VAISALA

# Radiosonde message authentication: Security challenges and solutions

Technical Paper

# Ensuring reliable weather intelligence

This technical paper examines security vulnerabilities in radiosonde weather data transmissions and presents a practical authentication solution to protect against data manipulation.


**Introduction** 3

Global importance and vulnerabilities of radiosonde data ..... 3
Addressing radiosonde message security ..... 3

**Non-authentication: The security problem** 4

Radiosonde message formats ..... 4
The vulnerability factor ..... 4
Hameiri demonstration ..... 4

**Recommendation: Message authentication codes** 5

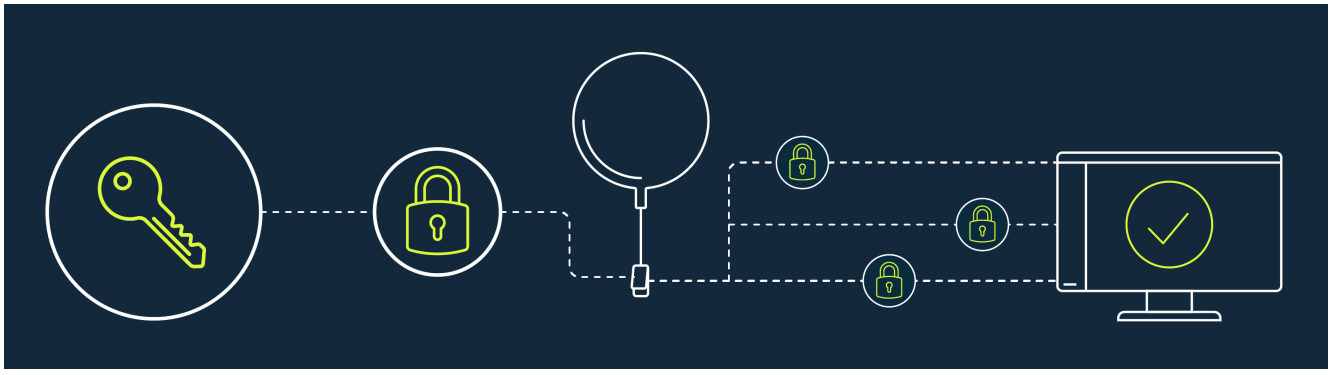
Generating the key ..... 5
Advantages of using HMAC-256 ..... 6

**Could public key-based authentication be used?** 7

Creating a digital signature ..... 7
Advantages of public key-based authentication ..... 8
Disadvantages of using public key-based authentication ..... 8

**Conclusion** 9

# Introduction



## Global importance and vulnerabilities of radiosonde data

Radiosondes are widely deployed around the world, with more than 800 launch sites and over 400,000 launches per year. These devices are attached to weather balloons and measure atmospheric parameters such as temperature, pressure, humidity, and wind speed and direction. They transmit the data to ground-based weather stations using radio signals, which meteorological organizations use for weather forecasting, climate research and aviation safety.

Historically, radiosonde messages have lacked any security mechanisms, meaning that anyone can send false or modified messages to the ground station, potentially compromising the integrity and reliability of the data. This poses a serious security threat, as malicious actors could manipulate the weather data for various purposes, such as disrupting air traffic, influencing financial markets, or causing public distrust.

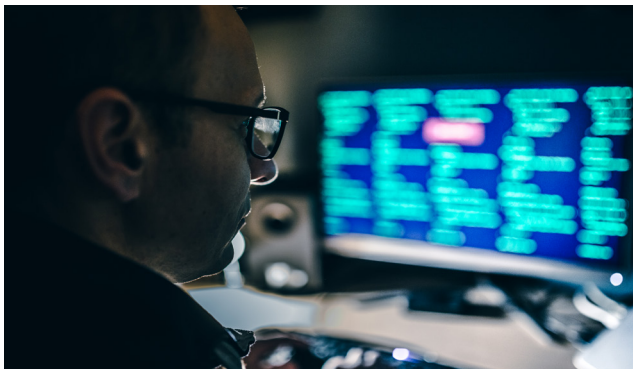
Encrypting radiosonde messages is one solution, but this would prevent data transmissions from being received by others such as radio amateurs and hobbyists. They collect the data and can also track the movements of radiosondes on the descent phase, retrieving fallen radiosondes for use in hobby projects.

## Addressing radiosonde message security

In this technical paper, we discuss the security problem with non-authenticated radiosonde messages and propose a solution using a hash-based message authentication code algorithm with a per-sounding generated random authentication key. We also compare this solution with public key-based authentication schemes such as the Edwards-curve digital signature algorithm and analyze their pros and cons in terms of security, performance and feasibility.

