NASA/TM–20220016658

# Non-Repudiation for Drone-Related Data

*Joseph L. Rios*
*Ames Research Center, Moffett Field, CA*

*Jaewoo Jung*
*Ames Research Center, Moffett Field, CA*

*Marcus A. Johnson*
*Ames Research Center, Moffett Field, CA*

**November 2022**

# NASA STI Program Report Series

The NASA STI Program collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NTRS Registered and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA Programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

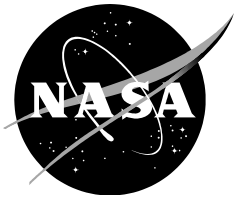
For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at http://www.sti.nasa.gov

- Help desk contact information:

https://www.sti.nasa.gov/sti-contact-form/ and select the "General" help request type.

NASA/TM–20220016658



# Non-Repudiation for Drone-Related Data

*Joseph L. Rios*
*Ames Research Center, Moffett Field, CA*

*Jaewoo Jung*
*Ames Research Center, Moffett Field, CA*

*Marcus A. Johnson*
*Ames Research Center, Moffett Field, CA*

National Aeronautics and
Space Administration

*Ames Research Center*
*Moffett Field, CA 94035-1000*

**November 2022**

## Acknowledgments

The authors wish to thank the Federal Aviation Administration for its on-going collaboration with NASA on all things related to uncrewed systems in the National Airspace System. In addition, the support of the Air Traffic Management – eXploration (ATM-X) Project at NASA was critical in the production of this Technical Memorandum. Finally, thanks to the internal reviewers at NASA for the thoughtful comments and corrections that made this paper that much better.

## 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 industries, the movement of data between entities is a common theme. While there is universal agreement on the 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.

## Introduction

There are many aspects to creating a secure system. The Civil Air Navigation Services Organisation (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 likely received the least attention, especially in the concepts and implementations of airspace management for UAS. It is not a coincidence that the CANSO list of security requirements align perfectly with the classic STRIDE[1] model for threat analysis [2], with CANSO adding a traceability requirement not present in the list of STRIDE's six threat categories. Repudiation is a long-known threat to the security of any system.

### What is Non-Repudiation?

Non-repudiation has several definitions, but they are all quite similar. Several documents[2] end up pointing to the National Institute of Standards and Technology (NIST) SP 800-53 Rev. 5 ("Security and Privacy Controls for Information Systems and Organizations") document [3] which defines it 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. Put into technical perspective, non-repudiation makes it unlikely or difficult for one party to successfully claim that it did not send a particular message when it, in fact, did send the message. Likewise, non-repudiation makes it unlikely or difficult for a party receiving a particular message to claim that it did not actually receive it.*

[1] STRIDE is an acronym and a mnemonic for the categories of security threats that the model purports to cover: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege.
[2] For example, ICAO's Cybersecurity Glossary of Terms [37] uses the NIST 800-53 Rev 2 definition, which is spiritually the same as the Rev. 5 definition cited in this document.

Non-repudiation takes on heightened importance in any federated system wherein multiple entities are exchanging data directly amongst themselves. For example, many UTM concepts involve the exchange of data between operators, often without a State-provided intermediary [4] [5] [6]. For ground-based systems, these data exchanges often leverage Hypertext Transfer Protocol (HTTP) with a Representational State Transfer (REST) [7] approach. REST relies on well-defined endpoints that accept and/or provide specific data elements. Note that this paper focuses on non-repudiation applied to REST because that is the dominant architecture 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." For an example of how blockchains may be leveraged to achieve non-repudiation, see Freeman et al. [8]

## 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") [9] and its associated Application Programming Interface (API) [10]. NASA, FAA, industry and others are working within ASTM on a related specification for larger, passenger-carrying UAS [11] that leverages the original ASTM work, adding new protocols and data models for the wider Advanced Air Mobility domain. The Federal Aviation Administration's (FAA's) Low Altitude Authorization and Notification Capability (LAANC) [12] also relies on HTTP and REST for communications. The Linux Foundation hosts the codebase for a Discovery and Synchronization Service (DSS) [13] that underpins several proposed and tested [14] [15] [16] 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 [17]; tested by the FAA [18] for U.S. use and SESAR in Europe [19] and is being developed/procured by Australia Air Services [20] for use in Australia, has relied on HTTP and REST for many data exchanges and likely will in the future. Recently, the Joint Authorities for Rulemaking on Unmanned Systems (JARUS) published a cyber annex [21] to the Specific Operations Risk Assessment (SORA) that calls out non-repudiation as an important attribute, but the focus there is on safety of specific operations rather than the management of air traffic.

