

# COSPAS Search and Rescue Satellite Uplink: A MAC-Based Security Enhancement

Syed Khandker  
New York University Abu Dhabi  
syed.khandker@nyu.edu

Krzysztof Jurczok  
Amateur Radio Operator  
sq3dho@gmail.com

Christina Pöpper  
New York University Abu Dhabi  
christina.poepper@nyu.edu

**Abstract**—COSPAS-Sarsat is a global satellite-based search and rescue system that provides distress alert and location information to aid in the rescue of people in distress. However, recent studies show that the system lacks proper security mechanisms, making it vulnerable to various cyberattacks, including beacon spoofing, hacking, frequency jamming, and more. This paper proposes a backward-compatible solution to address these longstanding security concerns by incorporating a message authentication code (MAC) and timestamp. The MAC and timestamp ensure the integrity and authenticity of distress signals, while backward compatibility enables seamless integration with existing beacons. The proposed solution was evaluated in both a laboratory setting and a real-world satellite environment, demonstrating its practicality and effectiveness. Experimental results indicate that our solution can effectively prevent attacks such as spoofing, man-in-the-middle, and replay attacks. This solution represents a significant step toward enhancing the security of COSPAS-Sarsat beacon communication, making it more resilient to cyberattacks, and ensuring the timely and accurate delivery of distress signals to search and rescue authorities.

## I. INTRODUCTION

Satellite communication is a critical technology for search and rescue (SAR) operations, providing a reliable and ubiquitous means of communication and data transmission in even the most remote and challenging environments. By enabling real-time communication and information sharing between emergency responders, satellite communication helps to coordinate their efforts more effectively, ultimately increasing the likelihood of a successful rescue. A key application of satellite communication in SAR is the detection and location of distress beacons. These devices, carried by aircraft, ships, and individuals, transmit emergency signals when activated. Satellite-based SAR systems can detect and locate these signals within minutes, providing vital information to emergency responders.

The origins of the SAR system can be traced back to the 1960s, with the use of portable radio transmitters in light aircraft and some marine vessels. Operating at the international distress frequency of 121.5 MHz, these transmitters offered line-of-sight communication, allowing distress signals to be picked up by nearby air traffic control towers or other aircraft. Seeking wider coverage, COSPAS-Sarsat began operation in

1982. It is a global satellite-based SAR system that provides distress alert and location information to aid in the rescue of people in distress [1]. It is a humanitarian cooperative of 45 nations and agencies, which is gradually expanding. Aiming to increase SAR efficiency, it is in cooperation with other satellite services such as NOAA, EUMETSAT, and INSAT for transponder sharing and weather updates, GPS, Glonass, and Galileo for location service, while COSPAS plays the main role in receiving distress signals and informing local SAR authorities for conducting rescue operations.

COSPAS-Sarsat initially supported the 121.5 MHz frequency. However, location accuracy using this older technology, relying on Doppler shift, resulted in an accuracy of approximately 2 km, often leading to unreliable results and impacting rescue efforts. On Feb 1, 2009, the international council of COSPAS-Sarsat decided to phase out this analog technology. By that time, 406.025 MHz digital beacons became fully operational, where location information is encoded into the signal. When satellites receive a beacon signal, it is stored onboard and retransmitted to each ground station as the satellite orbits the Earth. Initially, COSPAS relied solely on Low Earth Orbit (LEO) satellites. However, due to low altitude, LEO satellites' footprint is small, covering a small portion of the Earth's surface at a given time. To address this, COSPAS incorporated Geostationary Earth Orbit (GEO) satellites, expanding coverage significantly. Today, both LEO and GEO satellites seamlessly cooperate with the modern 406 MHz digital signals, pinpointing distress beacons through the encoded location information. Though currently not available, Medium Earth Orbit (MEO) satellites are part of future plans for COSPAS-Sarsat for better coverage and performance.

COSPAS-Sarsat supports three types of beacons, namely Emergency Locator Transmitter (ELT) for aircraft, Emergency Position-Indicating Radio Beacon (EPIRB) for marine vessels, and Personal Locator Beacon (PLB) for individual humans. When a distress beacon is activated, it transmits a signal on the 406 MHz frequency. This signal is detected by satellites in the COSPAS-Sarsat system, which then relay the signal to ground stations around the world. The ground stations use the encoded location information from the signal to determine the distress beacon's location, which is then relayed to SAR authorities. COSPAS-Sarsat is a highly effective SAR system, and it has helped to save over 50,000 lives since it became operational in 1982. It is a vital tool for SAR authorities, and it plays a critical role in keeping people safe in remote and dangerous environments.