We base our discussion on the findings and demonstrations of Paz Hameiri, a security researcher who presented his work on radiosonde hacking at the DEF CON 31 conference in 2023[1]. Hameiri showed how easy and inexpensive it is to intercept, decode, modify and inject radiosonde messages using commercial off-the-shelf hardware and software tools. He also showed the impact of such attacks on weather data and weather models, as well as potential countermeasures and defenses.

# Non-authentication: The security problem



## Radiosonde message formats

Radiosonde messages are transmitted over 403 MHz or 1680 MHz UHF frequency bands using vendor-specific proprietary formats, which typically consist of:

- Header: information such as the station identifier, the launch time, and the message type
- Body: measured atmospheric parameters encoded as binary or alphanumeric data
- Trailer: a checksum of the header and body for error detection

Radiosonde message formats can also include more sophisticated error correcting codes, such as Reed-Solomon, and several checksums for different parts of the message. Checksums are intended to detect transmission errors, but do not provide any authentication or integrity protection.

Anyone with enough knowledge of the message format can forge a radiosonde message by generating a valid checksum or modify an existing message by changing the data and the checksum accordingly. The ground station has no way of verifying the authenticity and integrity of the received message and must trust that the message is from a legitimate radiosonde and has not been tampered with.

## The vulnerability factor

This makes radiosonde messages vulnerable to various types of attacks, such as replay, insertion, deletion, modification, and impersonation. For example, an attacker could intercept, modify and retransmit messages to make the ground station believe that wind speeds are lower than actual, creating unsafe conditions for certain weather-dependent missions.

These attacks could have serious consequences for people who use radiosonde data, such as meteorologists, pilots, air traffic controllers, defense forces, researchers and the public. For instance, manipulated weather data could lead to inaccurate forecasts, erroneous warnings, or misinformed decisions, affecting the safety, efficiency, and economy of various sectors and activities. Furthermore, compromised radiosonde data could undermine the credibility and the reputation of the data providers and the data users, eroding trust and confidence in weather information.

## Hameiri demonstration

Hameiri demonstrated some of these attacks using a software-defined radio (SDR) device, a laptop, and a custom-made software tool. He was able to capture, decode, modify, and retransmit radiosonde messages in real time, affecting the data displayed on the ground station and the weather websites. He estimated that the cost of his setup was less than US $100, and that the range of his attacks was up to 100 km.

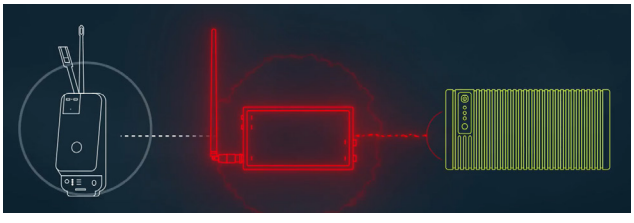


Figure 1: Software-defined radio sending falsified data to receiver

# Recommendation: Message authentication codes

To address the security problem with non-authenticated radiosonde messages, we propose using a hash-based message authentication algorithm, such as HMAC-256, with a per sounding generated random authentication key.

HMAC-256 is a cryptographic function that computes a message authentication code (MAC) based on a secret key and a message using the SHA-256 hash function[2,3]. The MAC can be used to verify the authenticity and integrity of the message, as only the sender and the receiver who share the same secret key can generate and validate the MAC. HMAC-256 is a widely used and standardized algorithm that offers strong security guarantees and resistance to various attacks.

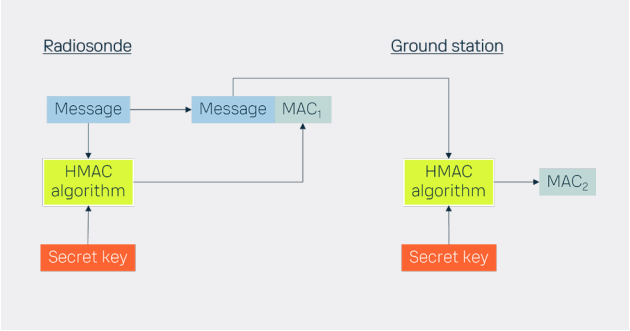



Figure 2: Message authentication code is calculated on both radiosonde and ground station. Messages are valid when $MAC_1$ is equal to $MAC_2$.

### Generating the key

The central idea of our recommendation is to generate a random authentication key for each radiosonde sounding and use it to compute the MAC for each radiosonde message. The key is generated by the ground station before the launch and is transmitted to the radiosonde during the ground preparation phase. The key is stored in the radiosonde and the ground station and is used to authenticate the messages during the sounding. The key is discarded after the sounding is completed, and a new key is generated for the next sounding.