Non-repudiation in the UTM domain has been studied in the United States by NASA and the FAA. In NASA's original specification for UAS Service Suppliers (USSs) [22], non-repudiation was considered by requiring digital signatures of operation plans as an approach to non-repudiation, with digital signatures provided by the Remote Pilot in Command (RPIC) as well as the vehicle for each shared operation plan:

> *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[3] that the exact method for digitally signing messages was yet to be determined.

NASA followed up in 2019 with a Technical Memorandum describing initial authentication and authorization requirements [23]. In that document, message signing was more fully defined and involved multiple HTTP exchanges. The approach was based on existing standards for JSON Web Signatures (JWS) [24] 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 partners in the UTM Project [25]. Message signing was included in further NASA flight testing during the Technical Capability Level 4 demonstration [26].The FAA continued such testing, especially as part of the UTM Pilot Program 2.0 (UPP2) [14]. A product of that 2020 event (written mostly by industry stakeholders with significant input from NASA and the FAA) was a detailed security analysis of USS communications [27]. That report touches on the value of non-repudiation and takes a deep dive into the Public Key Infrastructure (PKI) required to support it.

The use of NIST 800-53 Rev 5 (although recently being implemented and with nuances still being worked) to help organizations cover non-repudiation requirements, whether in the U.S. or elsewhere in the world, seems reasonable and justifiable given its more complete handling of the issue and the precedent for international aviation organizations to reference and leverage NIST documentation. The UPP2 industry-driven security report also leans on ISO/IEC 27001:2013 [28] as a key standard for security requirements. The UPP2 team sought to reference sources that may appear to have broader international acceptance, as opposed to the US-produced NIST documentation. In the US, the controls in NIST 800-53 are required for government systems per the Risk Management Framework [29] and these controls also may be required of service suppliers[4]. Regardless of what is ultimately required, cybersecurity best practices and potential domain-specific requirements may force security controls equivalent to what is found in NIST 800-53. It seems reasonable, even for non-U.S. entities, to look to NIST documentation for requirements guidance on non-repudiation (amongst other controls). As noted previously, CANSO references NIST documentation and ICAO uses NIST definitions. In addition, there is a mapping from ISO/IEC 27001-2013 to NIST 800-53 controls that is used in many contexts. However, control AU-10 in NIST 800-53 representing non-repudiation requirements do not have

[3] NASA's original draft was shared internally with UTM Project partners starting in 2015, with a public version published in 2019.

[4] NIST 800-37 notes that "FISMA and OMB policy require external providers that process, store, or transmit federal information or operate information systems on behalf of the federal government to meet the same security and privacy requirements as federal agencies. Federal security and privacy requirements also apply to external systems storing, processing, or transmitting federal information and any services provided by or associated with the external system. Furthermore, the assurance or confidence that the risk from using external providers is at an acceptable level depends on the trust that the organization places in the provider. In some instances, the level of trust is based on the amount of direct control the organization can exert on the provider regarding the employment of controls necessary to protect federal information and protect the privacy of individuals." NASA is not stating a position here in terms of what constitutes "federal information" nor whether service suppliers in UTM or related domains may constitute systems that are operated "on behalf of the federal government," but we are noting that it is important to have these conversations and converge on solutions as early as possible.

a direct mapping to any controls in ISO/IEC 27001:2013. However a couple of controls in the ISO document ("Collection of Evidence" and "Protection of Records") do map indirectly to AU-10 ("Non-Repudiation") and its sub elements in combination with other NIST controls.

It is fair to note that control AU-10 does not get applied in any baseline provided by NIST other than "high" indicating its need for high security systems. However, a given domain or application may tailor those baselines as needed. With the use cases provided herein as well as previous research cited throughout this document, together with the explicit need described by CANSO and others, a complete cybersecurity analysis is likely to find that at least some communications within an operational UTM system or service may require repudiation protections.

