNG considers the sensor characteristics that UAVs present in flight and in a stationary state, using accelerometer, gyroscope, and barometer signals captured by UAVs while flying and on the ground. Using real experiments, the authors noted that sensor outputs are different when the UAV is in flight and when it is stationary. The random numbers generated by DroneRNG achieve enhanced statistical randomness and passed all NIST randomization tests.

6.1.3 Ground Control Station (GCS)

Ground control stations are pilot-operated entities which send commands to one or more UASs over a network link. Due to the C2 connection between the GCS and the UAS(s), a compromise of the GCS has the potential to enable a wide range of future attack possibilities upon the UAS(s), which may lead to critical impacts such as UAS hijacking. Countermeasures mentioned in literature for protecting the GCS were not seen as often as other components, although what was found could be split into three categories: packing/obfuscating GCS applications, software protections, as well as authentication and authorization.

Packing/Obfuscating GCS Applications: One method seen in literature for protecting GCS from attackers is in Nassi et al., where the authors claim that code can be obfuscated in order to make reverse engineering the GCS application more difficult [80]. Methods of code obfuscation include tools called "packers," which can obfuscate a binary program to hide its true functionality and make vulnerability analysis more difficult. This can prevent attacks such as the one mentioned in Aaron Luo's 2017 DEFCON presentation, where he reverse-engineered a GCS application to identify hard-coded authentication tokens, which he then used to gain unauthorized access into a drone in a demo [60].

Software Protections: Another method for protecting GCS from attackers include the use of software protections, such as an IDS, antivirus, firewall, or other security solutions designed to block known attacks, or isolate the GCS from an untrusted network. These aren't mentioned in literature, although are deserving of being mentioned as they can serve a vital role in protecting the GCS from known attacks. For example, a firewall can isolate a GCS from unwanted network traffic generated by an attacker, and an IDS can detect and alert the GCS user if known attack methods targeting the GCS are identified.

Authentication & Authorization: To ensure only authorized users are authenticated to the GCS, methods which provide user authentication and verify user authorization should be supported. Our literature review was not able to find any work relevant to user authentication or authorization to the GCS alone, however existing methods for meeting this requirement include solutions such as multi-factor authentication.

6.1.4 Network Link

In this subsection, countermeasures against attacks on UAS network links found in literature will be reviewed. The definition of a network link that we will be working with throughout this subsection is the communication signals sent and received between drones, ground control stations, cloud environments/third party servers, or other nodes within the drones operating environment. The network link defense strategies mentioned in literature mainly focus on UAS-to-GCS links using the IEEE 802.11 suite (WiFi) for RF communications, although other mediums exist such as other sub-GHz RF channels, cellular networks (e.g., 3G, 4G/LTE), and satellite

communications (SATCOM). Categories included in this subsection include intrusion detection systems, encrypted & authenticated communications, secure routing protocols, blockchain technologies, trust models and network service best practices.

Intrusion Detection Systems: Similar to software-based IDSs, there seem to more behavior-based detection systems mentioned in literature, compared to knowledge-based and behavior-specification-based detection systems. The reason for this is identical to the reasoning provided in hardware-based and software-based IDS categories.

One example of a behavior-based IDS which protects UAS networks, or more generally nodes in a mobile ad-hoc network, from network-link attacks is proposed by Lauf et al. Their distributed IDS determines a static set of behaviors offline (which are specific to the nodes running in the ad-hoc network) and analyzes patterns within the probability density function of the application to capture the typical behavior of the network. Through evaluation, the authors were able to show that their IDS can detect spoofing and jamming attacks efficiently [52]. This methodology captures semantic information provided by the nodes application and can be used to detect compromised nodes within the network.

Moustafa and Jolfaei introduced an autonomous intrusion detection scheme based on machine learning to detect cybersecurity attacks such as DoS, DDoS and probing attacks [75]. To build an efficient machine learning-based IDS, the authors generated their dataset based on launching the events of the attack in the UAS system communications, stored it, and tagged it as either malicious or benign. To launch the attacks, the authors proposed a new synthetic testbed that includes multiple virtual machines that connect to multiple UASs. The testbed environment includes a Kali Linux virtual machine to launch DoS, DDoS and probing attacks. Decision tree, K-nearest neighbors, multi-layer perceptron, naïve bayes, and support vector machine were used for classification. Additionally, the models were compared using five metrics: accuracy, precision, recall, fall-out (FPR), and F1 score.

For anomaly estimation in UAV networks, Zhang et al. suggested a hybrid solution based on both spectral traffic analysis and a resilient controller/observer [112]. This method is based on the functional and dynamic behavior of TCP and UDP networking. A statistical signature of the traffic exchanged in the network is considered as a preparatory step in the suggested hybrid technique. Anomalies are detected by comparing this signature from a bank of signatures.

Miquel et al. proposed an IDS for UAS fleets [71]. This method is based on Lyapunov Krasovkii functional and dynamic behavior for TCP. It proposes a controller/observer algorithm that can detect traffic anomalies. This scheme can detect different type of DDoS attacks based on a traffic characterization analysis. Additionally, Ying et al. introduced the SODA (spoofing detector for ADS-B) framework, which uses Deep Neural Networks to identify ADS-B spoofing [110]. The framework includes a message classifier and an aircraft classifier. The airplane classifier detects faked communications using the phases of received messages as input, while the message classifier detects malicious network traffic from adversaries launching attacks from the ground.

Encrypted & Authenticated Communications: When network traffic is sent to/from a UAS, it's vital that the traffic is obfuscated to protect privacy, the messages have tamper protections in place to protect integrity and authenticity, and that the parties involved in the communications are authenticated. When these protections aren't in place, an attacker could capture and analyze traffic

being sent/received by the UAS, alter the message, send forged messages, or impersonate a network member. This subsection will discuss countermeasures which protect confidentiality, message integrity, message authenticity, and user authentication. Additional countermeasures which enable protocol-specific security guarantees will also be mentioned.

Confidentiality ensures that data sent over the network link cannot be intercepted and analyzed by an eavesdropper. Typically, this involves the utilization of symmetric (or secret-key) cryptography, along with a method for securely generating and exchanging the symmetric key, such as asymmetric (or public-key) cryptography. In work by Allouch et al., it was demonstrated that symmetric encryption algorithms could be implemented on the MAVLink protocol to provide data confidentiality [6]. ChaCha20, AES and RC4 were tested, and were shown to entail only slightly higher CPU/memory overheads. Additionally, He et al. proposed that if the communication medium is utilizing the IEEE 802.11 suite, then enabling WPA2 and using large keys can ensure data confidentiality [37].

When sending a message over a network link, message integrity entails the confirmation that the message hasn't been tampered with, while message authenticity ensures that the message originated from an authentic source. In order to protect data integrity, a message authentication code can be used to create a checksum of the data and can be appended to a message before it's sent. Samaila et al. stated that the use of a Hashed Message Authentication Code (HMAC) utilizing a symmetric key can provide two-way authentication, given the symmetric key is securely exchanged between the sending and receiving nodes [90]. Using a HMAC would serve as a message checksum, so this would also protect message integrity.

In addition to message authenticity, ensuring user authentication is an important feature for countering cyber-attacks. This entails allowing authorized users to a specific resource and putting protections in place to prevent any unauthorized use of resources. Additionally, Tanveer et al. presented an authenticated key exchange protocol for IoD deployments, which ensures secure and reliable communications by providing mutual authentication between mobile users and securely deriving session keys [99]. Through their multi-phase protocol which includes drone and user registration, user authentication and key exchanges, as well as password/biometric updates, a variety of attacks can be prevented, including impersonation, person-in-the-middle, DoS, and replay attacks.

In the literature review, there were several countermeasures proposed for providing security to ADS-B protocol communications. Costin and Francillon described several ADS-B threats that have been proposed in literature, including eavesdropping, jamming, spoofing, and message injection/modification [18]. They were also able to successfully perform replay and impersonation attacks using COTS transceivers. To mitigate the threats that were mentioned, they suggested a lightweight public key infrastructure (PKI) implementation for resource constrained devices, adding message authentication codes to ADS-B broadcasts, and introducing key distribution by certifying bodies (e.g., FAA, EUROCONTROL, etc.).

Additionally, Manesh and Kaabouch analyzed the risk existing within ADS-B [63]. Potential attacks that were mentioned in their work included eavesdropping, message deletion/modification/injection, and jamming. The researchers proposed several mitigation strategies, which can be categorized into two main groups: secure broadcast authentication, and

secure location verification. Secure broadcast authentication mechanisms ensure that ADS-B broadcasts originated from an authenticated source and were sub-categorized into cryptographic and non-cryptographic schemes. Secure location verification mechanisms attempt to verify the location of a broadcaster, and include techniques such as distance bounding, Kalman filtering, and data fusion. The breakdown of ADS-B attack countermeasures into secure broadcast authentication and secure location verification was also performed by Strohmeier et al. [97].

Secure Routing Protocols: To protect the availability of the network link, or to ensure that the link remains operational during the UAS mission, there needs to be protections against attacks focused on message routing between network members. Attacks against the UAS's routing protocols involve manipulating the routes which messages take using the UAS network link with malicious intent. Examples of routing protocol attacks include wormhole, blackhole, and sybil attacks.

Samaila et al. described several routing attacks that can take place within a UAS network, as well as corresponding mitigation strategies [90]. Sinkhole and wormhole attacks may be prevented using geographic routing, selective routing (gray/black hole) attacks may be prevented by using multipath routing, "hello floods" and other protocol-specific flooding attacks may be prevented by using bi-directional authentication, and sybil attacks can be mitigated through the use of random key pre-distribution schemes. Additionally, Maxa et al. proposed a secure reactive routing protocol (SUAP) [66]. To ensure message authenticity, the authors used public key cryptography, hash chains, and geographical leashes. The concept works well for identifying and preventing wormhole and blackhole attacks, as well as other types of attacks which focus on network link routing.

Lei et al. demonstrated that Named Data Networking (NDN) based UAV Ad-hoc networks (UAANETs) bring new security challenges such as content poisoning [53]. The NDN-based UAANET is an in-network caching mechanism. An attacker can alternate the cache on the routers, which leads to performance degradation. To address this security concern, the authors proposed an efficient framework that integrates Interest-Key-Content Binding (IKCB), forwarding strategy, and on-demand verification to efficiently discover poisoned content. To achieve a decentralized IKCB store and detect internal attackers, the authors use a permissioned blockchain technology to verify and record the IKCB rules that bind content name, Publisher Public Key Digest (PPKD), and content digest together. They design an efficient and scalable Adaptive Delegate Consensus Algorithm (ADCA) in the blockchain without the mining procedures. ADCA provides high scalability and performance and guarantees eventual consistency in this process.

Agron et al. proposed a secure routing protocol that utilizes a Flying Ad Hoc Network (FANET) between a GCS and UAS, ensures integrity and confidentiality, and provides authentication [4]. To ensure integrity, the authors use nonce hash mechanism. To ensure confidently, the TWINE algorithm - a lightweight and simple algorithm that uses 64-bit clock sized and support two key sizes: 80-bit and 128-bit - is used to protect the critical fields in routing messages. To provide authentication, the authors used a hybrid authentication procedure. A hybrid (symmetric and asymmetric) key encryption algorithm and digital signatures are used to protect the packet field of sensitive information such as geographic information of UASs. Moreover, the authors use a packet leashes mechanism to prevent wormhole attacks.

Blockchain Technology: One pattern found in our literature review includes the usage of blockchain technologies to secure UAS networks. Here, the tamper-resistant distributed storage properties of blockchains can be of use in creating secure protocols to protect UASs and their networks against a variety of attacks.

One example of this is proposed by Li et al., where the authors constructed a private blockchain at the GCS to distribute and store group key broadcast messages [54]. Through this, they were able to develop a group key distribution scheme for UAS networks which allows network entities to recover lost group keys in a secure and timely manner. The authors analyzed their protocol against two adversary models and showed that their scheme can effectively resist various attacks with limited time and storage overhead. Additionally, Liu et al. proposed a routing strategy by utilizing blockchain technology to prevent the disclosure of sensitive network topologies in the presence of compromised peers [57]. The authors leverage the consensus process in blockchain applications to automatically detect malevolent (or compromised) participants, preventing route rules from being intentionally modified or widely revealed.

Furthermore, Aggarwal et al. designed a system model to satisfy secure data dissemination in an IoD environment using Ethereum blockchain technology [3]. Their proposed model provides secure communications between the UAVs and the users in a decentralized manner. Through incorporating a blockchain-based approach to collect information from UASs, the authors' proposed scheme can provide integrity, authentication and authorization to the stored data.