1. The MAC is computed by applying the HMAC-256 algorithm to the radiosonde message with the authentication key as the secret key.

2. MAC is then added to the MAC-containing part of the header of the message.

3. The message with the MAC is transmitted to the ground station using the same radio channel as before.

4. The ground station receives the message and verifies the MAC by applying the HMAC-256 algorithm to the message, using the same authentication key as the radiosonde.

5. If the MAC matches, the message is accepted as authentic and intact. If the MAC does not match, the message is rejected as forged or corrupted.

While we recommend the use of HMAC-256, other standardized cryptographic hash algorithms, such as KECCAK Message Authentication Code (KMAC), could also be used.

### Advantages of using HMAC-256

1. Provides strong authentication and integrity protection for radiosonde messages, preventing various types of attacks such as insertion, modification, and impersonation.

2. Requires only small changes to the radiosonde message format. The header and the body of the message remain mostly unchanged, preserving the compatibility and the interoperability with the existing radiosonde systems.

3. Does not introduce any significant overhead or latency to the radiosonde transmission, as the MAC is only 32 bytes long and the HMAC-256 algorithm is fast and efficient.

4. The MAC can be truncated down to 8 bytes to reduce transmission overhead[4].

5. HMAC-256 algorithm is widely available in cryptographic software libraries, minimizing time to market and the possibility of security issues caused by implementation errors.

6. Does not rely on any external infrastructure or coordination, such as a central authority, public key infrastructure, or time synchronization service. The authentication key is generated and exchanged locally between the radiosonde and the ground station and is only valid for a single sounding.

7. Prevents malicious actors from learning the authentication key, if it would be possible to extract the key from a radiosonde, as authentication keys are generated randomly per sounding.

8. Does not prevent anyone who can receive the radio signals from decoding the radiosonde messages. However, the data is authenticated and verified by the ground station, ensuring the quality and the reliability of the data.

One disadvantage of using message authentication codes is the complexity of sharing the authentication key with several ground stations. However, in most cases there is only one ground station receiving the radiosonde transmission.

In his presentation, Hameiri suggested using message authentication codes to prevent falsifying radiosonde messages.

> The central idea of our recommendation is to generate a random authentication key for each radiosonde sounding and use it to compute the MAC for each radiosonde message.

# Could public key-based authentication be used?

An alternative solution to the security problem with non-authenticated radiosonde messages is to use a public key-based authentication scheme, such as Edwards-curve digital signature algorithm EdDSA[5].

Ed25519 is a cryptographic function, using SHA-512 hash function and Curve25519 elliptic curve, that computes an Edwards-curve digital signature based on a public key, private key pair, and a message.

It is possible to verify the authenticity and the integrity of the message using the digital signature. This is because only the sender who has the private key can generate the signature, and only the receiver who has the public key can validate the signature. Ed25519 is a state-of-the-art, standardized algorithm that offers high security, performance and usability.

The key idea of public key-based authentication is to generate a public and private key pair for each ground station and use them to sign and verify the radiosonde messages.

1. The key pair is generated by the ground station after installation or regenerated after a configured validity time, for example monthly or annually.

2. The private key is transmitted to the radiosonde during ground preparation phase.

3. The public key is stored in the ground station and is used to verify the messages during the sounding.

4. The private key is stored in the radiosonde and is used to sign the messages during the sounding.

### Creating a digital signature

The digital signature is computed by applying for example the Ed25519 algorithm to the header and the body of the radiosonde message, using the private key as the secret key.

1. The resulting digital signature is appended to the trailer of the message.

2. The message with the digital signature is then transmitted to the ground station using the same radio channel as before.

3. The ground station receives the message and verifies the digital signature by applying the Ed25519 algorithm to the header and the body of the message, using the public key as the verification key.

If the digital signature matches, the message is accepted as authentic and intact. If the digital signature does not match, the message is rejected as forged or corrupted

It is possible to verify the authenticity and the integrity of the message using the digital signature. This is because only the sender who has the private key can generate the signature, and only the receiver who has the public key can validate the signature.

### Advantages of public key-based authentication

1. Provides strong authentication and integrity protection for radiosonde messages, preventing various types of attacks such as insertion, modification, and impersonation.

2. Requires only small changes to the radiosonde message format. The header and the body of the message remain unchanged, preserving the compatibility with existing radiosonde systems.

3. Public key for message authentication could be shared with several ground stations and even with third parties who could then authenticate radiosonde messages independently.