Despite its long history of success and critical role in

SAR operations, the security of this satellite system has not been adequately analyzed. It lacks proper security mechanisms, making it vulnerable to various cyberattacks, including spoofing [2], hacking [3], frequency jamming [4], and more. The COSPAS-Sarsat system was designed and developed approximately forty years ago when satellite communication began to emerge. Resources, capacity, and connectivity were limited at that time. Moreover, adversarial bodies or hackers were extremely few in number, which may have influenced the designers to keep the design as simple as possible. However, four decades later, communication and security systems have advanced significantly. Additionally, new attacking tools have emerged, posing unprecedented challenges to almost all communication systems from the past century. In this paper, we aim to address these longstanding security issues using a simple approach based on message authentication codes (MAC) and timestamps. One of the main challenges is that COSPAS-Sarsat’s open-source software is not readily available. We encountered only one property-based beacon signal decoding software, which is not suitable for the security experiments. Therefore, we had to develop our own software for the experiments mentioned in this paper. We tested the proposed solution in both a laboratory and a real-world satellite environment, demonstrating the effectiveness of our approach. Our contributions in this paper are as follows:

- We developed a COSPAS-Sarsat beacon encoding, decoding, and validating program.
- To the best of our knowledge, using our MAC-based security implementation, we are the first to demonstrate the mitigation of attacks on COSPAS-Sarsat using a LEO satellite.

The rest of this paper is organized as follows. In Section II we discuss related works. Technical details are presented in Section III. Our attacker model is described in Section IV. Then we present the proposed method in Section V. Experiments and results are presented in Section VI. We discuss the challenges in Section VII. Finally, we conclude the paper with Section VIII. To support further research and investigations on this topic, we make the code available on GitHub at <https://github.com/s21sm/SpaceSec24>.

## II. RELATED WORK

The COSPAS-Sarsat system operates by detecting signals from emergency beacons. This is similar to other radio-frequency (RF) based surveillance systems like Automatic Dependent Surveillance-Broadcast (ADS-B) and Automatic Identification System (AIS), which are employed to report the positions of aircraft and ships, respectively. The main difference is that the latter group uses ground or air communication while COSPAS-Sarsat employs satellite communication. Nevertheless, both ADS-B and AIS were reported to have been hacked over a decade ago [6], [7]. Satellite communication system (SCS) hacking is not uncommon these days. The vulnerabilities of SCS primarily stem from outdated designs and the emergence of sophisticated hacking tools. Satellites are typically located at a remote distance, and ground-based receivers are distributed among various countries. This makes it challenging to implement substantial security changes after satellite deployment. Furthermore, the growing availability of

information, enhanced coordination among adversaries, and the advancement of radio technology tools like Software Defined Radio (SDR) have made satellite hacking attacks more rigorous and feasible. SDRs can generate targeted RF signals at a very low cost and effort [8], which can be quite challenging for embedded circuits. Additionally, wireless communication has long been an attractive target for malicious actors due to its susceptibility to remote exploitation without the need for physical intervention. Incidents of attacks on communication technologies such as 5G, Wi-Fi, and satellite communication have been steadily increasing. Satellite hacking has emerged as an integral component of contemporary cyberwarfare strategies. Nations involved in conflicts are increasingly deploying satellite hacking techniques against each other to gain a strategic advantage [9], [10]. These reports underscore the significant risks posed by security vulnerabilities within SCS. Consequently, ensuring robust and up-to-date security measures in this domain is of paramount importance. Numerous instances of satellite hacking and exploitation have been reported, including military satellite hacking [11], commercial satellite hacking [12], Iridium hacking [13], GPS hacking [14], and SkyNet hacking [15]. These reports underscore the significant risks posed by security vulnerabilities within SCS. Consequently, ensuring robust and up-to-date security measures in this domain is of utmost importance.