Trust Models: Another method for protecting UAS network links found in literature involves the utilization of trust models. This technique seems to be more popular with UAS swarms and entails the behavioral monitoring of nodes within the UAS network for the purpose of calculating a trust score and evicting untrustworthy nodes from the network if their trust score falls below a certain threshold.

Keshavarz et al. proposed a trust monitoring mechanism where a centralized unit (e.g. the GCS) regularly observes the behavior of a UAV network in terms of their motion path, their consumed energy, as well as the number of their completed tasks and measure a relative trust score for the UAVs to detect any abnormal behaviors in a real-time manner [48]. The authors used an audit unit to differentiate between the abnormal behaviors due to the cyber-physical attacks or the potential unusual actions (e.g., turbulence or irregular energy consumption) due to harsh environmental conditions. The proposed trust monitoring approach estimates the performance of the UAVs based on the observation of the audit unit while accounting for the potential uncertainty in such observations. The trust model can also detect malicious UAVs, which can be under various cyber-security attacks such as flooding attacks, man-in-the-middle attacks, and GPS spoofing attacks in real-time. Another example of trust models include work from Ge et al., where the authors utilized a trust-scoring mechanism for detecting network-based attacks, such as black/gray hole, sybil, DDoS and person-in-the-middle attacks [31].

Network Service Best Practices: Ensuring that the underlying drone network, as well as network services used by the drone and other members of the network, is crucial to preventing network link attacks, and has been mentioned often in literature. Many of the best practices mentioned in literature are often easily configurable within the UAS, or the device hosting the drone network.

Samland et al. discussed several strategies related to network service best practices. These included the use of WPA2 with strong passwords and replacing the use of Telnet and FTP services with SSH [91]. He et al. also discusses methods for securing WiFi networks such as disabling network SSID broadcast and restricting the network to pre-registered MAC addresses [37]. Furthermore, Samaila et al. emphasized educating end-users on cybersecurity best practices, having strong password requirements, and enabling/monitoring security event logging [90]. Additional recommendations include only enabling network services that are critical for UAS missions, as well as ensuring the operating system and network services are always up to date.

Finding vulnerabilities which can be exploited over the network is discussed by Hooper et al., where they proposed a fuzzy technique to discover weaknesses in software services (e.g., DoS and buffer overflows) in the Parrot Bebop UAV during AR Discovery [38]. The authors proposed a security framework for Wi-Fi based UASs to guard against basic attacks, such as those previously mentioned. For example, buffer overflow attacks can be defended against by filtering the input that the UAS receives over the network and ensuring the input won't corrupt the memory of the network service.

6.1.5 Cloud/Server

In this subsection, countermeasures relating to UAS data held by third parties, such as cloud providers, will be explored. Advancements in IoD applications have introduced the concept of utilizing cloud-based services to store data captured onto the cloud, and to use the cloud to perform computationally intensive operations. Although this has improved the capabilities of drones, it also has broadened the attack surface to include data being transmitted to the cloud. Topics mentioned include encrypting outsourced data, as well as the authentication and authorization of users accessing the outsourced data.

Encrypting Outsourced Data: To protect the confidentiality of information being outsourced from UAS networks to systems in the cloud, the data should be obfuscated prior to being transmitted to the cloud. In an article by Lin et al. two privacy-related challenges in the IoD were described: location/identity privacy and outsourced data privacy [55]. For the first challenge, lightweight and efficient symmetric key encryption algorithms, key management systems, and the utilization of zero-knowledge proofs can be potential solutions. For the latter challenge, they propose a lightweight identity-based encryption scheme utilizing both asymmetric and symmetric key cryptography to provide data privacy, while still allowing flexible access to the stored data when needed. Additionally, Xu and Zhu proposed a mechanism to support the encryption of data being outsourced to the cloud from networked control systems and verify the integrity of results computed from the cloud [109].

Authentication & Authorization: Similar to encrypting the outsourced data, protections need to be put in place in order to prevent unauthorized entities from accessing the data. One example of this includes work from Baboolal et al., where the authors proposed a proxy re-encryption technique for drones storing videos on the cloud [11]. Their scheme provides a one-time key for accessing videos, where key management is performed by a trusted control center. Through utilizing a key management and distribution scheme, only authorized entities can access data stored on a remote server.

Additionally, Wazid et al. proposed an authentication scheme in which IoD/ UAV users need to access data directly from UAV providers [107]. Either the server in the cloud or a GCS is responsible for registering each UAV prior to deployment. The scheme proposed by the authors includes authentication and key agreement protocols, in addition to password and biometric updates. UAV key management is required for secure communication between UAVs via pairwise keys established between neighboring UAVs. This scheme is effective against privileged insider and offline password guessing attacks, user/server/UAS impersonation attacks, denial of service, and much more.

6.2 Mitigation Strategies Found in Standards

The following table describes mitigation strategies for attacks described in Tables 10-14 NIST's Special Publication series and others were used to present standard mitigation strategies for UAS security system. Here is a list of publications used for UAS security guidelines, recommendations, and standard specifications:

- NISTIR 8323: Foundational PNT Profile: Applying the Cybersecurity Framework for the Responsible Use of Positioning, Navigation, and Timing (PNT) Services [68]
- NIST Special Publication 800-193: Platform Firmware Resiliency Guidelines [89]
- NIST Special Publication 800-161: Supply Chain Risk Management Practices for Federal Information Systems and Organizations [46]
- NIST Special Publication 800-44: Guidelines on Securing Public Web Servers [69]
- NISTIR 7682: Information System Security - Best Practices for UOCAVA- Supporting Systems [8]
- NIST Special Publication 800-147: BIOS Protection Guidelines [21]
- NIST Special Publication 800-53: Security and Privacy Controls for Information Systems and Organizations [44]
- NIST Special Publication 800-218: Secure Software Development Framework (SSDF) - Recommendations for Mitigating the Risk of Software Vulnerabilities [77]
- NIST Special Publication 800-83: Guide to Malware Incident Prevention and Handling for Desktops and Laptops [76]
- ICS Advisory (ICSA-19-015-01) [17]
- NIST Special Publication 800-63B: Digital Identity Guidelines [84]
- NIST Special Publication 800-189: Resilient Interdomain Traffic Exchange- BGP Security and DDoS Mitigation [51]
- NISTIR 8301: Blockchain Networks - Token Design and Management Overview [58]
- NIST Interagency report 7316: Assessment of Access Control System [105]
- NIST Special Publication 800-123: Guide to General Server Security [47]
- NIST Special Publication 800-63-3: Digital Identity Guidelines [86]
- NIST CVE-2017-12819: National Vulnerability Database [81]
- NISTIR 8397: Guidelines on minimum standards for developer verification of software [83]
- NIST Special Publication 800-95: Guide to Secure Web Services [9]
- NIST Special Publication 800-53B: Control Baselines for Information Systems and Organizations [45]

- NIST Special Publication 800-63A: Digital Identity Guidelines Enrollment and Identity Proofing Requirements [85]

Table 22. Attacks, Frameworks, and Mitigation Strategies.

Attack Reference	**Framework**	**Mitigation Strategy**
UAV Hardware Attack		
HW-S/*	NISTIR 8323	Identity verification, data verification, and validation of Positioning, Navigation, and Timing (PNT) components.
HW-J/*	NISTIR 8323	Software and hardware can be integrated into the system and critical infrastructure components to detect and mitigate GNSS jamming and spoofing events and preserve data availability, continuity, and integrity.
HW-FF	NIST SP 800-193	Firmware update images should be signed using an approved digital signature algorithm. The flash regions that contain device firmware should be protected so that it is modifiable only through an authenticated update mechanism to ensure the authenticity and integrity of the firmware update. The protection mechanisms shall ensure that authenticated update mechanisms are not bypassed. If Critical Platform Firmware uses RAM for temporary data storage, then this memory shall be protected from software running on the Platform until the data's use is complete.
HW-SCA	NIST SP 800-161	Establish an organization governance structure that ICT SCRM requirements and incorporates these requirements into the organizational policies. Perform internal checks and balances to assure compliance with security and quality requirements.
UAV Software attack		
SW-CI	NIST SP 800-44	Use secure programming practices and maintain secure configuration through application. Software, OS, web servers, firewalls, packet filtering routers and proxy should be periodically scanned for vulnerability.
SW-DI	NISTIR 7682	Check the values in every field of a web form, looking for any characters that should not be in that type of data, and looking for patterns that look like database commands. Monitor the logs of the database server, looking for anomalous queries coming from the web server.
SW-FM	NIST SP 800-147	Use digital signatures for secure BIOS authentication. Authenticated BISO update mechanism should be an exclusive

		mechanism for modification of system BIOS with proper authentication mechanism.
SW-BD	NIST SP 800-53	Use an uninterruptible power supply (UPS) that provides emergency power when there is a failure of the main power source. The battery duration of most UPS is short but provides sufficient time to start a standby power source such as a backup generator or properly shut down the system or perform emergency procedures.
SW-BO	NIST SP 800-218	Collect, protect, and regularly check provenance data for all software deployed in each environment, and determine if any of the software or their dependencies have new known vulnerabilities. Review and approve all changes made to the code after the code has been automatically scanned for vulnerabilities and any issues have been remediated. Periodically scan the software for buffer overflow flaws. Review and evaluate third-party software components in the context of their expected use.
SW-MI	NIST SP 800-83	Scanning of media from outside of the organization for malware before they can be used. Restricting or prohibiting the use of unnecessary software, such as user applications that are often used to transfer malware. Using security automation technologies with OS and application configuration checklists to help administrators secure hosts consistently and effectively.
SW-SCA	NIST SP 800-161	Similar to HW-SCA
Ground Control System (GCS) Attack		
GCS-RA	NIST SP 800-53	Employ automated mechanisms to facilitate the monitoring and control of remote access methods. Uses encryption to protect the confidentiality of remote access sessions.
GCS-FQA	NIST Special Publication 800-83	Use antivirus software, intrusion prevention software, firewall, content filtering/inspection and application whitelisting.
GCS-DE	ICS Advisory (ICSA-19-015-01)	Minimize network exposure for all control system devices and/or systems and ensure that they are not accessible from the Internet.
GCS-PB	NIST SP 800-53	Use secure passwords and passphrase. Passwords should have minimum length and be followed by either biometric

		authentication or two factor authentication. Stored passwords should be using an approved salted key derivate function, preferably a keyed hash.
GCS-RE	NIST CYBERSECURITY WHITE PAPER on Mitigating the Risk of Software Vulnerabilities by Adopting a Secure Software Development Framework (SSDF)	Perform peer review of code, to check code for backdoors and other malicious content. Use automated tools to identify and remediate documented and verified unsafe software practices on a continuous basis as human-readable code is checked into the code repository.
GCS-SE	NIST Special Publication 800-63B	Avoid use of authenticators that present a risk of social engineering of third parties such as customer service agents.
Network Link Attack		
NL-BH/GH	NIST Special Publication 800-189	Monitor the rate of queries/requests per source address and detect if an abnormally high volume of responses is headed to the same destination (i.e., same IP address).
NL-W	NIST SP 800-189	Similar to NL-BL/GH
NL-Syb	NISTIR 8301	Sybil attack resistance is achieved, respectively, through built-in crypto economic incentives that enable nodes to work together in zero-trust environments and through access control, wherein nodes must be authorized by system owners or consortium members.
NL-Sink	NIST Special Publication 800-83	Different situations necessitate various combinations of eradication techniques. The most common tools for eradication are antivirus software, spyware detection and removal utilities, and patch management software. Providing instructions and software updates to users works in some cases.
NL-RFJa	NISTIR 8323	Similar to HW-J/*