Given all of these systems exchanging data with other systems via APIs and REST calls, and the call for non-repudiation in aviation data systems, how do we reduce the likelihood of a repudiation attack? Granted that a holistic view of security is needed in aviation and the growing drone ecosystem and that repudiation attacks may rank lower than many other attack types in this ecosystem, it is important as a community of researchers, practitioners, regulators, and others that we work toward concrete solutions to cybersecurity issues. It is hoped that with this paper, we help patch up one more hole or provide one more slice of Swiss cheese on the stack to make a more secure and productive future for drone aviation.

## Problem Statement

Non-repudiation is a desired characteristic of many data exchanges. Currently in the proposed architectures and approaches throughout the world to support UAS traffic at scale, there has been little work on the issue of non-repudiation. Repudiation attacks are sometimes difficult to reason about since they do not protect day-to-day communications in the same way as, say, encryption or authentication might. In addition, repudiation attacks are not often associated with flight safety. Repudiation attacks are more likely to occur after some adverse, domain-specific event has taken place. In the domain of drones, one might imagine an event such as a collision or violation of airspace rules. Repudiation in these cases may be used as a deflection of responsibility. To make this argument more concrete, use cases are provided below.

Beyond the repercussions to 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. Why would organizations or individuals participate or support a system that has questions swirling around it related to the veracity of the data that are exchanged? Thus the acceptance and success of UTM and UAS operations at scale may rely, in part, on appropriate non-repudiation protections.

Figure 1 shows general entities within a UTM environment and is extensible to other similar domains such as Urban Air Mobility (UAM) or Upper Class E Traffic Management (ETM). The arrows indicate that there is an expected data exchange between the connected entities. The type of data that is exchanged is indicated in the yellow boxes along with a parenthetical label to reference specific use cases presented after the figure.

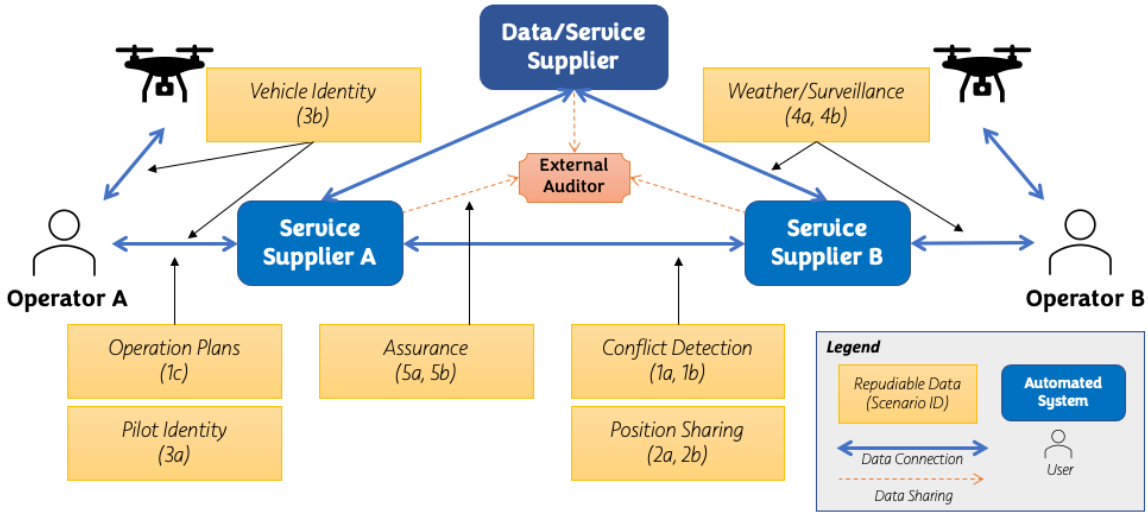



*Figure 1. Entities and their connections with a subset of the data that require non-repudiation protections.*

## Attack Use Case 1: Operation plan denial

In the ASTM standard for USS intent sharing, operators share their operation plans with each other via a USS. This increases awareness of other operations in the airspace for each operator and allows operators to make appropriate risk-based decisions about their operations. If two vehicles were to collide and then strike structures on the ground, there could be significant liability and regulatory impact to the operators.

### Scenario 1a: USS-USS denial

If Operator A is using its own in-house USS, USS A, and Operator B is using USS B, then USS A can claim to have not received the operation plan from USS B, thus impeding Operator A's ability to make an appropriate risk-based decision to operate. USS B may have a log that it sent the appropriate message to USS A, but these are typically just text files that are easily faked, as USS A might claim.