The authors in [16] discuss the feasibility of spoofing attacks against satellite downlink communication systems. They examine various types of satellites, including GNSS, telecommunication, Earth observation, and cubesats. Their findings demonstrate the feasibility of signal overshadowing attacks against all these satellite systems across long distances. They propose cryptographic authentication on both the uplink and downlink with regular updates as a potential solution. Yue et al. [17] discuss in detail the security and reliability challenges of LEO SCS. They argue that LEO SCSs are vulnerable to a diverse array of security threats, including eavesdropping, jamming, spoofing, and data tampering. They also note that LEO SCSs are prone to reliability risks, such as space debris collisions, solar flares, and hardware failures. They recommend the use of physical-layer security techniques, such as spread-spectrum modulation and artificial noise, for communication reliability and counteracting eavesdropping. They also suggest using cryptography and blockchains to protect data confidentiality and integrity. Kodheli et al. [18] surveyed the state-of-the-art in satellite communications, highlighting the key trends and challenges. They found that space communications are undergoing a renaissance, driven by technological advances, private investment, and new applications. For example, there has been a recent surge of interest in developing large LEO constellations that can deliver high-throughput broadband services with low latency. Major companies such as SpaceX, Amazon, OneWeb, and TeleSAT are all vying to build their own constellations. In response to this growing interest and the increasing importance of satellite communications, new encryption algorithms are being developed to protect SatCom data from eavesdropping, and new network architectures are being designed to be more resilient to attack. Bernsmed et al. [19] discuss the challenges of securing multimodal communication. Multimodal communication refers to the use of multiple communication technologies, such as VHF data exchange system (VDES), VDES-terrestrial, VDES-satellite,

	Bit Synchronization	Frame Synchronization	First Protected Data Field (PDF-1)				BCH-1	Second Protected Data Field (PDF-2)	BCH-2
Unmodulated Carrier (160 ms)	Bit Synchronization Pattern	Frame Synchronization Pattern	Format Flag	Protocol Flag	Country Code	Identification or Identification plus Position	21-Bit BCH Code	Supplementary and Position or National Use Data	12-Bit BCH Code
Bit No.	1-15	16-24	25	26	27-36	37-85	86-106	107-132	133-144
	15 bits	9 bits	1 bit	1 bit	10 bits	49 bits	21 bits	26 bits	12 bits

Fig. 1: COSPAS-Sarsat beacon’s uplink long message structure according to specification [5]

and AIS. This can be beneficial for coordinating search and rescue operations, sharing maritime traffic data, and other maritime activities. However, it also raises security challenges, as it can be difficult to ensure that communication is secure across different technologies and actors. They propose using a public key infrastructure (PKI) to verify the identity of a sender and the integrity of a message.

Yuqi et al. [4] investigated the interference of public walkie-talkies on the COSPAS-Sarsat system. They discovered that, due to adjacent-frequency-band walkie-talkies, there is extensive interference on COSPAS-Sarsat’s uplink in China, particularly at the medium earth orbit (MEO) satellite. Additionally, they reported that broadband interference from public walkie-talkies is more severe than stray interference. Mladenov et al. [20] demonstrated the implementation of a GNU Radio-based EPIRB receiver. Their goal was to develop a software-configurable SAR transponder on the satellite that could receive and decode emergency transmissions from terrestrial distress beacons and relay them to mission control. The SDR payload and UHF monopole antenna of OPS-SAT were used to receive the terrestrial RF transmissions from beacons in the 406 MHz band. The onboard processing involved acquiring in-phase and quadrature (IQ) samples from the SDR, decoding the beacon messages using GNU Radio libraries, and writing the decoded metadata and raw IQ samples to permanent storage. The decoded metadata was then downloaded to mission control at the European Space Operations Centre. The experiment was successful, and the results demonstrate the feasibility of using GNU Radio for in-orbit SAR signal processing. Costin et al. [2] spoofed COSPAS-Sarsat beacons using a HackRF device. Their over-the-air experiment demonstrated that the absence of security mechanisms makes it possible to spoof the signal. This was verified using a third-party beacon decoder called EPIRB Plotter [21].

### III. TECHNICAL DETAILS

The early history of SAR began in the 1960s. During that time, commercial aviation used the analog international distress frequency of 121.5 MHz, while the military used 243 MHz. However, these services were limited to specific regions. To improve and modernize distress signal capabilities and expand the service area, COSPAS-Sarsat was established in 1975 and initiated its operations in 1982, utilizing the 121.5 MHz analog and later adopting 406.025 MHz digital uplink channel. It uses two types of satellites to detect distress signals: LEO and GEO satellites. LEO satellites orbit the Earth at an

altitude of about 800 kilometers, while GEO satellites orbit at an altitude of about 36,000 kilometers.