NL-PBJa	NISTIR 8323	Similar to HW-J/*
NL-D	NIST Interagency report 7316	Use an access control list and access control matrix. Implement Separation of duty (SOD) where no user should be given enough privileges to misuse the system.
NL-PS/A	NIST Special Publication 800-83	Use antivirus, firewalls, application whitelisting sandboxing techniques. Eliminating unsecured file shares, which are a common way for malware to spread.
NL-PB	NIST Special Publication 800-63B	Use an authenticator with high entropy authenticator secret. Store memorized secrets in a salted, hashed form including a keyed hash.
NL-PitM	NIST Special Publication 800-63B	Communication between the claimant and verifier should be via an authenticated protected channel to provide confidentiality of the authenticator output.
NL-CJ	NIST SP 800-44	Similar to SW-CI
NL-M	NIST SP 800-123	Remove or disable unneeded default accounts, disable non interactive accounts. Create user groups, configure automated time synchronization. Implement strong organization password policy.
NL-ReplayA	NIST SP 800-63-3	Use nonce that is used as challenge in challenge-response authentication protocol that are not repeated.
NL-RelayA	NIST CVE-2017-12819	Locate control system networks and remote devices behind firewalls and isolate them from the business network. When remote access is required, use secure methods, such as Virtual Private Networks (VPNs), recognizing that VPNs may have vulnerabilities and should be updated to the most current version available.
NL-F	NISTIR 8397, NIST SP 800-95, and NIST SP 800-53B	A type of dynamic analysis, known as fuzz testing, induces program failures by deliberately introducing malformed or random data into software programs. Fuzz testing strategies are derived from the intended use of applications and the functional and design specifications for the applications. To understand the scope of dynamic code analysis and the assurance provided, organizations may also consider conducting code coverage analysis and/or concordance analysis.
Server Attack		

SRV-DL	ICS Advisory (ICSA-19-015-01)	Similar to GCS-DE
SRV-PIL	NIST Special Publication 800-63A	Use a Credential Service Provider (CSP) that validates personal details in the evidence with the issuer or other authoritative source. It verifies identity evidence and biometric of applicant against information obtained from issuer or other authoritative source.
SRV-LL	ICS Advisory (ICSA-19-015-01)	Similar to GCS-DE

7 CYBERSECURITY USE CASES: THREAT PROFILES

In this section, the team will delve into the characteristics of use cases described in Section 2.3 from the view of cybersecurity to develop how they are related to existing attacks and defenses. Depending on the types of missions that UAS should fulfill, each use case has different aspects - phases of operation, type of UAS, collaboration with other UAS, etc. Among them, the team has chosen three major cybersecurity attributes and two minor cybersecurity attributes to distinguish use case groupings with similar operational movements. Due to the groupings' differences in muscle movements, these use case groupings (or *cybersecurity use cases*) will also be shown to have differences in attack surface.

7.1 Major Cybersecurity Attributes

In order to build a set of use case groupings which have different cybersecurity implications, the first step should be to determine what these implications look like. In previous sections, it was established that the possibilities of UAS use cases is immense, and that UAS use case categorization should be accomplished through the use of UAS muscle movements. Therefore, in this subsection, the use case attributes which are derived from UAS muscle movements and can create displacements in the UAS attack surface will be identified.

7.1.1 UAS Autonomy

Autonomous: Having a UAS or UAS swarm be considered "autonomously operated" would imply that the system is able to rely on controller operations synthesized from the system itself in order to complete a mission, given some amount of pre-loaded information. In other words, the system is able to use sensed information from the environment to control itself, and ultimately control itself to complete its predefined mission without the need for human control. There exists the capability for a human to intervene and override its autonomous controls with human operations, however the system should be capable of completing the mission, from take-off to landing, without the need for human operations. As long as there are no commands being sent to the system, the only network link activity between the GCS and UAS/swarm would be the retrieval of sensor readings or payload data (e.g., camera pictures). The use cases where autonomous control can be incorporated include tasks which can be easily automated, and typically only incorporate passive

interaction with the environment - although autonomous UAS (or UAS swarms) with active environment interaction is possible.

Manual: Having a UAS or UAS swarm be considered "manually operated" implies that within the system, the operations received by the UAS controller are sent by a human operator, most likely over a network link between a GCS and the UAS. There exists no autonomous capabilities within the system. The use cases associated with manual control typically involve missions where delicate control of the UAS and/or expert knowledge in performing a given action is essential to the completion of the mission.

7.1.2 UAS Operational Range

BVLOS: When a UAS or UAS swarm use case is considered "beyond visual line of sight" (BVLOS), then the system has the capability of being controlled at a distance no longer visible to the operators. Incorporating BVLOS operations might entail the use of a specific network link protocol (e.g., SATCOM or 4G-LTE) to accommodate the potentially large distance between the UAS and GCS. Use cases which are BVLOS typically involve missions where the flight plan covers a wide area or has waypoints which are far away from each other.

VLOS: When a UAS or UAS swarm is considered within "visual line of sight" (VLOS), then the system will remain visible to the operator for the entirety of the mission. The network links used during a mission where the UAS (or UAS swarm) will be within VLOS do not need to accommodate long-distance communications. Use cases which are considered VLOS typically only have a single geographical point which is both essential to the mission and near the take-off point.

7.1.3 UAS Collaboration

Single: When a UAS is considered as a “single” system, then it's assumed that the use case can be completed by an individual UAS. Hence, there exists a single network link between the UAS and the GCS. Use cases which only need a single drone typically aren't complex and don't require a UAS to cover long distances.

Swarm: When a UAS is considered as a "swarm" system, then the use case can potentially be completed by more than one UAS acting together - this is referred to as a UAS swarm. In a swarm environment, there exists network links both between any number of UASs and the GCS, as well as links between UASs. This creates a flying ad-hoc network (FANET), where the UASs can be used to route network packets between other UASs. Use cases which can incorporate swarms involve several sub-tasks to complete a given mission, where each UAS within a swarm can be assigned a certain sub-task. Additionally, use cases for swarms can cover repetitive actions over a wide geographical area, where each UAS can be assigned to a specific area. In some cases, these use cases can be completed by a single drone, however it would likely take much longer than it would if a UAS swarm was utilized.

7.2 Minor Cybersecurity Attributes

There exists additional attributes associated with UAS use cases, also derived from UAS muscle movements, which do not play as significant of a role as the previously depicted attributes. These are described in this subsection, being defined as “minor attributes.”

UAS Customization: Some use cases might require a UAS to be equipped with specialized payloads to execute the given mission, rather than using an unmodified COTS UAS. If a UAS needs to actively interact with its environment (e.g. ultrasonic testing), it would be more likely to include customized components as well. It is also plausible that COTS UAS is customized by the user or is professionally tuned by an intermediate supplier. Depending on the level of customization and the integration with UAS, it might include more cybersecurity vulnerabilities, because the UAS manufacturers cannot guarantee the security for those highly customized parts.

UAS Mission Criticality: This attribute indicates whether human life is under critical situation if the UAS fails to complete its given mission within the required timeframe. In those cases, any cybersecurity attacks that can trigger the delay could lead to the loss of human life. Because a fatal accident might happen in any case if a UAS crashes into the ground, it is not deemed that such a use case is life-critical to emphasize the specifics of use cases.

7.3 Cybersecurity Use Cases

As a preliminary effort, the team has looked at each use case from Section 2.3 and labeled them with relevant attributes, based on the cybersecurity attributes defined previously. The results are described in this subsection, specifically in the following tables shown, where the use cases with the same major cybersecurity attributes are grouped together. We understand that drawing a boundary whether an attribute holds or not is not clear in many cases because the utilization is up to the end user and thus can have different set-up even for similar jobs. Many "mixed" situations can exist if the attribute holds in one example and doesn't in another. To reduce the complexity, these "mixed" cases are merged with either one of two possible choices: "Autonomous" in autonomy, "BVLOS" in operation range, and "Swarm" in collaboration field.

The researchers expect this table will change over time, as technologies regarding UAS are evolving. As UAS will be more widely adopted in many different industries, the team expects that the trend will be toward "Autonomous/BVLOS/Swarm" to fulfill various types of missions, handle various situations, and replace traditional methods in wider areas.

7.3.1 Autonomous, BVLOS, and Swarm

In this cybersecurity use case, the UAS are in a swarm formation, and are autonomously operated outside the line of sight of the pilot. Use cases in this grouping likely entail repetitive and easily performed actions over a large area.

Some UAS attacks that may be especially impactful for this use case include:

- *Sensor spoofing/jamming*: These attacks may impact the data that the UAS receives for the autopilot, and that the GCS receives during the mission.
- *Battery draining*: For BVLOS operations, this attack could make the UAS run out of battery before completing its mission, and while far away from the GCS.
- *Database injection*: Depending on the information stored in the database (e.g., sensor readings, swarm network information), this could be especially critical for autonomous, BVLOS and swarm use cases.
- *Network routing attacks*: The incorporation of a FANET network in these use cases increases the attack surface of these use cases. Routing attacks such as blackhole, grayhole, wormhole, sybil, sinkhole, and masquerading can especially impact swarm networks.

Since this cybersecurity use case encompasses a great portion of the use cases listed in previous sections and appears to be the direction that modern use cases are approaching, this cybersecurity use case will be broken down into subgroups. These subgroups include:

1. Searching, exploring, inspecting, and assessing with a mounted camera
2. Entertainment and environmental interaction
3. Shipping, storage, and delivery

The subgroups are in the order given from the subgroup list, for example: the first subgroup table corresponds to the first element in the subgroup list, which is "searching, exploring, inspecting, and assessing with a mounted camera." These cybersecurity use cases can be found in Tables 27-29 under Appendix A: Cybersecurity Use Case Groupings.

7.3.2 Autonomous, BVLOS, and Single

In this cybersecurity use case, there is only one UAS which is autonomously operated outside the line of sight of the pilot. Use cases in this grouping likely entail repetitive and easily performed actions over a large area.

Some UAS attacks that may be especially impactful for this use case include:

- *Sensor spoofing/jamming*: These attacks may impact the data that the UAS receives for the autopilot, and that the GCS receives during the mission.
- *Battery draining*: For BVLOS operations, this attack could make the UAS run out of battery before completing its mission, and while far away from the GCS.
- *Database injection*: Depending on the information stored in the database (e.g., sensor readings), this could be especially critical for autonomous and BVLOS use cases.

This cybersecurity use case also includes a large portion of the previously mentioned use cases, so this will also be broken into subgroups. These subgroups are the same as was mentioned for the previous cybersecurity use case:

1. Searching, exploring, inspecting, and assessing with a mounted camera
2. Entertainment and environmental interaction
3. Shipping, storage, and delivery

Similar to the previous cybersecurity use case, the subgroups are in the order given from the subgroup list. These cybersecurity use cases can be found in Tables 30-32 under Appendix A: Cybersecurity Use Case Groupings.

7.3.3 Autonomous, VLOS, and Swarm

In this cybersecurity use case, the UAS are in a swarm formation, and are autonomously operated within the line of sight of the pilot. Use cases in this grouping likely entail repetitive and easily performed actions within a small range of the GCS.

Some UAS attacks that may be especially impactful for this use case include:

- *Sensor spoofing/jamming*: These attacks may impact the data that the UAS receives for the autopilot, and that the GCS receives during the mission.
- *Database injection*: Depending on the information stored in the database (e.g., sensor readings, swarm network information), this could be especially critical for autonomous and swarm use cases.

- *Network routing attacks*: The incorporation of a FANET network in these use cases increases the attack surface of these use cases. Routing attacks such as blackhole, grayhole, wormhole, sybil, sinkhole, and masquerading can especially impact swarm networks.

These cybersecurity use cases can be found in Table 33 under Appendix A: Cybersecurity Use Case Groupings.

7.3.4 Autonomous, VLOS, and Single

In this cybersecurity use case, a single UAS is autonomously operated within the line of sight of the pilot. Use cases in this grouping likely entail repetitive and easily performed actions in a small area, close to the GCS.

Some UAS attacks that may be especially impactful for this use case include:

- *Sensor spoofing/jamming*: These attacks may impact the data that the UAS receives for the autopilot, and that the GCS receives during the mission.
- *Database injection*: Depending on the information stored in the database (e.g., sensor readings), this could be especially critical for autonomous use cases.

These cybersecurity use cases can be found in Table 34 under Appendix A: Cybersecurity Use Case Groupings.

7.3.5 Manual, BVLOS, and Swarm

In this cybersecurity use case, the UAS are in a swarm formation, and are manually operated outside the line of sight of the pilot. Use cases in this grouping likely entail complex actions and procedures over a large area.

Some UAS attacks that may be especially impactful for this use case include:

- *Sensor spoofing/jamming*: These attacks may impact the data that the GCS receives when the UAS swarm is BVLOS, which can fool the pilot into making incorrect decisions when controlling the UAS swarm.
- *Battery draining*: For BVLOS operations, this attack could make a UAS run out of battery before completing its mission, and while far away from the GCS.
- *Force quitting application*: An attacker with access to the GCS could force-quit the GCS application, cutting the network link to the UAS swarm.
- *Database injection*: Depending on the information stored in the database (e.g., swarm network information), this could be especially critical for BVLOS and swarm use cases.
- *Network routing attacks*: The incorporation of a FANET network in these use cases increases the attack surface of these use cases. Routing attacks such as blackhole, grayhole, wormhole, sybil, sinkhole, and masquerading can especially impact swarm networks.
- *Network DoS*: Disrupting the network link to the UAS swarm could cause the swarm to lose control or crash.