### Scenario 1b: USS-USS operation alteration

In the current implementation of the Discovery and Synchronization Server (DSS) by the Linux Foundation in support of ASTM standards, the details of operation plans are not stored. This is a feature of the system as it is designed to allow USSs to discover operations supported by other USSs that *may* interact with its own supported operations. The USSs must communicate with each other to obtain operation details. The representation within DSS for a given operation can be thought of as a bounding polygon or series of bounding polygons. There are an infinite number of detailed operation plans that are possible within any given representation within DSS. As such, it would be possible for USS A to claim it shared a plan with USS B that is different than what it actually shared and still remain consistent with data from DSS. This could sow doubt as to who owns responsibility in the event of a collision between vehicles.

### Scenario 1c: Operator-USS denial

If Operator A is using a third-party USS, USS C, then Operator A may claim that it never received the operation plan. Again, USS C may have system logs that indicate a message was

sent and received, but these are usually simple text files that are easily faked, as Operator A may claim.

Note that there are other attacks possible due to the communications and contracts between all the stakeholders in this simple scenario. For example, a pilot may repudiate that he or she was involved with the operation. A vehicle may be repudiated as being part of a particular operation. The non-repudiation of all actors may be long term goals of UTM as operations scale and the likelihood of such attacks increase. Initially it makes sense to secure USS-USS and USS-Operator communications against repudiation attacks as part of a larger strategy to secure the future UTM system.

### Attack Use Case 2: Position reports

In the ASTM standard for USS information sharing, there are cases where the position of the aircraft is required to be shared, notably in off-nominal situations.

#### Scenario 2a: Off-nominal positions not provided

Since the current ASTM specification has a pull-based mechanism for position data (i.e., a USS requests positions from another USS when allowed/needed), in an off-nominal operation, a USS may neglect to share position as required, but may claim that it did so. Another USS may have logs indicating that requests were made and not fulfilled, but the USS that was supposed to provide positions may deny that those requests were made.

#### Scenario 2b: Inaccurate positions shared

A USS may provide positions that are inaccurate (either intentionally or through a flaw in their system) to others. This may be discovered through observations by other operators or checked via independent surveillance. In such a case, a USS or operator may claim that the positions shared were accurate and may provide logs to that effect. These may be at odds with the positions that were retrieved by another USS, thus setting up a repudiation problem.

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

In the case of an adverse event involving the loss of a vehicle or a collision with a structure or other aircraft, it will be important to trace responsibility. In this use case, assume a vehicle smashes through an office building window causing injury to those inside.

#### Scenario 3a: Pilot denial

In a follow-on investigation, investigators would request logs and records to understand who the pilot-in-command was at the time of the collision. Without signed/secured logs, a pilot may refute that he or she was actually the pilot-in-command at that time.

#### Scenario 3b: Vehicle denial

While there may be digital evidence that the operation plan shared with the network was the one being flown at the time of the collision, it may be possible for an operator or pilot to claim that it was a different vehicle being flown at that time, further claiming that the operation in question was successfully flown and landed in accordance with the plan and the airspace rules. Their operation and vehicle were not involved in the incident. They may claim that the vehicle that

smashed the window was not being operated by them at that time and may have actually been stolen without their knowledge.

### 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. These dependencies may be the basis for a safety case or an operational approval from the regulator. As such, in an adverse situation, there may be legal or contractual ramifications related to data that are exchanged between the operator and the SDSP.

#### Scenario 4a: Inaccurate weather reports

A pilot flies into winds with predicted gusts that are beyond the acceptable operational parameters. The pilot performs this action despite weather data supplied by another party. The operational approval obtained by the operator from the regulator specifies that weather provider's data must be referenced prior to flight as part of the overall safety case. The pilot loses contact with the vehicle and sensor values indicate high winds forced the vehicle into the roof of a home. The pilot claims that the weather reports indicated calm winds and edits data files received from the weather provider to support this claim. The weather provider has logs indicating that high winds were very likely.

#### Scenario 4b: Surveillance targets not provided

An operator subscribes to a surveillance service that purports to provide targets for all airspace activity within a specific radius, capturing targets the size of large birds up to conventional aircraft. The operator, using this service, sees a clear airspace and commences a BVLOS operation from a remote location without the use of visual observers, per that operator's approvals from the regulator. The operator's vehicle is struck by a Cessna, causing significant damage to the Cessna and a total loss of the operator's vehicle. The surveillance operator claims that targets were successfully sent to the operator, while the UAS operator claims that its client to the surveillance operator was active and no such targets were received.