4. Does not prevent anyone who can receive the radio signals from decoding the radiosonde messages. However, the data is authenticated and verified by the ground station, ensuring the quality and the reliability of the data.

### Disadvantages of using public key-based authentication

1. Introduces a significant overhead and latency to the radiosonde transmission, as the digital signature is usually longer than a message authentication code. For example, with Ed25519 the signature is 64 bytes without any truncation[6].

2. Requires the private key to be securely stored on the radiosonde, as a malicious actors will have physical access to fallen radiosondes and might be able to extract the private key.

3. Requires more computational and storage resources on the radiosonde. The public and private key pair is longer and more computationally intensive to generate and store than the authentication key.

A public key-based authentication system could even be extended to include a chain of trust from a certificate authority using for example standardized X.509 certificates. However, this will further increase the complexity of the solution and is not in the scope of this technical paper.

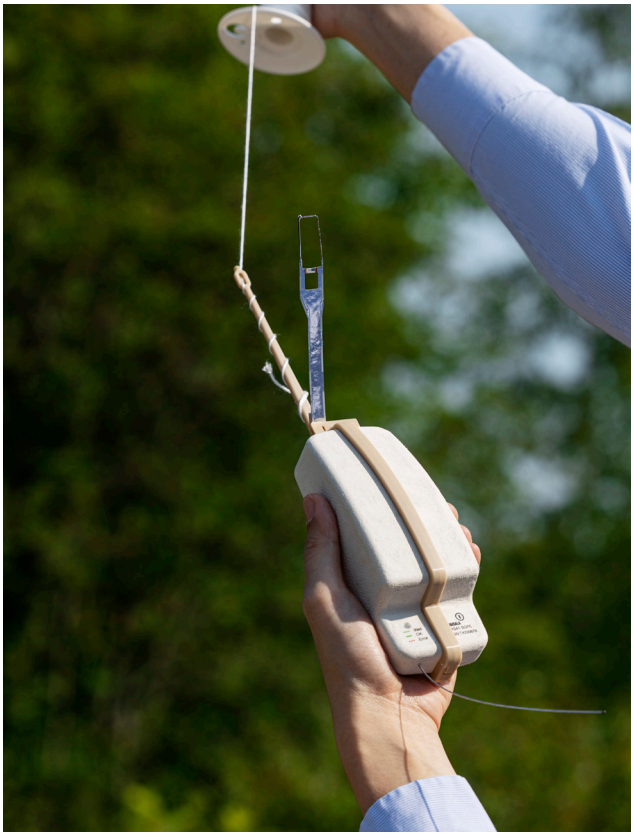
# Conclusion

In this technical paper, we have discussed the security challenges with non-authenticated radiosonde messages and recommend using the HMAC-256 algorithm, which Vaisala uses, to authenticate radiosonde messages with a per sounding generated random authentication key. We have also compared this solution with public key-based authentication schemes and analyzed their pros and cons in terms of security, performance, and feasibility.

Using HMAC-256 offers a simple, efficient and effective way to enhance the security and reliability of radiosonde data transmission. As more powerful processors and cryptographical hardware acceleration becomes feasible for cost-constrained radiosondes, public key based digital signatures could be used in the future.

Vaisala has implemented message authentication using the HMAC-256 algorithm to the latest versions of Vaisala Radiosonde RS41 and Vaisala Cirrus® Sounding System MW51. We recommend that radiosonde and sounding system users adopt our solution to improve the quality and trustworthiness of their weather information systems.


The Vaisala Cirrus Sounding System MW51 uses industry-leading security measures to protect against data threats and unauthorized access.



Multi-GNSS technology in our standard RS41 radiosonde models uses data from multiple navigation satellites to increase resilience against GPS interference. Unlock these advancements using Cirrus Sounding System MW51.

## References


1. P. Hameiri, "CON Trolling The Weather," presented at the DEF CON 31, Las Vegas, NV, USA, 2023.
2. J. Kelsey, B. Schneier, and N. Ferguson, "The Secure Hash Algorithm (SHA-256)," in Handbook of Applied Cryptography, 2nd ed., A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, Eds. Boca Raton, FL, USA: CRC Press, 2001, pp. 449–463.
3. H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication," RFC 2104, Feb. 1997.
4. Q. Dang, NIST Special Publication 800-107, Revision 1, Recommendation for Applications Using Approved Hash Algorithms, Aug. 2012, pp. 14.
5. S. Josefsson and I. Liusvaara, "Edwards-Curve Digital Signature Algorithm (EdDSA)," RFC 8032, Jan. 2017.
6. T. Pornin, "Truncated EdDSA/ECDSA Signatures", Jul. 2022.


Ref. B213037EN-A