These cybersecurity use cases can be found in Table 35 under Appendix A: Cybersecurity Use Case Groupings.

7.3.6 Manual, BVLOS, and Single

In this cybersecurity use case, a single UAS is manually operated outside the line of sight of the pilot. Use cases in this grouping likely entail complex actions and procedures over a large area.

Some UAS attacks that may be especially impactful for this use case include:

- *Sensor spoofing/jamming*: These attacks may impact the data that the GCS receives when the UAS is BVLOS, which can fool the pilot into making incorrect decisions when controlling the UAS.
- *Battery draining*: For BVLOS operations, this attack could make a UAS run out of battery before completing its mission, and while far away from the GCS.
- *Force quitting application*: An attacker with access to the GCS could force-quit the GCS application, cutting the network link to the UAS swarm.
- *Network DoS*: Disrupting the network link could cause the UAS to lose control or crash.

These cybersecurity use cases can be found in Table 36 under Appendix A: Cybersecurity Use Case Groupings.

7.3.7 Manual, VLOS, and Swarm

In this cybersecurity use case, the UAS are in a swarm formation, and are manually operated within the line of sight of the pilot. Use cases in this grouping likely entail complex actions and procedures over a large area, but within a short distance to the GCS.

Some UAS attacks that may be especially impactful for this use case include:

- *Force quitting application*: An attacker with access to the GCS could force-quit the GCS application, cutting the network link to the UAS swarm.
- *Database injection*: Depending on the information stored in the database (e.g., swarm network information), this could be especially critical for BVLOS and swarm use cases.
- *Network routing attacks*: The incorporation of a FANET network in these use cases increases the attack surface of these use cases. Routing attacks such as blackhole, grayhole, wormhole, sybil, sinkhole, and masquerading can especially impact swarm networks.
- *Network DoS*: Disrupting the network link to the UAS swarm could cause the swarm to lose control or crash.

These cybersecurity use cases can be found in Table 37 under Appendix A: Cybersecurity Use Case Groupings.

7.3.8 Manual, VLOS, and Single

In this cybersecurity use case, a single UAS is manually operated within the line of sight of the pilot. Use cases in this grouping likely entail complex actions and procedures within a small area, close to the GCS.

Some UAS attacks that may be especially impactful for this use case include:

- *Force quitting application*: An attacker with access to the GCS could force-quit the GCS application, cutting the network link to the UAS swarm.
- *Network DoS*: Disrupting the network link to the UAS swarm could cause the swarm to lose control or crash.

These cybersecurity use cases can be found in Table 38 under Appendix A: Cybersecurity Use Case Groupings.

8 CYBERSECURITY USE CASES: MITIGATING THREATS

The previous sections have explored potential UAS use cases, the definitions of cybersecurity use cases, and the differences in impact between different types of cyber-attacks among the cybersecurity use cases. This section will expand on previous sections by describing how these attacks can be used to accomplish tasks which threaten the NAS, and how these attacks can be mitigated. This will be shown by grouping UAS cyber-attacks by how they impact the UAS, and how the resulting state of the UAS might pose a hazard to the NAS. The three impacts that were considered include crashing the UAS, hijacking the UAS, and exfiltrating or eavesdropping sensitive information from a GCS or third-party server.

These three impacts have direct comparisons to the threats to NAS that were covered in Section 4.6. These threats include performing hostile surveillance, smuggling contraband, disruption of government services, and weaponization of the UAS. Under the pretense of exploring threats that occur as an impact from a cyberattack performed on a UAS, this can rule out smuggling of contraband. The remaining three threats to NAS will be mapped to the three UAS impacts from cyber-attacks, and the methods for performing/mitigating the associated attacks will also be explored. Specifically, the attacks mentioned in this section are derived from Section 4.3, and the mitigation methods are derived from Section 6.1.

8.1 UAS Eavesdropping & Data Exfiltration

With respect to the previously mentioned threats to NAS, attacks which eavesdrop on UAS network links and exfiltrate data from the GCS or a third-party server entail hostile surveillance if the data being transferred/held is video or image files taken from the UAS. This has the potential to pose a threat to the NAS, as the data leaked from the network link might include surveillance footage of private NAS-related infrastructure or areas.

In the case of eavesdropping communications, an attacker could perform packet sniffing/analysis on a network link to identify relevant communication data. If this data is a video feed or a camera roll, and it is not encrypted (or is weakly encrypted), then the attacker might be able to extract surveillance data from the network link. In the case of data exfiltration from the GCS or a third-party server, unauthorized access would need to take place, which would allow an attacker access to (surveillance) data. Once this unauthorized access is established, an attacker can use an available network link to exfiltrate the data to a server that they own. Again, if this data is unencrypted or weakly encrypted, then this could allow an attacker access to the surveillance data.

To mitigate against these attacks, network links between the UAS and other entities should be encrypted using strong encryption algorithms. Additionally, methods for authenticating and authorizing users to access private data should be used to prevent unauthorized access to the surveillance data. Finally, software protection should be put in place at the GCS to prevent attackers from gaining unauthorized access. Table 23 provides a list of previously mentioned attacks and their associated mitigation methods.

Table 23. UAS Eavesdropping & Data Exfiltration Attacks & Mitigations.

Attack Category	Attack Method	Mitigation Method

Network	• Packet Sniffing/Analysis	• Encrypted & authenticated communications
GCS	• Data exfiltration	• Software protections • Authentication & authorization
Server	• Data leakage • Pilot identity leakage • Location leakage	• Encrypting outsourced data • Authentication & authorization

8.2 UAS Crashing & Loss of Control

A cyber-attack that causes a UAS to crash or lose control has the potential to disrupt government services. One example could be a benign UAS losing control next to or within controlled airspace, such as an airport, which would create a hazard for other airspace users and temporarily cease airport operations. Additionally, a cyber-attack that causes a UAS to lose control or crash could be seen as a method for weaponizing the UAS, as selectively choosing where to cause the UAS to lose control could allow an attacker to control where the UAS crashes. An example of this could include an attacker waiting for a UAS to fly by a NAS-critical building or piece of infrastructure and causing it to crash then. Thus, disrupting government services and UAS weaponizations are possibilities that can occur after a cyber-attack is performed on a UAS, where the focus of the attack is to crash the UAS or cause it to lose control.

Cyber-attacks which might cause a UAS to lose control or crash typically fall under the realm of DoS-style attacks. Hardware attacks include those that attempt to jam GPS, ADS-B, Remote ID, actuators, and other sensors. Software and GCS attacks focus on higher level applications, and can include buffer overflows, database injections, and battery draining on the UAS, as well as forced quitting the application on the GCS. Network attacks can include routing attacks (e.g., blackhole, grayhole, wormhole, sinkhole, sybil), and jamming attacks (RF/Protocol jamming and fuzzing). Table 24 provides a list of attacks which could potentially cause a UAS to crash or lose control, as well as mitigation methods to protect against these attacks.

To mitigate against these attacks, a variety of methods should be used to protect each component of the UAS. For hardware protection, ensure the signals being received by the sensors, actuators and controllers are valid by incorporating a hardware-based IDS, controller security functions, and sensor fusion. Software components should validate the integrity of application-level behaviors through a software-based IDS, and separate application execution by utilizing software isolation. Network routing attacks can be prevented through secure routing protocols, blockchain technologies and trust models. Additionally, network jamming/DoS attacks and fuzzing can be mitigated by using encrypted and authenticated communications, trust models and a network link IDS. Finally, unauthorized access to the GCS can be mitigated by using software protections and incorporating authentication and authorization. Table 24 provides a list of previously mentioned attacks and their associated mitigation methods.

Table 24. UAS Crashing Attacks & Mitigations.

Attack Category	**Attack Method**	**Mitigation Method**

Hardware	• GPS jamming • ADS-B / Remote ID jamming • Actuator jamming • Other sensor jamming	• Hardware IDS • Controller security • Sensor fusion
Software	• Buffer overflow • Database injection • Battery draining	• Software IDS • Software isolation
GCS	• Forced quitting application	• Software protections • Authentication & authorization
Network	• Blackhole • Grayhole • Wormhole • Sybil • Sinkhole	• Secure routing protocols • Blockchain tech • Trust models
Network	• RF/Protocol Jamming • Deauthentication	• Network link IDS • Encrypted & authenticated communications • Trust models
Network	• Fuzzing	• Network link IDS

8.3 UAS Hijacking

Since the outcome of UAS hijacking can take so many directions, it's assumed that an attacker could perform all three previously mentioned threats to NAS: perform hostile surveillance, disrupt government services, and potentially become weaponized. Given that a UAS has been hijacked, some examples of these threats include a camera-mounted UAS sending its video feed to an attacker-controlled server instead of a GCS, a UAS being hijacked to fly recklessly around a NAS-critical area (e.g., an airport), or a UAS being hijacked to crash into a building. One aspect that exists in the hijacking scenario that doesn't exist in others is the possibility that an attacker might upload malicious tools to the UAS in order to perform proximity-based cyber-attacks. In another example, a hijacked UAS could be ordered to sit outside a NAS-critical building and be used as a proxy to attempt infiltration of the building's wireless computer network.

Attacks which may provide an attacker with the ability to hijack a UAS typically include data spoofing or modification, or already having direct control of the UAS via an installed backdoor. GPS, ADS-B, actuator, and other sensor spoofing can be utilized by an attacker to potentially have partial (or even complete) control over a UAS's movements, especially when sensor readings are directly used to control the UAS (e.g., when sensor readings are fed into an autopilot program). Additionally, a UAS infected with malware could host a backdoor, giving an attacker remote access to the drone. Firmware flashing/modification and supply chain attacks may also embed a hidden backdoor into the hardware/software components of a UAS. Furthermore, if an attacker is capable of remotely accessing the GCS, then they could potentially hijack the UAS by sending commands to the UAS while disallowing a legitimate user from entering controls. If the attacker

has access to the GCS application, an attacker could also reverse engineer the binary and exploit the inner workings of the program to take control of the UAS. Finally, in the context of the UAS network-link, attacks such as person-in-the-middle, command injection, replay/relay attacks, and masquerading may allow for an attacker to send spoofed or modified controls to the UAS.

Mitigation methods for preventing UAS hijacking seem to involve verifying the integrity of data received by the UAS, the integrity of the UAS's configurations, and using component-based IDS for detecting intrusions. Spoofing attacks which target hardware components such as sensors and actuators can be prevented through incorporating controller security and sensor fusion and using a hardware-based IDS. Additionally, a software-based IDS, as well as utilizing remote attestation or software isolation, can assist in mitigating UAS malware infections. Using a hardware and/or software IDS, performing hardware verification and remote attestation can also assist in mitigating hardware-based and software-based firmware modifications and supply chain attacks. Encrypting and authenticating network link traffic and using a network-link IDS can mitigate a variety of network-link attacks, such as the ones mentioned previously. Finally, unauthorized remote access to the GCS can be mitigated through the use of software protections and authentication/authorization mechanisms, while GCS application reverse engineering can be made more difficult through packing or obfuscating the GCS binary. Table 25 provides a list of previously mentioned attacks and their associated mitigation methods.

Table 25. UAS Hijacking Attacks & Mitigations.

Attack Category	**Attack Method**	**Mitigation Method**
Hardware	● GPS spoofing ● ADS-B / Remote ID spoofing ● Actuator spoofing ● Other sensor spoofing	● Hardware IDS ● Controller security ● Sensor fusion
Hardware / Software	● Firmware flashing/modification ● Supply chain attack	● Hardware/Software IDS ● Hardware verification ● Remote attestation
Software	● Malware infection	● Software IDS ● Remote attestation ● Software isolation
GCS	● Remote access	● Software protections ● Authentication & authorization
GCS	● Reverse Engineering GCS Application	● Packing/Obfuscating GCS application
Network	● Person-In-The-Middle ● Command Injection ● Replay/Relay attack Masquerading	● Network link IDS ● Encrypted & authenticated communications

8.4 Other Attacks

This final category of attacks doesn't have explicit links to creating hazards for the NAS, but they are worth mentioning as they may provide attack vectors for adversaries to exploit. In the context of the GCS, password breaking, and social engineering could allow an attacker physical or remote access to the device, which could allow them to crash or hijack the UAS, or even exfiltrate surveillance data (if it exists on the device). Additionally, network-based password breaking might give an adversary access to the network-link that is used for sending/receiving controls and data between UAS(s) and the GCS. This opens the network entities to a variety of remote attacks, such as UAS hijacking or crashing, or hostile surveillance. These attacks, as well as their countermeasures, are listed in the Table 26.