### Attack Use Case 5: Assurance and Audit

An important aspect enabling future BVLOS operations is the ability for an operator to assure it is following appropriate requirements to operate. This will likely involve providing evidence of compliance to some external entity, like the Civil Aviation Authority (CAA) or an insurance company. These scenarios are somewhat implicit in the previous scenarios in that these organizations are likely the organizations that are receiving the denials or repudiations as they look at adverse events involving operations. The following scenarios make these interactions more explicit.

#### Scenario 5a: CAA requires assurance of processes

A CAA may require regular evidence of compliance wherein non-reputable data is required to provide such evidence.

### Scenario 5b: Insurance provider requires evidence of compliance

An operator may have a need to carry appropriate insurance for BVLOS operations. An insurance provider may levy requirements related to the appropriate use of support services, like weather or surveillance, to continue providing coverage or to pay a particular claim. Non-reputable data is a likely requirement to indicate such support services are being properly leveraged.

## Proposed Solution

This document focuses on providing an approach to non-repudiation for HTTP REST communications since, as discussed above, this is a major form of communications within proposed UTM frameworks. HTTP is a request-response protocol, which aligns well with need to provide protections against denials that a message was sent as well as protections against claims that a message was not received. 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 such as latency, data volume, or other aspects of data security.

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 and will assume appropriate PKI exists to support the proposed approach.

Using these guidelines (or proto-requirements), the solution proposed herein is to implement the IETF Draft RFC for HTTP Message Signatures [30] together with the IETF Draft RFC for Digest Fields [31]. While these are still in draft form within IETF, they represent the best known approaches to standardizing how one can sign 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. The Draft RFCs' 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. [30]*

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, the following decisions would need to be formalized for any context:

1. Which component identifiers are required in a signature, which are optional, and which are unallowed?
2. How is key material obtained to verify signatures?
3. How does a verifier know what signature algorithm was used?
4. Which requests require (or allow or forbid) which algorithms?

Answering these questions defines a "profile" of the standard applied to a particular domain or application. A standards body may define a recommended profile and/or a regulatory body may define a required profile. A likely scenario may be a combination of both, with a regulator using a standard profile and making certain modifications. The profiles may differ depending on the participating parties and the type of data exchanged. An example profile for, say, USS-USS data exchanges might answer the questions thusly:

1. Required covered components. The field in the following table would be required per the application (UTM USS exchanges, etc.) to be covered by the signature algorithm. Some fields are well-defined in the HTTP standards, others are "derived components" that are described in the message signing RFC. The derived fields will have a leading '@' when they are put in normalized form for signing.

| Component | Example | Notes |
|---|---|---|
| **target-uri** | "@target-uri":\<br>https://www.example.com/path?param=value | The full absolute target URI of the request, potentially assembled from all available parts including the authority and request target [30]. |
| **method** | "@method": POST | The component label ("@method") is case-insensitive but provided in lowercase per convention, however the actual method (e.g. "POST") is case-sensitive per [32]. |
| **date** | "date": Tue, 20 Apr 2021 02:07:56 GMT | |
| **content-digest** | "content-digest": sha-512=:WZDPaVn/7XgHaAy8\<br>pmojAkGWoRx2UFChF41A2svX+TaPm+AbwAg\<br>BWnrIiYllu7BNNyealdVLvRwEmTHWXvJwew==: | The content-digest HTTP header as described in [31] |
| **content-length** | "content-length": 18 | Only required in signature when the content-length header is required in the HTTP message. |
| **status** | "@status": 200 | Must not be used in a request message [30]. This is a "derived component" because it is not a stand-alone, standardized HTTP field. |
| **req** | req=:vR1E+sDgh0J3dZyVdPc7mK0ZbEMW3N47eDpFjXLE9g95Gx1K\QLpdOmDQfedgdLzaFCqfD0WPn9e9/jubyUuZRw==: | Must be used for responses, must not be used in a request message. This is the signature of the request, which is to be included in a response to bind the request-response pair. |

*Table 1. Draft "covered components" for message signing.*

2. Key material is obtained via an agreed PKI with one or more defined, acceptable Certificate Authorities. If necessary due to PKI design decisions or constraints, an additional header may be needed to indicate information related to key exchange. That additional header needs to be a covered component of the signature. One concrete example might be the inclusion of a JWS JOSE header [24] that includes the 'kid' and 'x5u' parameters.
3. The signature algorithm is communicated via the 'alg' parameter in the Signature-Input field. If necessary due to PKI design decisions or constraints, an additional header may be needed to indicate information related to key exchange instead of using the 'alg' parameter. That additional header needs to be a covered component of the signature. One concrete example might be the inclusion of a JWS JOSE header [24] that includes the 'alg' parameter.

