# Non-Repudiation for Drone-Related Data

ICAO Drone Enable 2022

Joseph Rios
Jaewoo Jung
Marcus Johnson
*NASA Ames Research Center*

## Abstract

Concepts for the management of Uncrewed Aircraft Systems (UAS) at scale rely on the exchange of data amongst multiple stakeholders. Even as these concepts vary across nations and industry, the movement of data between different entities is a common theme. While there is universal agreement on necessity of appropriate cybersecurity measures to address data communication, there has been minimal focus on the feasibility of implementing non-repudiation solutions for UAS systems. This means that data exchanged in support of UAS operations are open to "attack" via parties that may deny sending or receiving certain data, which can weaken the effectiveness and acceptability of these systems. This paper highlights the current and future need for non-repudiation, supported by references to multiple international organizations, and an approach to implementing non-repudiation leveraging open standards.

## Intro

There are many aspects to creating a secure system. CANSO defined seven top-level security requirements for data and information [1]; non-repudiation is one of those seven requirements. Amongst the other six in that list (confidentiality, integrity, availability, authentication, authorization, and traceability), it can be argued that non-repudiation has received the least attention, especially in airspace management of UAS.

Non-repudiation has several, similar definitions. Many documents reference National Institute of Standards and Technology (NIST) SP 800-53 Rev. 5 [2] which defines non-repudiation as:

> *Protection against an individual who falsely denies having performed a certain action and provides the capability to determine whether an individual took a certain action, such as creating information, sending a message, approving information, or receiving a message.*

Non-repudiation makes it difficult for one to successfully claim that it did not send/receive a message when it, in fact, did send/receive that message.

Non-repudiation takes on heightened importance in any federated system wherein multiple entities are exchanging data directly amongst themselves. UAS Traffic Management (UTM) concepts involve the exchange of data between operators, often without a State-provided intermediary. For ground-based systems, these data exchanges often leverage Hypertext Transfer Protocol (HTTP) with a Representational State Transfer (REST) approach, which is the dominate approach today. Non-repudiation protections may look much different in other communications approaches, and some of those approaches, like those leveraging blockchains, may have non-repudiation "built-in."

## Non-Repudiation in UTM

A key example of RESTful data exchanges is the ASTM F3548-21 standard ("Standard Specification for UAS Traffic Management (UTM) UAS Service Supplier (USS) Interoperability"). NASA and others are working on a related specification for larger, passenger-carrying UAS that leverages the ASTM work. The Federal Aviation Administration's (FAA's) Low Altitude Authorization and Notification Capability (LAANC) also relies on HTTP and REST for communications. The Linux Foundation's Discovery and Synchronization Service (DSS) underpins several UTM services such as strategic conflict detection and remote identification; DSS works via a REST API. The Flight Information Management System (FIMS) as originally described by NASA has versions developed or tested by the FAA, SESAR, and Australia Air Services. FIMS has relied on HTTP and REST. The Joint Authorities for Rulemaking on Unmanned Systems (JARUS) published a cyber annex to the Specific Operations Risk Assessment (SORA) that calls out non-repudiation as an important attribute, despite the focus on safety of specific operations rather than the management of air traffic.

## Previous Work

Non-repudiation in the UTM domain has been studied by NASA and the FAA. In NASA's original specification for USSs [3], non-repudiation was considered by requiring operation plans to be digitally signed by the Remote Pilot in Command (RPIC) as well as the vehicle:

> *The signing of an operation plan by a vehicle and an RPIC provides assurance that the resources noted within the operation plan are indeed the resources to be used in execution of the plan. This is a non-repudiation and data integrity step. RPICs will have confidence that plans are not altered after they have signed/agreed to serve as RPIC. UAS operators and USSs will have confidence that a RPIC will not be able to claim they were not part of the operation. Similar arguments can be made for the vehicle: all stakeholders will have confidence regarding the exact vehicle performing an operation.*

That document also noted at the time that the exact method for digitally signing messages was yet to be determined.