The Cospas beacon message comes in two variants based on length: a concise version with 112 bits and a more detailed version with 144 bits. The data rate is 400 bits per second, resulting in 280 milliseconds for the short message and 360 milliseconds for the long message. The RF signal consists of a 160-millisecond unmodulated carrier followed by either type of message. Messages start with a 15-bit preamble where all the bits are set to 1. Beacons are activated only in emergency situations, which occur very rarely. Additionally, after maintenance tasks, such as cleaning or changing the battery, users may want to check if the beacon is functioning correctly or might accidentally press the activation button, potentially causing a false alarm. To prevent such situations, two modes are utilized: normal mode and test mode. The frame synchronization bits define the mode, with 011010000 indicating the test mode and 000101111 indicating the normal mode. If a test mode message is received, it can be considered a test, and no further action is required. Figure 1 shows a long message structure according to [5]. Two segments of Bose-Chaudhuri-Hocquenghem (BCH) codes are used in the long message, that allow the satellite to detect and correct errors by comparing the received data with the redundant bits.

The message employs Manchester encoding, a binary encoding scheme used to transmit data over a communication channel. This encoding method ensures synchronization between the sender and receiver by representing each bit of data as both high and low voltage levels within a fixed time period. Cospas beacons utilize the biphasic-L variant, where each bit period contains a transition in the middle. In this variant, for a logical 0-bit, the signal levels are low-high, with a low level in the first half of the bit period and a high level in the second half. Conversely, for a logical 1-bit, the signal levels are high-low, with a high level in the first half and a low level in the second half. The carrier is modulated using phase modulation, with a positive peak of +1.1 radians and a negative peak of -1.1 radians. This type of modulation closely resembles Narrowband Frequency Modulation (NFM) when the message signal contains only binary information. In such cases, the message signal modulates the carrier signal by varying two different frequencies. Consequently, EPIRB plotter [21] operates by tuning to NFM to decode and process the transmitted data. Similarly, if an audio signal is created with Manchester encoded COSPAS-Sarsat binary data and transmitted using NFM modulation, the outcome would be the

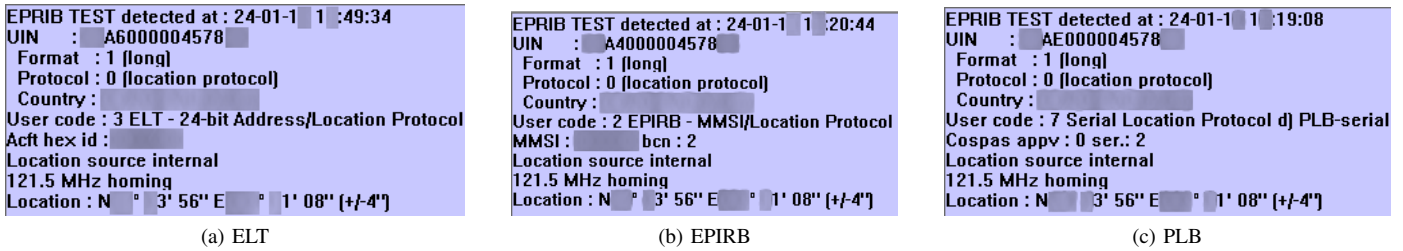


Fig. 2: Spoofed COSPAS-Sarsat beacon messages on EPIRB plotter

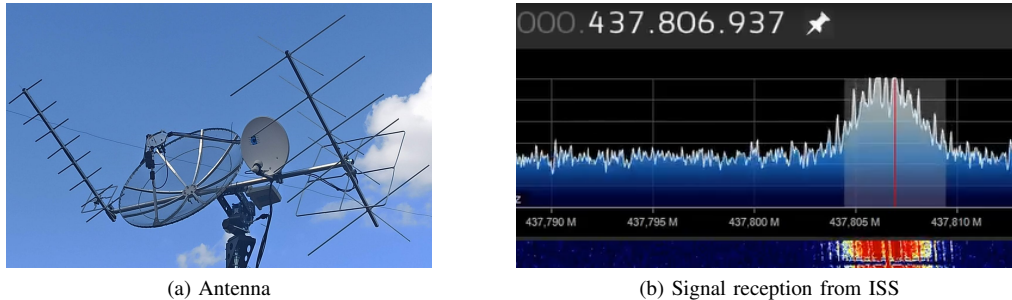


Fig. 3: Signal transmission and reception

same.