4. All requests and non-5XX responses are required to be signed using an algorithm from the IANA registry [33] titled "HTTP Signature Algorithms."

This proposed profile could be a reasonable starting point for any standards defining organization or regulator to begin considering adoption of message signing for HTTP communications. It is assumed that many more details will require consideration before implementing an operational version of non-repudiation.

## Requirements

The following requirements are an example of a profile that follows the IETF RFC. A standards defining organization may codify requirements such as these and a regulator may accept that standard or may modify the requirements per the needs of its airspace. These requirements have not been validated, but given the alignment with the IETF RFC are a likely reasonable start for the application of non-repudiation with UTM, UAM, Regional Air Mobility (RAM), ETM [34], Space Traffic Management (STM) [35], or any Extensible Traffic Management (XTM) [36] environment that leverage RESTful HTTP exchanges. There are likely gaps in this set of requirements that could be dutifully filled by the appropriate experts in the appropriate venue. For example, there are no incident response requirements listed, but those would certainly be part of a robust set of security requirements. Most of the 'MUST' statements in the IETF RFC are not replicated in this list, rather they are assumed per the higher-level requirements, though some may still be included herein for additional clarity. Most of the requirements provide detail to the message signing profile described above. There are new references to IANA [33] and HTTP [32] standards in these requirements. Parameters that should be set by appropriate organizations are indicated with a 'PARAM-XX' and are summarized after the requirements list. There are likely several other parameters that can be pulled from these draft requirements. The particular ones chosen should be viewed as an example of how certain details can be decided or documented.

| | |
|---|---|
| **XTM-SIG-01** | Messages exchanged via Hypertext Transmission Protocol shall be signed using the methodology described in the Messages Signatures RFC. |
| **XTM-SIG-01-01** | HTTP messages must contain a valid Signature-Input field supplied in the header. |
| **XTM-SIG-01-02** | HTTP messages shall contain a valid Signature HTTP field supplied in the header. |
| **XTM-SIG-01-03** | The label for the primary signature as described in Messages Signatures RFC shall be 'xtm-sig'. |
| **XTM-SIG-01-04** | The Signature-Input field shall contain the 'created' parameter as described in Messages Signatures RFC. |
| **XTM-SIG-01-05** | The Signature-Input field shall contain the 'expires' parameter as described in Messages Signatures RFC. |
| **XTM-SIG-01-06** | The Signature-Input filed shall contain the 'keyid' parameter as described in Messages Signatures RFC. |
| **XTM-SIG-01-07** | The Signature-Input field shall contain the 'alg' parameter as described in Messages Signatures RFC. |
| **XTM-SIG-01-08** | HTTP messages shall be signed using an algorithm from the IANA 'HTTP Signature Algorithms' registry. |
| **XTM-SIG-01-09** | HTTP messages with non-zero length content shall contain a Content-Digest header field as defined in the Digest Headers RFC. |
| **XTM-SIG-01-10** | The Content-Digest header field shall use the PARAM-01 algorithm. |