Table 26. Other UAS Attacks & Mitigations.

Attack Category	**Attack Method**	**Mitigation Method**
GCS	● Password Breaking ● Social Engineering	● Authentication and authorization
Network	● Password Breaking	● Network services best practice

9 CONCLUSIONS

This project undertook a literature review to identify cybersecurity risks to UAS and their integration into the NAS. As part of this review the project surveyed and documented 128+ UAS use cases (Section 2) and organized them to eight main categories using three attributes relevant to cybersecurity and phases of UAS operation (Section 7). A preliminary threat profile for each category (Section 7) was also developed. The project also surveyed common UAS platforms covering hardware, software, communication, and coordination protocols, as well as commercially available components used for construction of sUAS (Section 3).

The project identified and cataloged nearly 1300 published articles and undertook a detailed review of nearly 550 of them to identify and categorize potential cybersecurity risks to UAS and their integration. To provide a starting point, the project identified 41 different cybersecurity threats and organized them into five categories representing the key UAS components they target (Section 4). This project also leveraged the FAA Safety Management System framework and UAS Operational Phases to provide a preliminary assessment of the cybersecurity risks identified. The project also identified and documented potential mitigations against the 41 cybersecurity threats catalogued through both a survey of academic articles and by reviewing the available cybersecurity standards (Section 6). This report also includes a preliminary assessment of the impact of UAS cyber threats to the NAS (Section 4) and some potential mitigations against them (Section 8).

While the project covered a lot of ground in a short amount of time, the assessment of cybersecurity risks and the identification of mitigation measures is necessarily preliminary and

need to be treated as such. A more thorough and detailed assessment of cybersecurity risks and their impact needs to be undertaken to help FAA better manage the risks associated with integration of UAS.

In summary, the outcomes from this project provide baseline information on cybersecurity threats to UAS and the risks from their integration into the NAS. The outcomes from this work are directly relevant to the efforts in other research projects that are studying automation failures in UAS including cyber induced and demonstrating the need for a cybersecurity oversight and risk management in UAS.

10 REFERENCES

[1] A. VanHoudt, Z. LaRue, S. Hottman, and H. Cathey. 2017. FAA Interim Technical Report: Small Unmanned Aircraft Systems Use Cases and Detect and Avoid Approaches.

[2] F. Abdi, C. Chen, M. Hasan, S. Liu, S. Mohan, and M. Caccamo. 2018. Guaranteed Physical Security with Restart-Based Design for Cyber-Physical Systems. In *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, 10–21. DOI:https://doi.org/10.1109/ICCPS.2018.00010

[3] S. Aggarwal, M. Shojafar, N. Kumar, and M. Conti. 2019. A New Secure Data Dissemination Model in Internet of Drones. In *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 1–6. DOI:https://doi.org/10.1109/ICC.2019.8761372

[4] D. J. S. Agron, M. R. Ramli, J. Lee, and D. Kim. 2019. Secure Ground Control Station-based Routing Protocol for UAV Networks. In *2019 International Conference on Information and Communication Technology Convergence (ICTC)*, 794–798. DOI:https://doi.org/10.1109/ICTC46691.2019.8939885

[5] Mohammad Al Faruque, Francesco Regazzoni, and Miroslav Pajic. 2015. Design Methodologies for Securing Cyber-Physical Systems. In *Proceedings of the 10th International Conference on Hardware/Software Codesign and System Synthesis* (CODES '15), IEEE Press, 30–36.

[6] A. Allouch, O. Cheikhrouhou, A. Koubâa, M. Khalgui, and T. Abbes. 2019. MAVSec: Securing the MAVLink Protocol for Ardupilot/PX4 Unmanned Aerial Systems. In *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, 621–628. DOI:https://doi.org/10.1109/IWCMC.2019.8766667

[7] Moreno Ambrosin, Mauro Conti, Riccardo Lazzeretti, Md Masoom Rabbani, and Silvio Ranise. 2017. Toward Secure and Efficient Attestation for Highly Dynamic Swarms: Poster. In *Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks* (WiSec '17), Association for Computing Machinery, New York, NY, USA, 281–282. DOI:https://doi.org/10.1145/3098243.3106026

[8] Andrew Regenscheid, Geoff Beier, Santosh Chokhani, Paul Hoffman, Jim Knoke, and Scott Shorter. 2011. NISTIR 7682: Information System Security - Best Practices for UOCAVA-Supporting Systems. Retrieved from https://csrc.nist.gov/publications/detail/nistir/7682/final

[9] Anoop Singhal, Theodore Winograd, and Karen Scarfone. 2007. NIST Special Publication 800-95: Guide to Secure Web Services. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-95/final

[10] N. Asokan, Ferdinand Brasser, Ahmad Ibrahim, Ahmad-Reza Sadeghi, Matthias Schunter, Gene Tsudik, and Christian Wachsmann. 2015. SEDA: Scalable Embedded Device Attestation. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (CCS '15), Association for Computing Machinery, New York, NY, USA, 964–975. DOI:https://doi.org/10.1145/2810103.2813670

[11] Vashish Baboolal, Kemal Akkaya, Nico Saputro, and Khaled Rabieh. 2019. Preserving Privacy of Drone Videos Using Proxy Re-Encryption Technique: Poster. In *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks* (WiSec '19), Association for Computing Machinery, New York, NY, USA, 336–337. DOI:https://doi.org/10.1145/3317549.3326319

[12] Henry Cathey. 2019. Development of an Operational Framework for Small UAS Beyond Visual Line of Sight (BVLOS) Operations—New Use Cases, Industry Focus, and Framework Expansion.

[13] J. Chen, Z. Feng, J. Wen, B. Liu, and L. Sha. 2019. A Container-based DoS Attack-Resilient Control Framework for Real-Time UAV Systems. In *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 1222–1227. DOI:https://doi.org/10.23919/DATE.2019.8714888

[14] S. Cho, E. Hong, and S. Seo. 2020. Random Number Generator Using Sensors for Drone. *IEEE Access* 8, (2020), 30343–30354. DOI:https://doi.org/10.1109/ACCESS.2020.2972958

[15] Hongjun Choi, Wen-Chuan Lee, Yousra Aafer, Fan Fei, Zhan Tu, Xiangyu Zhang, Dongyan Xu, and Xinyan Deng. 2018. Detecting Attacks Against Robotic Vehicles: A Control Invariant Approach. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security* (CCS '18), Association for Computing Machinery, New York, NY, USA, 801–816. DOI:https://doi.org/10.1145/3243734.3243752

[16] Christopher Todd and Charles Werner. 2020. Spring 2020 Public Safety UAS Survey Results. Retrieved from https://airt.ngo/research/

[17] CISA. 2019. ICS Advisory (ICSA-19-015-01). Retrieved from https://us-cert.cisa.gov/ics/advisories/ICSA-19-015-01

[18] Andrei Costin and Aurélien Francillon. 2012. Ghost in the Air (Traffic): On insecurity of ADS-B protocol and practical attacks on ADS-B devices. *Black Hat USA* (2012), 1–12.

[19] Cybersecurity and Infrastructure Security Agency. 2020. Protecting Against the Threat of Unmanned Aircraft Systems (UAS) - An Interagency Security Committee Best Practice. Retrieved from https://www.cisa.gov/sites/default/files/publications/Protecting%20Against%20the%20Threat%20of%20Unmanned%20Aircraft%20Systems%20November%202020_508c.pdf

[20] Pritam Dash, Mehdi Karimibiuki, and Karthik Pattabiraman. 2019. Out of Control: Stealthy Attacks against Robotic Vehicles Protected by Control-Based Techniques. In *Proceedings of the 35th Annual Computer Security Applications Conference* (ACSAC '19), Association for Computing Machinery, New York, NY, USA, 660–672. DOI:https://doi.org/10.1145/3359789.3359847

[21] David Cooper, W. Polk, Andrew Regenscheid, and Murugiah Souppaya. 2011. NIST Special Publication 800-147: BIOS Protection Guidelines. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-147/final

[22] Drew Davidson, Hao Wu, Rob Jellinek, Vikas Singh, and Thomas Ristenpart. 2016. Controlling UAVs with Sensor Input Spoofing Attacks. In *10th USENIX Workshop on Offensive Technologies (WOOT 16)*, USENIX Association, Austin, TX. Retrieved from https://www.usenix.org/conference/woot16/workshop-program/presentation/davidson

[23] Nicholas DeMarinis and Rodrigo Fonseca. 2017. Toward Usable Network Traffic Policies for IoT Devices in Consumer Networks. In *Proceedings of the 2017 Workshop on Internet of Things Security and Privacy* (IoTS&P '17), Association for Computing Machinery, New York, NY, USA, 43–48. DOI:https://doi.org/10.1145/3139937.3139949

[24] M. Elnaggar and N. Bezzo. 2018. An IRL Approach for Cyber-Physical Attack Intention Prediction and Recovery. In *2018 Annual American Control Conference (ACC)*, 222–227. DOI:https://doi.org/10.23919/ACC.2018.8430922

[25] Sriharsha Etigowni, Shamina Hossain-McKenzie, Maryam Kazerooni, Katherine Davis, and Saman Zonouz. 2018. Crystal (Ball): I Look at Physics and Predict Control Flow! Just-

Ahead-Of-Time Controller Recovery. In *Proceedings of the 34th Annual Computer Security Applications Conference* (ACSAC '18), Association for Computing Machinery, New York, NY, USA, 553–565. DOI:https://doi.org/10.1145/3274694.3274724

[26] Federal Aviation Administration. 2016. Pilot's Handbook of Aeronautical Knowledge, Chapter 13: Aviation Weather Services. Retrieved from https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/phak/media/15_phak_ch13.pdf

[27] Federal Aviation Administration. 2020. National Airspace System. Retrieved from https://www.faa.gov/air_traffic/nas/

[28] Federal Aviation Administration. 2020. Ins and Outs. Retrieved from https://www.faa.gov/nextgen/equipadsb/capabilities/ins_outs/

[29] Federal Aviation Administration. 2020. Ground-Based Navigation – Instrument Landing System (ILS). Retrieved from https://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/navservices/gbng/ils/

[30] Federal Aviation Administration. 2021. Aeronautical Information Manual, Section 1: Services Available to Pilots. Retrieved from https://www.faa.gov/air_traffic/publications/atpubs/aim_html/chap4_section_1.html

[31] C. Ge, L. Zhou, G. P. Hancke, and C. Su. 2020. A Provenance-Aware Distributed Trust Model for Resilient Unmanned Aerial Vehicle Networks. *IEEE Internet of Things Journal* PP, 99 (2020), 1–1. DOI:https://doi.org/10.1109/JIOT.2020.3014947

[32] Jairo Giraldo, David Urbina, Alvaro Cardenas, Junia Valente, Mustafa Faisal, Justin Ruths, Nils Ole Tippenhauer, Henrik Sandberg, and Richard Candell. 2018. A Survey of Physics-Based Attack Detection in Cyber-Physical Systems. *ACM Comput. Surv.* 51, 4 (July 2018). DOI:https://doi.org/10.1145/3203245

[33] R. Gomes, J. Straub, A. Jones, J. Morgan, S. Tipparach, A. Sletten, K. W. Kim, D. Loegering, N. Feikema, K. Dayananda, G. Miryala, A. Gass, K. Setterstrom, J. Mischel, D. Shipman, and C. Nazzaro. 2017. An interconnected network of UAS as a system-of-systems. In *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, 1–7. DOI:https://doi.org/10.1109/DASC.2017.8102148

[34] GRA Inc. and Daniel Gettinger. 2020. Use of Drones in Public Safety.