**Preliminary Experiments.** We conducted experiments for both transmission types (e.g., phase modulation and NFM) and obtained identical results. Figure 2 depicts our spoofed signal on EPIRB plotter for both transmission types and show the locations of the beacons that are supposed to be in distress. Some location-leaking information has been blurred for anonymity reasons.

COSPAS-Sarsat offers support for three distinct types of emergency beacons: ELT, EPIRB, and PLB. ELTs are primarily designed for aircraft and serve a critical role in aviation safety. They can be triggered either automatically or manually in the unfortunate event of a physical impact, such as a plane crash. These beacons are identified by the unique aircraft hex code, which serves as their beacon identity. The protocol code for ELTs is 0011, encoded in the 37th to 40th bit. EPIRBs are commonly utilized within the maritime industry and are often carried aboard passenger or cargo ships. These beacons share a similar message structure to ELTs, but they use the Maritime Mobile Service Identity (MMSI) of the marine vessel as their identifier. EPIRBs are associated with the protocol code 0010. PLBs are intended for use by individuals who may find themselves in situations requiring emergency evacuation, such as desert adventures or cross-country skiing. The protocol code for PLBs is 0111. Each of these beacon types plays a crucial role in ensuring the safety and timely response to emergencies across different domains, be it in the air, on the water, or in remote wilderness areas.

#### IV. ATTACKER MODEL

We assume that an attacker has full knowledge of the COSPAS-Sarsat protocol, the ability to track down the satellite,

and is equipped with an SDR and antenna that allows them to generate and transmit a targeted radio signal. Since the current protocol lacks any security measures, it would not be difficult for an attacker to generate a fake signal, as we have demonstrated. Therefore, attacks can produce a fake signal and transmit it to the satellite to create a false alarm. Additionally, an attacker can record a signal from a legitimate beacon and conduct a replay attack. Furthermore, a transmission-capable SDR is a highly effective attacking tool. In embedded electronic circuits, it is difficult to change the encoded message; however, an SDR can be controlled using a computer program, enabling the creation of a vast number of fake signals with different beacon IDs. This could lead to a flooding attack, and using many SDRs in a flooding attack could result in a Denial of Service (DoS) attack. Since attackers know the orbital time and frequency, at the right time and frequency, they can transmit a high-power noise signal to create jamming and interference. The attackers do not necessarily need to be positioned on the ground. The needed equipment to conduct the attack is simple and light and can be fitted into a drone, making it hard to detect. Additionally, co-orbital satellites can be used to conduct attacks from satellite to satellite.

#### V. PROPOSED METHOD

In the preceding sections, we presented technical details, attacker model, and demonstrated a spoofing attack. In this section, we outline our methodology for ensuring security, before presenting our experimental results in Section VI.

Satellite communications employ various security measures to safeguard the integrity, confidentiality, and availability of transmitted data. Cryptographic techniques play a crucial role in enhancing the overall security posture of satellite networks, protecting against unauthorized access, data manipulation, and

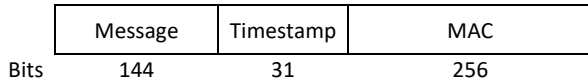


Fig. 4: Proposed message structure

other security threats. As part of cryptographic solutions, MACs, digital signatures, and encryption are widely employed in SCS. The specific choice between MACs or digital signatures may depend on factors such as the cryptographic algorithms used, the desired level of security, and available computational resources. However, encryption might not be a suitable choice for the COSPAS-Sarsat system, as it would require a complete overhaul of all existing beacons worldwide. In this case, a backward-compatible solution would be more effective. In our proposed method, we opted for using MAC to safeguard the security of COSPAS-Sarsat uplink communication. Specifically, we used a 256-bit Hash-based Message Authentication Code (HMAC) employing the SHA-256 hash function. The reason behind our selection is that this type of MAC is designed to resist various cryptographic attacks, including collision attacks. Using a strong hash function (e.g., SHA-256) enhances the security of the MAC. Besides, MACs are generally computationally less expensive than digital signatures. They are also a good fit for two-party communication, in this case, between the beacon and the satellite. Digital signatures require a PKI-like infrastructure, which is complex to design and cumbersome to maintain.