| | |
|---|---|
| **XTM-SIG-01-11** | HTTP messages shall contain Content-Length header as defined per HTTP RFC [31] when not expressly prohibited per HTTP standards. |
| **XTM-SIG-01-12** | All responses to signed requests shall implement 'request-response signature binding' as described in Messages Signatures RFC. |
| **XTM-SIG-01-13** | The target-uri shall be a covered component of a message signature in a request message. |
| **XTM-SIG-01-14** | The method shall be a covered component of a message signature for all messages. |
| **XTM-SIG-01-15** | The date shall be a covered component of a message signature for all messages. |
| **XTM-SIG-01-16** | The content-digest shall be a covered component of a message signature when content-digest is a header in the message. |
| **XTM-SIG-01-17** | The content-length shall be a covered component of a message signature when content-length is a header in the message. |
| **XTM-SIG-01-18** | The status shall be a covered component of a message signature for all response messages. |
| **XTM-SIG-01-19** | The req (as described in the Message Signing RFC) shall be a covered component for a response message. |
| **XTM-SIG-01-20** | The requestor shall keep a copy of the message signature for each request long enough to validate signed responses that may include that message signature. |
| **XTM-SIG-01-21** | All keys used in message signing must be of length PARAM-02 or greater. |
| **XTM-SIG-02** | Messages exchanged via Hypertext Transmission Protocol shall have all signatures validated by the receiver. |
| **XTM-SIG-02-01** | Request messages with missing signatures shall be sent a signed HTTP 403 (forbidden) response with a message body indicating a missing signature. |
| **XTM-SIG-02-02** | Request messages with an invalid signature shall be sent a signed HTTP 403 (forbidden) response with a message body indicating an invalid signature. |
| **XTM-SIG-02-03** | Response messages with an invalid or missing signature shall be reported by the owner of the requesting server to the appropriate oversight entity within PARAM-03 minutes. |
| **XTM-SIG-02-04** | Content-Length headers, when provided, shall be checked by the receiving server. |
| **XTM-SIG-02-05** | Content-Digest headers, when provided, shall be checked by the receiving server. |
| **XTM-SIG-02-06** | Request messages with an invalid or missing content-digest header shall be sent a signed HTTP 403 (forbidden) response with a message body indicating an invalid or missing content-digest header. |
| **XTM-SIG-02-07** | Request messages with an invalid or missing content-length header shall be sent a signed HTTP 403 (forbidden) response with a message body indicating an invalid or missing content-length header. |
| **XTM-SIG-02-08** | Response messages that cannot be correlated to a request sent by that server shall be reported by the owner of the requesting server to the appropriate oversight entity within PARAM-04 minutes. |
| **XTM-SIG-03** | All HTTP communications within scope of these requirements shall be logged. |
| **XTM-SIG-03-01** | All HTTP communication logs shall be archived for at least PARAM-04 days. |
| **XTM-SIG-03-02** | All HTTP communication logs shall be archived with their respective signatures. |
| **XTM-SIG-03-03** | All HTTP communication logs shall be protected via appropriate Data-at-Rest requirements. |

*Table 2. Draft requirements for nonrepudiation.*

| | |
|---|---|
| **PARAM-01** | Algorithm to be used in the creation of a Content-Digest header. |
| **PARAM-02** | Key length, in bits, required for signing messages. |
| **PARAM-03** | Maximum number of minutes to report invalid or missing response signatures. |
| **PARAM-04** | Minimum number of days that logs must be retained. |

*Table 3. Parameter definitions.*

## Concluding Remarks

This paper provides background, attack use cases, a proposed approach, and initial requirements related to non-repudiation in federated air traffic management environments that rely on HTTP communications between disparate entities. While the issue of non-repudiation may not be the most immediate need within UTM, UAM, RAM, ETM, or any XTM environment, its proper implementation may be vital to gain trust in any such system in the longer term. By implementing emerging standards from the Internet Engineering Task Force related to HTTP message security (i.e., message signing and message digests), a solid approach to non-repudiation is possible. The approaches presented herein are specific to HTTP communications, but the philosophies can aid in the cybersecurity of other message protocols (e.g., websockets, server-side events, gRPC) that may be present in current and future systems.

## Bibliography


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

[2] Microsoft, "The STRIDE Threat Model," 12 11 2009. [Online]. Available: https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20)?redirectedfrom=MSDN. [Accessed 09 06 2022].

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

[4] P. Kopardekar, J. Rios, T. Prevot, M. Johnson, J. Jung and J. E. Robinson III, "Unmanned Aircraft System Traffic Management (UTM) Concept of Operations," in *16th AIAA Aviation Technology, Integration, and Operations Conference*, Washington, D.C., 2016.

[5] Federal Aviation Administration, "Unmaned Aircraft Systems (UAS) Traffic Management (UTM) Concept of Operation v2.0," FAA, 2020.

[6] C. Barrado and e. al., "U-Space Concept of Operations: A Key Enabler for Opening Airspace to Emerging Low-Altitude Operations," *Aerospace,* vol. 7, no. 24, 2019.