[35] P. Guo, H. Kim, N. Virani, J. Xu, M. Zhu, and P. Liu. 2018. RoboADS: Anomaly Detection Against Sensor and Actuator Misbehaviors in Mobile Robots. In *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 574–585. DOI:https://doi.org/10.1109/DSN.2018.00065

[36] J. Habibi, A. Gupta, S. Carlsony, A. Panicker, and E. Bertino. 2015. MAVR: Code Reuse Stealthy Attacks and Mitigation on Unmanned Aerial Vehicles. In *2015 IEEE 35th International Conference on Distributed Computing Systems*, 642–652. DOI:https://doi.org/10.1109/ICDCS.2015.71

[37] D. He, S. Chan, and M. Guizani. 2017. Communication Security of Unmanned Aerial Vehicles. *IEEE Wireless Communications* 24, 4 (August 2017), 134–139. DOI:https://doi.org/10.1109/MWC.2016.1600073WC

[38] M. Hooper, Y. Tian, R. Zhou, B. Cao, A. P. Lauf, L. Watkins, W. H. Robinson, and W. Alexis. 2016. Securing commercial WiFi-based UAVs from common security attacks. In *MILCOM 2016 - 2016 IEEE Military Communications Conference*, 1213–1218. DOI:https://doi.org/10.1109/MILCOM.2016.7795496

[39] Zhenqi Huang, Yu Wang, Sayan Mitra, and Geir Dullerud. 2016. Controller Synthesis for Linear Dynamical Systems with Adversaries. In *Proceedings of the Symposium and Bootcamp on the Science of Security* (HotSos '16), Association for Computing Machinery, New York, NY, USA, 53–62. DOI:https://doi.org/10.1145/2898375.2898378

[40] Interagency Supply Chain Group (ISG) Unmanned Aircraft Systems (UAS) Coordinating Body. 2018. UAVs in Global Health: Use Case Prioritization. Retrieved from https://isghealth.org/wp-content/uploads/2020/04/UAV-use-case-prioritization_Dec2018.pdf

[41] Radoslav Ivanov, Miroslav Pajic, and Insup Lee. 2016. Attack-Resilient Sensor Fusion for Safety-Critical Cyber-Physical Systems. *ACM Trans. Embed. Comput. Syst.* 15, 1 (February 2016). DOI:https://doi.org/10.1145/2847418

[42] James H. Williams and T.L. Signore. 2011. National Airspace System Security Cyber Architecture. Retrieved from https://www.mitre.org/sites/default/files/publications/10_4169.pdf

[43] Johann-Sebastian Pleban, Ricardo Band, and Reiner Creutzburg. 2014. Hacking and securing the AR.Drone 2.0 quadcopter: investigations for improving the security of a toy. DOI:https://doi.org/10.1117/12.2044868

[44] Joint Task Force. 2020. NIST Special Publication 800-53: Security and Privacy Controls for Information Systems and Organizations. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final

[45] Joint Task Force. 2021. NIST Special Publication 800-53B: Control Baselines for Information Systems and Organizations. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-53b/final

[46] Jon Boyens, Celia Paulsen, Rama Moorthy, and Nadya Bartol. 2015. NIST Special Publication 800-161: Supply Chain Risk Management Practices for Federal Information Systems and Organizations. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-161/final

[47] Karen Scarfone, Wayne Jansen, and Miles Tracy. 2008. NIST Special Publication 800-123: Guide to General Server Security. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-123/final

[48] M. Keshavarz, A. Shamsoshoara, F. Afghah, and J. Ashdown. 2020. A Real-time Framework for Trust Monitoring in a Network of Unmanned Aerial Vehicles. In *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 677–682. DOI:https://doi.org/10.1109/INFOCOMWKSHPS50562.2020.9162761

[49] Taegyu Kim, Chung Hwan Kim, Junghwan Rhee, Fan Fei, Zhan Tu, Gregory Walkup, Xiangyu Zhang, Xinyan Deng, and Dongyan Xu. 2019. RVFuzzer: Finding Input Validation Bugs in Robotic Vehicles through Control-Guided Testing. In *28th USENIX Security Symposium (USENIX Security 19)*, USENIX Association, Santa Clara, CA, 425–442. Retrieved from https://www.usenix.org/conference/usenixsecurity19/presentation/kim

[50] Florian Kohnhäuser, Niklas Büscher, Sebastian Gabmeyer, and Stefan Katzenbeisser. 2017. SCAPI: A Scalable Attestation Protocol to Detect Software and Physical Attacks. In *Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks* (WiSec '17), Association for Computing Machinery, New York, NY, USA, 75–86. DOI:https://doi.org/10.1145/3098243.3098255

[51] Kotikalapudi Sriram and Douglas Montgomery. 2019. NIST Special Publication 800-189: Resilient Interdomain Traffic Exchange- BGP Security and DDoS Mitigation. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-189/final

[52] Adrian P. Lauf, Richard A. Peters, and William H. Robinson. 2010. A distributed intrusion detection system for resource-constrained devices in ad-hoc networks. *Ad Hoc Networks* 8, 3 (2010), 253–266. DOI:https://doi.org/10.1016/j.adhoc.2009.08.002

[53] K. Lei, Q. Zhang, J. Lou, B. Bai, and K. Xu. 2019. Securing ICN-Based UAV Ad Hoc Networks with Blockchain. *IEEE Communications Magazine* 57, 6 (June 2019), 26–32. DOI:https://doi.org/10.1109/MCOM.2019.1800722

[54] X. Li, Y. Wang, P. Vijayakumar, D. He, N. Kumar, and J. Ma. 2019. Blockchain-Based Mutual-Healing Group Key Distribution Scheme in Unmanned Aerial Vehicles Ad-Hoc Network. *IEEE Transactions on Vehicular Technology* 68, 11 (November 2019), 11309–11322. DOI:https://doi.org/10.1109/TVT.2019.2943118

[55] C. Lin, D. He, N. Kumar, K. R. Choo, A. Vinel, and X. Huang. 2018. Security and Privacy for the Internet of Drones: Challenges and Solutions. *IEEE Communications Magazine* 56, 1 (January 2018), 64–69. DOI:https://doi.org/10.1109/MCOM.2017.1700390

[56] Renju Liu and Mani Srivastava. 2017. PROTC: PROTeCting Drone's Peripherals through ARM TrustZone. In *Proceedings of the 3rd Workshop on Micro Aerial Vehicle Networks, Systems, and Applications* (DroNet '17), Association for Computing Machinery, New York, NY, USA, 1–6. DOI:https://doi.org/10.1145/3086439.3086443

[57] Y. Liu, J. Wang, H. Song, J. Li, and J. Yuan. 2019. Blockchain-based Secure Routing Strategy for Airborne Mesh Networks. In *2019 IEEE International Conference on Industrial Internet (ICII)*, 56–61. DOI:https://doi.org/10.1109/ICII.2019.00021

[58] Loïc Lesavre, Priam Varin, and Dylan Yaga. 2021. NISTIR 8301: Blockchain Networks - Token Design and Management Overview. Retrieved from https://csrc.nist.gov/publications/detail/nistir/8301/final

[59] Sixing Lu and Roman Lysecky. 2019. Data-Driven Anomaly Detection with Timing Features for Embedded Systems. *ACM Trans. Des. Autom. Electron. Syst.* 24, 3 (April 2019). DOI:https://doi.org/10.1145/3279949

[60] Aaron Luo. 2017. Drones Hijacking: Multi-Dimensional Attack Vectors and Countermeasures. In *DEFCON 24*. Retrieved from https://infocon.org/cons/DEF%20CON/DEF%20CON%2024/DEF%20CON%2024%20presentations/DEF%20CON%2024%20-%20Aaron-Luo-Drones-Hijacking-Multi-Dimensional-Attack-Vectors-And-Countermeasures-UPDATED.pdf

[61] M. R. Manesh, J. Kenney, W. C. Hu, V. K. Devabhaktuni, and N. Kaabouch. 2019. Detection of GPS Spoofing Attacks on Unmanned Aerial Systems. In *2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, 1–6. DOI:https://doi.org/10.1109/CCNC.2019.8651804

[62] M. R. Manesh, M. Mullins, K. Foerster, and N. Kaabouch. 2018. A preliminary effort toward investigating the impacts of ADS-B message injection attack. In *2018 IEEE Aerospace Conference*, 1–6. DOI:https://doi.org/10.1109/AERO.2018.8396610

[63] Mohsen Riahi Manesh and Naima Kaabouch. 2017. Analysis of vulnerabilities, attacks, countermeasures and overall risk of the Automatic Dependent Surveillance-Broadcast (ADS-B) system. *International Journal of Critical Infrastructure Protection* 19, (2017), 16–31. DOI:https://doi.org/10.1016/j.ijcip.2017.10.002

[64] Askelson Mark. 2020. UAS Test Data Collection and Analysis: Phase I Final Report.

[65] Mark Askelson, Henry Cathey, Ronald Marsh, Zachary P. Waller, Paul Snyder, Gary M. Ullrich, Chris Theisen, Naima Kaabouch, William Semke, Michael Mullins, Kyle Foerster, Rosa Brothman, Stephen B. Hottman, Kerry Williamson, Eric Johnson, Alexander VanHoudt,

and Zachariah LaRue. 2017. Small UAS Detect and Avoid Requirements Necessary for Limited Beyond Visual Line of Sight (BVLOS) Operations. University of North Dakota Scholarly Commons. Aviation Faculty Publications. *UND Commons* (2017). Retrieved from https://commons.und.edu/avi-fac/5/

[66] J. Maxa, M. S. B. Mahmoud, and N. Larrieu. 2016. Extended verification of secure UAANET routing protocol. In *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*, 1–16. DOI:https://doi.org/10.1109/DASC.2016.7777970

[67] J. Mead, C. Bobda, and T. J. Whitaker. 2016. Defeating drone jamming with hardware sandboxing. In *2016 IEEE Asian Hardware-Oriented Security and Trust (AsianHOST)*, 1–6. DOI:https://doi.org/10.1109/AsianHOST.2016.7835557

[68] Michael Bartock, Suzanne Lightman, Ya-Shian Li-Baboud, James McCarthy, Karen Reczek, Joseph Brule, Doug Northrip, Arthur Scholz, and Theresa Suloway. 2021. NISTIR 8323: Foundational PNT Profile: Applying the Cybersecurity Framework for the Responsible Use of Positioning, Navigation, and Timing (PNT) Services. Retrieved from https://csrc.nist.gov/publications/detail/nistir/8323/final

[69] Miles Tracy, Wayne Jansen, Karen Scarfone, and Theodore Winograd. 2007. NIST Special Publication 800-44: Guidelines on Securing Public Web Servers. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-44/version-2/final

[70] Minnesota Department of Transportation. 2019. Navigation Technologies. Retrieved from https://www.dot.state.mn.us/aero/planning/sasp/documents/MnSASP%20Navigation%20Technologies%20Trend%20Paper.pdf

[71] T. Miquel, J. Condomines, R. Chemali, and N. Larrieu. 2017. Design of a robust controller/observer for TCP/AQM network: First application to intrusion detection systems for drone fleet. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1707–1712. DOI:https://doi.org/10.1109/IROS.2017.8205982

[72] R. Mitchell and I. Chen. 2014. Adaptive Intrusion Detection of Malicious Unmanned Air Vehicles Using Behavior Rule Specifications. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 44, 5 (May 2014), 593–604. DOI:https://doi.org/10.1109/TSMC.2013.2265083

[73] Robert Mitchell and Ing-Ray Chen. A survey of intrusion detection techniques for cyber-physical systems. *ACM Computing Surveys (CSUR), Volume 46, Issue 4*. DOI:https://doi.org/10.1145/2542049

[74] Daniel Moser, Vincent Lenders, and Srdjan Capkun. 2019. Digital Radio Signal Cancellation Attacks: An Experimental Evaluation. In *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks* (WiSec '19), Association for Computing Machinery, New York, NY, USA, 23–33. DOI:https://doi.org/10.1145/3317549.3319720

[75] Nour Moustafa and Alireza Jolfaei. 2020. Autonomous Detection of Malicious Events Using Machine Learning Models in Drone Networks. In *Proceedings of the 2nd ACM MobiCom Workshop on Drone Assisted Wireless Communications for 5G and Beyond* (DroneCom '20), Association for Computing Machinery, New York, NY, USA, 61–66. DOI:https://doi.org/10.1145/3414045.3415951

[76] Murugiah Souppaya and Karen Scarfone. 2013. NIST Special Publication 800-83: Guide to Malware Incident Prevention and Handling for Desktops and Laptops. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-83/rev-1/final

[77] Murugiah Souppaya, Karen Scarfone, and Donna Dodson. 2021. NIST Special Publication 800-218: Secure Software Development Framework (SSDF) -Recommendations for Mitigating the Risk of Software Vulnerabilities. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-218/draft