NASA followed up with a Technical Memorandum describing initial authentication and authorization requirements [4] where message signing was more fully defined. The approach was based on existing standards for JSON Web Signatures (JWS) and associated Internet Engineering Task Force (IETF) standards. Message signing had an early draft from IETF at that time, but was internally assessed to not be mature enough for implementation. The approach recommended involved creating a JWS from the body of the HTTP message, but "detaching" the body of the JWS to reduce the size of the HTTP header that would contain the JWS. The need to manage the size of that header was discovered via early NASA testing with industry in the UTM Project [5]. Message signing was part of NASA flight testing during the TCL4 demonstration [6]. The FAA continued such testing, including within UTM Pilot Program 2.0 (UPP2) [7]. A product of that event was a detailed security analysis of USS communications [8]. That report touches on the value of non-repudiation and dives into the Public Key Infrastructure (PKI) required to support it.

### Key Lessons Learned

- Standards for message signing are currently maturing, but provide the best known approaches for HTTP exchanges
- Attempts to develop a UTM-specific approach to message signing are frought with implementation pitfalls and open the need for extensive security analysis
- Consider a holistic approach for all components of the UTM ecosystem, rather than trying to solve on a connection-by-connection basis

## Problem Statement

Non-repudiation is a desired characteristic of many data exchanges. In currently proposed architectures and approaches related to UTM, there has been little work on the issue of non-repudiation. Repudiation attacks are often difficult to reason about since they do not protect day-to-day communications in the same way as encryption or authentication. Additionally, repudiation is not typically associated with flight safety. Repudiation attacks more likely occur after some adverse, domain-specific event. Repudiation may be used as a deflection of responsibility.

Beyond the repercussions to particular operators or service providers in the case of a repudiation attack, the system as a whole may suffer due to loss of confidence when such attacks occur. The acceptance and success of UTM and UAS operations at scale may rely, in part, on effective non-repudiation.

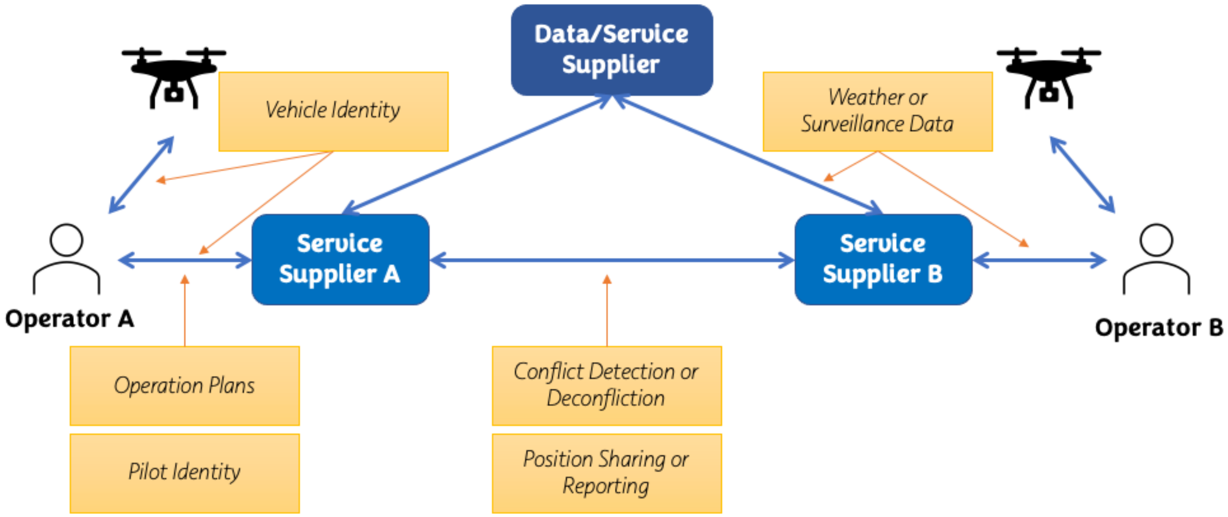

*Figure 1. Entities and connections with a subset of the data that require repudiation protection.*

### Attack Use Case 1: Operation plan denial

In the ASTM standard for USS intent sharing, operators share their operational plans with each other via a USS. If two operations were to collide and then strike structures on the ground, there could be significant liability and regulatory impact to the operators, opening the opportunity or incentive for the repudiation of data exchanges under this standard.

### Attack Use Case 2: Position reports

A USS may claim that positions were provided upon request when they were actually not provided or that positions logged by one USS were not the positions that were actually sent. Again, in these cases doubt is sown within the system as a whole in adverse circumstances.