[7] R. Fielding, "Architectural Styles and the Design of Network-based Software Architectures," Irvine, CA, 2000.

[8] K. Freeman, N. Gillem, A. Jones, B. Sridhar and N. Sharma, "Immutable Secure Data Exchange and Storage for Urban Air Mobility Environments," in *AIAA SciTech Forum*, San Diego, CA, 2022.

[9] ASTM, "Standard Specification for UAS Traffic Management (UTM) UAS Service Supplier (USS) Interoperability," 2021.

[10] ASTM, "UTM API (USS->DSS and USS->USS)," [Online]. Available: https://github.com/astm-utm/Protocol/blob/master/utm.yaml. [Accessed 06 2022].

[11] N. Craven and e. al., "X3 Simulation with National CampaignDevelopmental Test (NC-DT) Airspace Partners," NASA, Moffett Field, CA, 2021.

[12] FAA, "UAS Data Exchange (LAANC)," [Online]. Available: https://www.faa.gov/uas/programs_partnerships/data_exchange. [Accessed 10 06 2022].

[13] The Linux Foundation, "The InterUSS Project," [Online]. Available: https://interussplatform.org/. [Accessed 10 06 2022].

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

[15] Federal Aviation Administration, "UTM Field Test (UFT)," 2022. [Online]. Available: https://www.faa.gov/uas/research_development/traffic_management/field_test/. [Accessed 10 06 2022].

[16] Federal Office of Civil Aviation (FOCA), "SUSI's Remote ID Demonstration of September 16 2019," 2019.

[17] A. S. Aweiss, B. D. Owens, J. L. Rios, J. R. Homola and C. P. Mohlenbrink, "Unmanned Aircraft Systems (UAS) Traffic Management (UTM) National Campaign II," in *AIAA Information Systems-AIAA Infotech @ Aerospace*, Kissimmee, Florida, 2018.

[18] Federal Aviation Administration, "UPP Phase 1 Technical Report," 2020.

[19] SESAR Joint Undertaking, "SESAR 2020 GOF USPACE Summary FIMS Design and Architecture," 2020.

[20] Airservices Australia, "FIMS (PROTOTYPE) System Requirements Specification," 2021.

[21] Joint Authorities for Rulemaking on Unmanned Systems, "Annex E (Cyber)," JARUS, 2022.

[22] 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.

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

[24] M. Jones, J. Bradley and S. N., "JSON Web Signature (JWS), IETF Standard, RFC 7515," 2015.

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

[26] 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.

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

[28] International Organization for Standardization and Intenational Electrotechnical Commision, "Information technology — Security techniques — Information security management systems — Requirements," ISO/IEC, 2013.

[29] NIST, "Risk Management Framework for Information Systems and Organizations: A System Life Cycle Approach for Security and Privacy," 2018.

[30] A. Backman, J. Richer and M. Sporny, "HTTP Message Signatures, IETF Draft Standard, draft-ietf-httpbis-message-signatures-10," 2022.

[31] R. Polli and L. Pardue, "Digest Fields, IETF Draft Standard, draft-ietf-httpbis-digest-headers-10," 2022.

[32] R. Fielding, M. Nottingham and J. Reschke, "HTTP Semantics, IETF Standard, RFC 9110," 2022.

[33] R. Housley and O. Kolkman, "Principles for Operation of Internet Assigned Numbers Authority (IANA) Registries," 2020.

[34] NASA Aeronautics Research Institute, "2021 Upper Class E Traffic Management (ETM) Workshop," NASA, 2021. [Online]. Available: https://nari.arc.nasa.gov/etm2021workshop. [Accessed 11 July 2022].

[35] D. Murakami, S. Nag, M. Lifson and P. Kopardekar, "Space Traffic Management with a NASA UAS Traffic Management (UTM) Inspired Architecture," in *AIAA SciTech Forum*, San Diego, CA, 2019.

[36] J. Jung, J. X. M. Rios, J. Homola and P. Lee, "Overview of NASA's Extensible Traffic Management (xTM) Research," in *AIAA SciTech Forum*, San Diego, CA, 2022.

[37] ICAO, "ICAO Cybersecurity Glossary," 09 06 2022. [Online]. Available: https://www.icao.int/cybersecurity/lists/glossary/glossary.aspx. [Accessed 09 06 2022].