[78] Shoei Nashimoto, Daisuke Suzuki, Takeshi Sugawara, and Kazuo Sakiyama. 2018. Sensor CON-Fusion: Defeating Kalman Filter in Signal Injection Attack. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security* (ASIACCS '18), Association for Computing Machinery, New York, NY, USA, 511–524. DOI:https://doi.org/10.1145/3196494.3196506

[79] B. Nassi, R. Ben-Netanel, A. Shamir, and Y. Elovici. 2019. Drones' Cryptanalysis - Smashing Cryptography with a Flicker. In *2019 IEEE Symposium on Security and Privacy (SP)*, 1397–1414. DOI:https://doi.org/10.1109/SP.2019.00051

[80] B. Nassi, R. Bitton, R. Masuoka, A. Shabtai, and Y. Elovici. 2021. SoK: Security and Privacy in the Age of Commercial Drones. In *2021 2021 IEEE Symposium on Security and Privacy (SP)*, IEEE Computer Society, Los Alamitos, CA, USA, 1434–1451. DOI:https://doi.org/10.1109/SP40001.2021.00005

[81] NIST. 2017. NVD Advisory (CVE-2017-12819). Retrieved from https://nvd.nist.gov/vuln/detail/CVE-2017-12819

[82] Juhwan Noh, Yujin Kwon, Yunmok Son, Hocheol Shin, Dohyun Kim, Jaeyeong Choi, and Yongdae Kim. 2019. Tractor Beam: Safe-hijacking of Consumer Drones with Adaptive GPS Spoofing. *ACM Trans. Priv. Secur.* 22, 2 (April 2019), 12:1-12:26. DOI:https://doi.org/10.1145/3309735

[83] Paul E. Black, Vadim Okun, and Barbara Guttman. 2021. NISTIR 8397: Guidelines on minimum standards for developer verification of software. Retrieved from https://www.nist.gov/publications/guidelines-minimum-standards-developer-verification-software

[84] Paul Grassi, Elaine Newton, James Fenton, Ray Perlner, Andrew Regenscheid, William Burr, Justin Richer, Naomi Lefkovitz, Jamie Danker, Yee-Yin Choong, Kristen Greene, and Mary Theofanos. 2020. NIST Special Publication 800-63B: Digital Identity Guidelines. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-63b/final

[85] Paul Grassi, James Fenton, Naomi Lefkovitz, Jamie Danker, Yee-Yin Choong, Kristen Greene, and Mary Theofanos. 2020. NIST Special Publication 800-63A: Digital Identity Guidelines Enrollment and Identity Proofing Requirements. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-63a/final

[86] Paul Grassi, Michael Garcia, and James Fenton. 2020. NIST Special Publication 800-63-3: Digital Identity Guidelines. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-63/3/final

[87] Josh Pozner. 2020. A Comprehensive List of Commercial Drone Use Cases (128+ And Growing). Retrieved from https://www.dronegenuity.com/commercial-drone-use-cases-comprehensive-list/

[88] Shahed E. Quadir, Junlin Chen, Domenic Forte, Navid Asadizanjani, Sina Shahbazmohamadi, Lei Wang, John Chandy, and Mark Tehranipoor. 2016. A Survey on Chip to System Reverse Engineering. *J. Emerg. Technol. Comput. Syst.* 13, 1 (April 2016). DOI:https://doi.org/10.1145/2755563

[89] Andrew Regenscheid. 2018. NIST Special Publication 800-193: Platform Firmware Resiliency Guidelines. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-193/final

[90] Musa G. Samaila, João B. F. Sequeiros, Mário M. Freire, and Pedro R. M. Inácio. 2018. Security Threats and Possible Countermeasures in IoT Applications Covering Different Industry Domains. In *Proceedings of the 13th International Conference on Availability, Reliability and Security* (ARES 2018), Association for Computing Machinery, New York, NY, USA. DOI:https://doi.org/10.1145/3230833.3232800

[91] Fred Samland, Jana Fruth, Mario Hildebrandt, Tobias Hoppe, and Jana Dittmann. 2012. AR.Drone: security threat analysis and exemplary attack to track persons. In *Intelligent Robots and Computer Vision XXIX: Algorithms and Techniques*, SPIE, 158–172. DOI:https://doi.org/10.1117/12.902990

[92] B. Satchidanandan and P. R. Kumar. 2017. Dynamic Watermarking: Active Defense of Networked Cyber–Physical Systems. *Proceedings of the IEEE* 105, 2 (February 2017), 219–240. DOI:https://doi.org/10.1109/JPROC.2016.2575064

[93] V. Sharma, I. You, K. Yim, I. Chen, and J. Cho. 2019. BRIoT: Behavior Rule Specification-Based Misbehavior Detection for IoT-Embedded Cyber-Physical Systems. *IEEE Access* 7, (2019), 118556–118580. DOI:https://doi.org/10.1109/ACCESS.2019.2917135

[94] Abdulhadi Shoufan, Hassan AlNoon, and Joonsang Baek. 2015. Secure communication in civil drones. In *International Conference on Information Systems Security and Privacy*, Springer, 177–195. DOI:https://doi.org/10.1007/978-3-319-27668-7_11

[95] Yunmok Son, Hocheol Shin, Dongkwan Kim, Youngseok Park, Juhwan Noh, Kibum Choi, Jungwoo Choi, and Yongdae Kim. 2015. Rocking Drones with Intentional Sound Noise on Gyroscopic Sensors. In *24th USENIX Security Symposium (USENIX Security 15)*, USENIX Association, Washington, D.C., 881–896. Retrieved from https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/son

[96] C. Stracquodaine, A. Dolgikh, M. Davis, and V. Skormin. 2016. Unmanned Aerial System security using real-time autopilot software analysis. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, 830–839. DOI:https://doi.org/10.1109/ICUAS.2016.7502633

[97] M. Strohmeier, V. Lenders, and I. Martinovic. 2015. On the Security of the Automatic Dependent Surveillance-Broadcast Protocol. *IEEE Communications Surveys Tutorials* 17, 2 (2015), 1066–1087. DOI:https://doi.org/10.1109/COMST.2014.2365951

[98] P. Sun, L. Garcia, and S. Zonouz. 2019. Tell Me More Than Just Assembly! Reversing Cyber-Physical Execution Semantics of Embedded IoT Controller Software Binaries. In *2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 349–361. DOI:https://doi.org/10.1109/DSN.2019.00045

[99] M. Tanveer, A. H. Zahid, M. Ahmad, A. Baz, and H. Alhakami. 2020. LAKE-IoD: Lightweight Authenticated Key Exchange Protocol for the Internet of Drone Environment. *IEEE Access* 8, (2020), 155645–155659. DOI:https://doi.org/10.1109/ACCESS.2020.3019367

[100] The MITRE Corporation. 2006. CWE - Common Weakness Enumeration. Retrieved from https://cwe.mitre.org/

[101] The White House. 2018. National Strategy for Aviation Security of the United States of America. Retrieved from

https://www.globalsecurity.org/security/library/policy/national/national-strategy-for-aviation-security_2019.pdf

[102] K. Thiha, B. Soong, V. Vaiyapuri, and S. Nadarajan. 2019. A new method of defeating reactive jamming: Hardware Design Approach. In *2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 184–188. DOI:https://doi.org/10.1109/ICIEA.2019.8834161

[103] Andrew Thompson. 2015. 20 Commercial Drone Use Cases and Leading Innovators. Retrieved from https://blog.ventureradar.com/2015/12/29/20-commercial-drone-use-cases-and-leading-innovators/

[104] T. Trippel, O. Weisse, W. Xu, P. Honeyman, and K. Fu. 2017. WALNUT: Waging Doubt on the Integrity of MEMS Accelerometers with Acoustic Injection Attacks. In *2017 IEEE European Symposium on Security and Privacy (EuroS&P)*, 3–18. DOI:https://doi.org/10.1109/EuroSP.2017.42

[105] Vincent Hu, David Ferraiolo, and Richard Kuhn. 2006. NIST Interagency report 7316: Assessment of Access Control System. Retrieved from https://csrc.nist.gov/publications/detail/nistir/7316/final

[106] T. P. Vuong, G. Loukas, and D. Gan. 2015. Performance Evaluation of Cyber-Physical Intrusion Detection on a Robotic Vehicle. In *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, 2106–2113. DOI:https://doi.org/10.1109/CIT/IUCC/DASC/PICOM.2015.313

[107] M. Wazid, A. K. Das, N. Kumar, A. V. Vasilakos, and J. J. P. C. Rodrigues. 2019. Design and Analysis of Secure Lightweight Remote User Authentication and Key Agreement Scheme in Internet of Drones Deployment. *IEEE Internet of Things Journal* 6, 2 (April 2019), 3572–3584. DOI:https://doi.org/10.1109/JIOT.2018.2888821

[108] Wikipedia. 2021. Flight Dispatcher. Retrieved from https://en.wikipedia.org/wiki/Flight_dispatcher

[109] Zhiheng Xu and Quanyan Zhu. 2015. Secure and Resilient Control Design for Cloud Enabled Networked Control Systems. In *Proceedings of the First ACM Workshop on Cyber-Physical Systems-Security and/or PrivaCy* (CPS-SPC '15), Association for Computing Machinery, New York, NY, USA, 31–42. DOI:https://doi.org/10.1145/2808705.2808708

[110] X. Ying, J. Mazer, G. Bernieri, M. Conti, L. Bushnell, and R. Poovendran. 2019. Detecting ADS-B Spoofing Attacks Using Deep Neural Networks. In *2019 IEEE Conference on Communications and Network Security (CNS)*, 187–195. DOI:https://doi.org/10.1109/CNS.2019.8802732

[111] M. Yoon, B. Liu, N. Hovakimyan, and L. Sha. 2017. VirtualDrone: Virtual Sensing, Actuation, and Communication for Attack-Resilient Unmanned Aerial Systems. In *2017 ACM/IEEE 8th International Conference on Cyber-Physical Systems (ICCPS)*, 143–154.

[112] R. Zhang, J. Condomines, N. Larrieu, and R. Chemali. 2018. Design of a novel network intrusion detection system for drone communications. In *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*, 1–10. DOI:https://doi.org/10.1109/DASC.2018.8569300

APPENDIX A: CYBERSECURITY USE CASE GROUPINGS

Table 27. Subgroup #1 of the Autonomous/BVLOS/Swarm Cybersecurity Use Case.

Industry	Use Case	Autonomy	Operation Range	Collaboration	Customization	Life-critical
Hospitality and Tourism	Visual Marketing	Autonomous	BVLOS	Swarm	Low	No
Humanitarian and Disaster Relief	Restoration of Vital Services (Power, Phone, Wifi)	Autonomous	BVLOS	Swarm	High	Yes
Humanitarian and Disaster Relief	Monitor and Combat Natural Disasters (Forest Fires)	Autonomous	BVLOS	Swarm	High	Yes
Disease Control	Pest Control / Collection	Autonomous	BVLOS	Swarm	High	No
Disease Control	Pollution Monitoring and Control	Autonomous	BVLOS	Swarm	High	No
Retail	Product Organization, Storage, and Inventory	Autonomous	BVLOS	Swarm	High	No
Advertising / Visual / News	Advertising	Autonomous	BVLOS	Swarm	High	No
Sports and Entertainment	Drone Racing	Autonomous	BVLOS	Swarm	High	No
Agriculture	Pest Detection and Control	Autonomous	BVLOS	Swarm	High	No
Agriculture	Warning and Remedy of Crop Failure	Autonomous	BVLOS	Swarm	High	No
Agriculture	Perform Manual Redundant	Autonomous	BVLOS	Swarm	High	No

	Tasks (i.e. seeding, planting, and spraying)					
Weather Forecasting	Gather Data in Inhospitable or Extreme Locations (i.e. ocean depths, high atmosphere)	Autonomous	BVLOS	Swarm	High	No
Shipping	Navigational Aids	Autonomous	BVLOS	Swarm	High	Yes
Mining and Resource Exploration	Mining Operations	Autonomous	BVLOS	Swarm	High	No
Urban Planning	Traffic Direction	Autonomous	BVLOS	Swarm	Low	Yes
Telecommunications / Entertainment	Connectivity	Autonomous	BVLOS	Swarm	High	Yes
Airlines and Airports	Flight / Navigation System Testing and Verification.	Autonomous	BVLOS	Swarm	High	No
Others	Repair Drones	Autonomous	BVLOS	Swarm	High	No
Others	Firefighting	Autonomous	BVLOS	Swarm	High	Yes

Table 28. Subgroup #2 of the Autonomous/BVLOS/Swarm Cybersecurity Use Case.