### Attack Use Case 3: Pilot or Vehicle Repudiation

A pilot may refute that he or she was the pilot-in-command in the absence of non-repudiation, which can be especially problematic in a post-collision investigation. Additionally, the vehicle that was associated with an operation via an operation plan may be repudiated if that vehicle was not part of the signature chain.

### Attack Use Case 4: Supplementary Data Service Provider

In future federated systems for air traffic management, there may be an operator dependence upon other parties for certain data or services. There may be legal or contractual ramifications related to data exchanges between the operator and SDSP. In the event of a crash, the pilot may claim to have not received the data or may alter the data afterward.

## Proposed Solution

This document focuses on providing an approach to non-repudiation for HTTP REST communications. This paper provides the following design considerations for choosing a solution for non-repudiation of HTTP communications within UTM (and related) systems:

1. Provide a mechanism for both requests and responses.
2. Leverage the same approach for protection of both requests and responses.
3. Use standards whenever possible.
4. Ensure approaches do not overly impact other aspects of communications.

Message signing is a strong approach to providing non-repudiation. By using accepted approaches to key management, the owner of a private key can sign messages and the receiver or another party can check that signature. This requires a Public Key Infrastructure (PKI) that acceptable to the participants. This paper will not cover PKI.

Using these guidelines, the solution proposed herein implements the IETF Draft RFCs for HTTP Message Signatures [9] and Digest Fields [10]. Despite being drafts, these standards represent the best known approach to signing HTTP messages. One of the drawbacks of previous approaches to message signing in a UTM context is that they only signed certain requests and certain responses depending on the signing approach or the qualities of the data exchanged. This approach allows for signing requests and responses in a uniform manner, even those that do not have a message body.

The approach provides:

> *  *A common nomenclature and canonicalization rule set for the different protocol elements and other components of HTTP messages, used to create the signature base.*
>
> *  *Algorithms for generating and verifying signatures over HTTP message components using this signature base through application of cryptographic primitives.*
>
> *  *A mechanism for attaching a signature and related metadata to an HTTP message, and for parsing attached signatures and metadata from HTTP messages.  To facilitate this, this document defines the "Signature-Input" and "Signature" fields. [9]*

For full details of how to apply the message signing, approach, see the RFC document.  To summarize briefly, the RFC prescribes how to indicate which fields are signed, how to properly name the fields, how to indicate the signing algorithm, the allowable signing algorithms, and related elements.

To apply the approach to UTM or other elements of UAS operation, some design decisions are still necessary when using the standard.  For example, decisions would need to be formalized for any context that answer questions as to which data elements are required/optional/unallowed, how key material is managed, how signature algorithms are indicated, and which requests require/forbid which algorithms.

Answering these questions defines a "profile" of the standard applied to a particular domain or application.  A standards or regulatory body may define a profile.  A likely scenario may collaboration between standards and regulations, with a regulator using a standard profile and making modifications.  Profiles may differ depending on the participating parties and the type of data exchanged.

This overall approach is applicable to any HTTP exchanges within any UTM ecosystem.

## Bibliography

[1] CANSO, "CANSO Standard of Excellence in Cybersecurity," Civil Air Navigation Services Organisation, 2020.

[2] NIST, "Security and Privacy Controls for Information Systems and Organizations," Department of Commerce, Washington, DC, 2020.

[3] J. L. Rios, I. S. Smith, P. Venkatesan, J. R. Homola, M. A. Johnson and J. Jung, "UAS Service Supplier Specification," Moffett Field, CA, 2019.

[4] J. L. Rios, I. Smith and P. Vekatesan, "UAS Service Supplier Framework for Authentication and Authorization," Moffett Field, CA, 2019.

[5] J. L. Rios and e. al., "UTM UAS Service Supplier Development Sprint 2 Toward Technical Capability Level 4," NASA, Moffett Field, CA, 2018.

[6] J. Rios, A. Aweiss, J. Homola, M. Johnson and R. Johnson, "Flight Demonstration of Unmanned Aircraft System (UAS) Traffic Management (UTM) at Technical Capability Level 4," in *AIAA AVIATION Forum* , Virtual, 2020.

[7] Federal Aviation Administration, "UPP Phase 2 Final Report," FAA, 2021.

[8] Mid-Atlantic Aviation Partnership, "Security Considerations for Operationalization of UTM Architecture," 2021.

[9] A. Backman, J. Richer and M. Sporny, "HTTP Message Signatures," 2022.

[10] R. Polli and L. Pardue, "Digest Fields," 2022.