Industry	**Use Case**	**Autonomy**	**Operation Range**	**Collaboration**	**Customization**	**Life-critical**
Emergency Response	Search and Rescue (Infrared and Visuals)	Autonomous	BVLOS	Swarm	Low	Yes

Emergency Response	Inspect and Explore Disaster Areas (Indoor, Outdoor, and confined spaces)	Autonomous	BVLOS	Swarm	Low	Yes
Humanitarian and Disaster Relief	Damage and Infrastructure Assessment	Autonomous	BVLOS	Swarm	Low	No
Humanitarian and Disaster Relief	Predict and Access Natural Disasters and Effected Areas	Autonomous	BVLOS	Swarm	Low	Yes
Humanitarian and Disaster Relief	Create 3D Models of the aftermath	Autonomous	BVLOS	Swarm	Low	No
Disease Control	Disease Tracking and Monitoring	Autonomous	BVLOS	Swarm	Low	No
Advertising / Visual / News	News Coverage	Autonomous	BVLOS	Swarm	Low	No
Agriculture	Predict and Analyze Crop Growth	Autonomous	BVLOS	Swarm	Low	No
Agriculture	Provide Aerial Views	Autonomous	BVLOS	Swarm	Low	No
Weather Forecasting	Follow Weather Patterns	Autonomous	BVLOS	Swarm	Low	No
Weather Forecasting	Explore, Document, and Predict Severe Weather	Autonomous	BVLOS	Swarm	High	Yes
Weather Forecasting	Severe Weather Warnings	Autonomous	BVLOS	Swarm	Low	Yes

Conservation	Monitor and Track Animals	Autonomous	BVLOS	Swarm	Low	No
Conservation	Combat poachers	Autonomous	BVLOS	Swarm	Low	No
Conservation	Research Ecosystems	Autonomous	BVLOS	Swarm	Low	No
Insurance	Natural Disaster Monitoring and Modeling	Autonomous	BVLOS	Swarm	High	Yes
Mining and Resource Exploration	Surveying and Mapping	Autonomous	BVLOS	Swarm	Low	No
Mining and Resource Exploration	Security	Autonomous	BVLOS	Swarm	Low	Yes
Urban Planning	City Centers Redesign	Autonomous	BVLOS	Swarm	Low	No
Manufacturing and Inventory Management	Inventory Location	Autonomous	BVLOS	Swarm	High	No
Manufacturing and Inventory Management	Inventory Measurement	Autonomous	BVLOS	Swarm	High	No
Manufacturing and Inventory Management	Order Compilation and Inspection	Autonomous	BVLOS	Swarm	High	No
Others	Infrared Thermography	Autonomous	BVLOS	Swarm	High	No

Table 29. Subgroup #3 of the Autonomous/BVLOS/Swarm Cybersecurity Use Case.

Industry	**Use Case**	**Autonomy**	**Operation Range**	**Collaboration**	**Customization**	**Life-critical**

Humanitarian and Disaster Relief	Distribute Food and Water	Autonomous	BVLOS	Swarm	Low	Yes
Retail	Product Delivery	Autonomous	BVLOS	Swarm	High	No
Advertising / Visual / News	Promotional Item Delivery	Autonomous	BVLOS	Swarm	High	No
Shipping	Autonomous Shipping	Autonomous	BVLOS	Swarm	High	No
Others	Carry equipment	Autonomous	BVLOS	Swarm	High	No

Table 30. Subgroup #1 of the Autonomous/BVLOS/Single Cybersecurity Use Case.

Industry	**Use Case**	**Autonomy**	**Operation Range**	**Collaboration**	**Customization**	**Life-critical**
Food / Restaurant Industry	Drone Waiter	Autonomous	BVLOS	Single	Low	No
Hospitality and Tourism	Food and Beverage Preparation	Autonomous	BVLOS	Single	Low	No
Hospitality and Tourism	Entertainment / Activity	Autonomous	BVLOS	Single	High	No
Hospitality and Tourism	Property Maintenance	Autonomous	BVLOS	Single	Low	Yes
Construction	Soil Analysis	Autonomous	BVLOS	Single	High	No
Energy	Operate in Contaminated or Hazardous Areas	Autonomous	BVLOS	Single	High	No
Energy	Leakages and Spread Detection	Autonomous	BVLOS	Single	High	Yes
Energy	Energy Exploration	Autonomous	BVLOS	Single	Low	No
Mining and Resource Exploration	Inventory Management	Autonomous	BVLOS	Single	Low	No

Others	Machine learning service	Autonomous	BVLOS	Single	Low	No

Table 31. Subgroup #2 of the Autonomous/BVLOS/Single Cybersecurity Use Case.

Industry	Use Case	Autonomy	Operation Range	Collaboration	Customization	Life-critical
Hospitality and Tourism	Security	Autonomous	BVLOS	Single	Low	Yes
Hospitality and Tourism	Life-guarding	Autonomous	BVLOS	Single	High	Yes
Advertising / Visual / News	Cinematography	Autonomous	BVLOS	Single	High	No
Advertising / Visual / News	Videography	Autonomous	BVLOS	Single	High	No
Advertising / Visual / News	Photography	Autonomous	BVLOS	Single	High	No
Construction	Topographic Mapping and Analysis	Autonomous	BVLOS	Single	Low	No
Construction	Surveying and Digital Mapping	Autonomous	BVLOS	Single	Low	No
Construction	Inspections	Autonomous	BVLOS	Single	Low	Yes
Insurance	Inspection Of Claims	Autonomous	BVLOS	Single	Low	Yes
Insurance	Fraud Detection / Prevention	Autonomous	BVLOS	Single	Low	Yes
Energy	Infrastructure and Compliance Inspection	Autonomous	BVLOS	Single	Low	No
Mining and Resource Exploration	Safety Inspections	Autonomous	BVLOS	Single	Low	Yes

Urban Planning	Traffic and Population Studies	Autonomous	BVLOS	Single	Low	No
Urban Planning	Terrain, Weather, Water, and Resource change Mapping	Autonomous	BVLOS	Single	Low	No
Telecommunications / Entertainment	Infrastructure and Compliance Inspection	Autonomous	BVLOS	Single	Low	No
Airlines and Airports	Airport Air Security	Autonomous	BVLOS	Single	Low	Yes

Table 32. Subgroup #3 of the Autonomous/BVLOS/Single Cybersecurity Use Case.

Industry	**Use Case**	**Autonomy**	**Operation Range**	**Collaboration**	**Customization**	**Life-critical**
Food / Restaurant Industry	Food Delivery	Autonomous	BVLOS	Single	Low	No
Food / Restaurant Industry	Convenience Store / Grocery Delivery	Autonomous	BVLOS	Single	Low	No
Food / Restaurant Industry	Food and Beverage Service (i.e at pools, on golf courses)	Autonomous	BVLOS	Single	Low	No
Hospitality and Tourism	Transportation of Materials	Autonomous	BVLOS	Single	Low	No
Healthcare	Medication / Prescription Delivery	Autonomous	BVLOS	Single	High	Yes
Healthcare	Blood Donation Delivery	Autonomous	BVLOS	Single	High	Yes
Healthcare	Laboratory Sample	Autonomous	BVLOS	Single	High	Yes

	Collection and Delivery					
Healthcare	Vaccine Storage and Delivery	Autonomous	BVLOS	Single	High	Yes
Healthcare	Organ Transport	Autonomous	BVLOS	Single	High	Yes
Emergency Response	Equipment Transport	Autonomous	BVLOS	Single	Low	No
Manufacturing and Inventory Management	Equipment Transport	Autonomous	BVLOS	Single	Low	No
Others	Home Delivery (i.e. Dry Cleaned Laundry Delivery)	Autonomous	BVLOS	Single	High	No

Table 33. Use Cases Under the Autonomous/VLOS/Swarm Cybersecurity Use Case.

Industry	**Usecase**	**Autonomy**	**Operation Range**	**Collaboration**	**Customization**	**Life-critical**
Sports and Entertainment	Synchronized Light Shows	Autonomous	VLOS	Swarm	Low	No
Sports and Entertainment	Floating Projection Screens	Autonomous	VLOS	Swarm	High	No
Sports and Entertainment	Drone Puppeteers	Autonomous	VLOS	Swarm	High	No
Airlines and Airports	Pest Control	Autonomous	VLOS	Swarm	High	No
Manufacturing and Inventory Management	Manufacturing	Autonomous	VLOS	Swarm	High	No
Others	Low or High Pressure	Autonomous	VLOS	Swarm	High	No

	Cleaning Solutions					
Others	Use a spotlight	Autonomous	VLOS	Swarm	High	No

Table 34. Use Cases Under the Autonomous/VLOS/Single Cybersecurity Use Case.

Industry	Usecase	Autonomy	Operation Range	Collaboration	Customization	Life-critical
Construction	Monitor Building Progress	Autonomous	VLOS	Single	Low	No
Construction	3D Renderings	Autonomous	VLOS	Single	Low	No
Real Estate	Property Tours	Autonomous	VLOS	Single	High	No
Energy	Buildings and Transmission Efficiency Mapping	Autonomous	VLOS	Single	Low	No
Telecommunications / Entertainment	Radio Planning and Line-of-Sight Mapping	Autonomous	VLOS	Single	Low	No
Airlines and Airports	Infrastructure and Airplane Inspections	Autonomous	VLOS	Single	Low	Yes
Others	3D Renderings	Autonomous	VLOS	Single	Low	No
Others	Fitness	Autonomous	VLOS	Single	Low	Yes
Others	Video Games	Autonomous	VLOS	Single	Low	No
Others	Security	Autonomous	VLOS	Single	Low	Yes
Others	Spray Paint	Autonomous	VLOS	Single	High	No
Others	Ultrasonic Testing (UT)	Autonomous	VLOS	Single	High	No
Others	Dry Film Thickness (DFT)	Autonomous	VLOS	Single	High	No

Table 35. Use Cases Under the Manual/BVLOS/Swarm Cybersecurity Use Case.

Industry	Usecase	Autonomy	Operation Range	Collaboration	Customization	Life-critical
Conservation	Collect Samples	Manual	BVLOS	Swarm	High	No
Shipping	Detect Emission Infractions and Identify Offenders	Manual	BVLOS	Swarm	High	Yes
Shipping	Search and Rescue	Manual	BVLOS	Swarm	Low	Yes
Insurance	Drone Insurance	Manual	BVLOS	Swarm	High	No

Table 36. Use Cases Under the Manual/BVLOS/Single Cybersecurity Use Case.

Industry	Usecase	Autonomy	Operation Range	Collaboration	Customization	Life-critical
Shipping	Safety and Compliance Inspections	Manual	BVLOS	Single	Low	Yes
Real Estate	3D Renderings	Manual	BVLOS	Single	High	No
Real Estate	Infrared Analysis	Manual	BVLOS	Single	High	No
Mining and Resource Exploration	Exploration	Manual	BVLOS	Single	High	No
Airlines and Airports	Search & Rescue	Manual	BVLOS	Single	High	Yes

Table 37. Use Cases Under the Manual/VLOS/Swarm Cybersecurity Use Case.

Industry	Usecase	Autonomy	Operation Range	Collaboration	Customization	Life-critical
Sports and Entertainment	Drone Combat	Manual	VLOS	Swarm	High	No

Sports and Entertainment	Broadcasting Sports	Manual	VLOS	Swarm	Low	No
Sports and Entertainment	Instant Replay / Officiating Assistance	Manual	VLOS	Swarm	Low	No
Manufacturing and Inventory Management	Raw Materials Discovery	Manual	VLOS	Swarm	Low	No
Others	Film: Wedding, Fireworks, Concerts, Parties, etc.	Manual	VLOS	Swarm	Low	No

Table 38. Use Cases Under the Manual/VLOS/Single Cybersecurity Use Case.

Industry	**Usecase**	**Autonomy**	**Operation Range**	**Collaboration**	**Customization**	**Life-critical**
Construction	Physical Construction	Manual	VLOS	Single	High	No
Real Estate	Photography and Videography (Exterior and Interior)	Manual	VLOS	Single	Low	No
Real Estate	Showcase and Suggestion of Amenities, Additions, or Additional Structures	Manual	VLOS	Single	High	No
Manufacturing and Inventory Management	Assembly Lines Inspection	Manual	VLOS	Single	Low	Yes
Others	Fishing	Manual	VLOS	Single	High	No