MAC is a cryptographic checksum that is applied to a message to ensure its integrity and authenticity. It is a short piece of information that is generated using a secret key shared between the sender and receiver. The MAC is appended to the message and transmitted along with it. Upon receiving the message, the receiver generates their own MAC using the secret key and compares it to the received MAC. If the two MACs match, then the receiver can be assured that the message is authentic and has not been tampered with. While MAC ensures integrity and authenticity, an adversary can still use recorded transmissions to launch replay attacks. To mitigate such attacks, we propose incorporating a timestamp into the message. If the timestamp is outdated for a certain threshold at the receiver end, the message will not be validated. For a time-dependent system of this nature, having a proper reference clock is crucial for time synchronization purposes. Ground-based beacons can utilize GNSS-based services (e.g., GPS) to obtain accurate time. On the backend, satellites will offload the signal to the ground station, where the GPS or the National Institute of Standards and Technology (NIST) time service will be readily available via the Internet. The proposed message structure is illustrated in Figure 4.

According to our proposal, the future satellite transponder can function as a bent pipe. It receives signals from the beacons and transports them to the satellite's ground station. This enables the satellite to offload additional tasks, such as key retrieval and verification, on board, thereby increasing overall satellite efficiency. Now, when the ground station receives a message, it needs to verify the MAC and timestamp. The signal from the advanced generation beacon will send a message with a timestamp and MAC. The satellite system would first check

the MAC. If the MAC is valid, it would then proceed to check the timestamp. If the time difference between the message's timestamp and the current timestamp at the receiver end is within a pre-defined threshold, then the timestamp is also valid, considering it as a legitimate message. However, when legacy beacons send a message, there will be no MAC or timestamp. This can be detected from the signal, as shown in Figure 5c where, in the absence of data, the signal does not have any amplitude. Therefore, no MAC or timestamp checking would be done for old-generation beacons. If an attacker removes the MAC and replays the message without the MAC, pretending to be a legacy device, upon decoding the message, based on the decoded beacon ID the satellite system would detect that the message comes from a new generation beacon but does not have a valid MAC. Therefore, the message would be discarded. Thus, the system would support both generation beacons, ensuring backwards compatibility. However, old-generation beacons will be given sufficient time to be upgraded to the secure version.

## VI. EXPERIMENT AND RESULTS

In this section, we describe the testbed, experiment, and results. We conducted the test in both laboratory and real-satellite environments. In the laboratory setting, we used COSPAS-Sarsat frequency but within a Faraday cage. For the real-satellite transmission, we utilized an amateur radio frequency, holding a valid amateur radio license.

### A. Testbed

Our test-bed comprises two laptops: one for transmission and another for reception, a HackRF, an RTL-SDR, a Faraday cage, a Yaesu FT-991A transceiver, and a dual-band VHF-UHF antenna system. For the software solution, we employed GNU Radio Companion [22], SDR Sharp [23], and EPIRB plotter [21] for signal transmission, reception, and decoding, respectively. Additionally, we developed our own Python-based program to encode, decode, and validate the beacon signal. To test the proposed concept in a real satellite system, we require a satellite that facilitates NFM signal reception and transmission back to Earth. Generally, commercial satellites are not accessible to the ordinary user. However, certain Amateur Satellites (AMSAT) [24] are available for such testing. Among them, the International Space Station (ISS) amateur radio module stands out as the most suitable candidate for the experiment, as other AMSATs are either malfunctioning or operate in incompatible modes. We utilized the ISS's cross-band repeater to get the signal back from space.

### B. Laboratory Experiment

To implement the proposed idea, we initially generated a COSPAS-Sarsat beacon message. A Python program was developed to create all three types of beacon messages. Then a timestamp was added to the message. Subsequently, we generated a shared key for the beacon and satellite. Finally, the MAC was calculated for the message and timestamp and appended to create the total payload, as depicted in Figure 4. The payload was transmitted into the air using three methods:

- 1) via HackRF using a phase modulation through a GRC script [25],

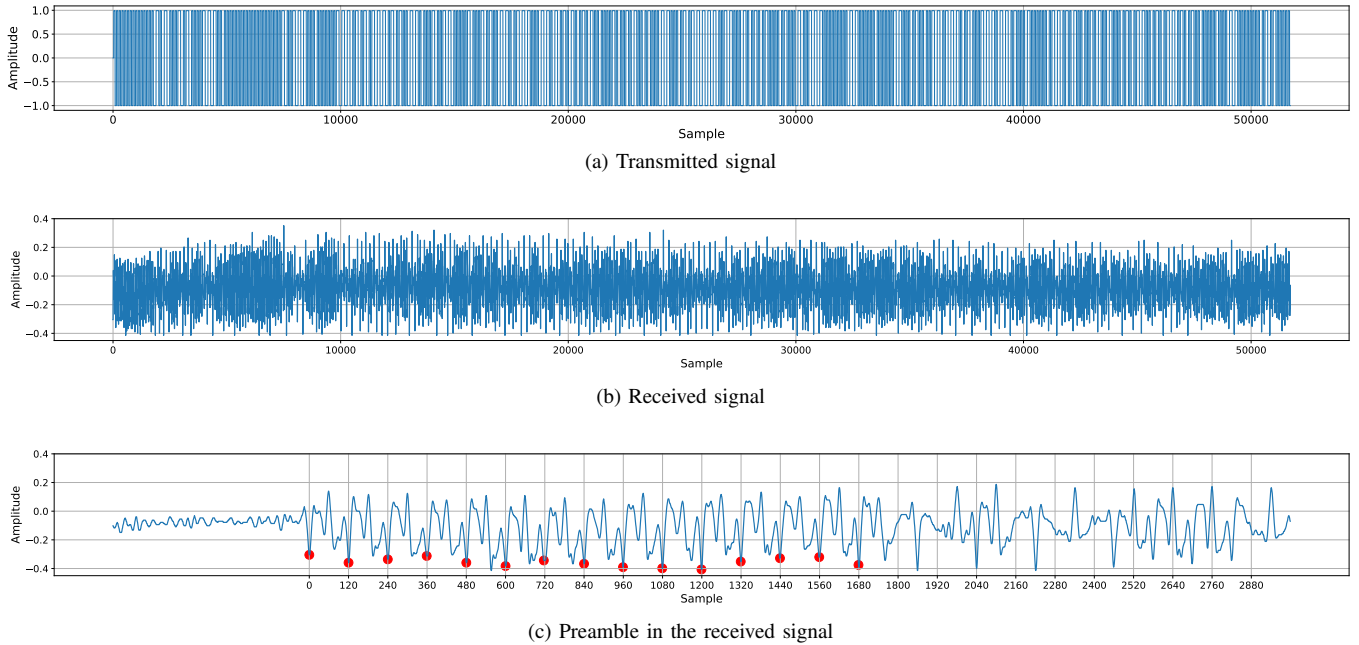


Fig. 5: Transmitted and received signal to and from International Space Station

- 2) via HackRF using NFM modulation, and
- 3) via Yaesu FT-991A using NFM modulation.

The reason behind testing these different transmission methods lies in the fact that HackRF or similar SDRs such as USRP, BladeRF, and Pluto SDR are primarily designed for laboratory experiments, resulting in relatively low output power insufficient to reach satellites. However, our experiment necessitates the signal to reach to satellite, and we identified Yaesu FT-991A as a suitable alternative with controllable output power. Hence, we aimed to ensure consistency in the transmission results across all these methods.

At the receiver end (assuming the satellite), the signal was received by RTL-SDR, processed by SDR sharp, and then the resulting audio was fed to the EPIRB plotter using a virtual audio cable. EPIRB plotter successfully decoded the signal, disregarding the added MAC and timestamp, thus proving backward compatibility. We repeated the test for all three types of beacons and all three types of transmission techniques, and all of them were successful. However, since the EPIRB plotter does not have logic for MAC and timestamp validation, we had to develop a program for validation purposes. We developed a Python program to detect the preamble from the received signal and subsequently retrieve the encoded data from it. In our lab experiment, we found that the message, MAC, and timestamp were successfully received and validated.

### C. Satellite Experiment

The ISS is a LEO satellite that resides approximately 400 kilometers above Earth. However, the point-to-point distance between the ISS and a specific location on Earth varies depending on the satellite's elevation. When the satellite is directly overhead (e.g., at a 90-degree elevation), the distance is shorter. Conversely, when the satellite is near the horizon,

the distance is greater. Since distance significantly impacts RF transmission, we conducted our experiment using both shorter and longer distances. From our location, we utilized a satellite tracker to determine that the shortest distance during our experiment was approximately 407 km, while the longest distance was approximately 2,200 km. The ISS cross-band repeater receives a radio signal at 145.990 MHz and transmits it back at 437.800 MHz. We used different antennas and radios for transmitting and receiving (see Figure 3a). In both cases, our software suite corrected the Doppler shift automatically. A Yaesu FT-991A was used to transmit the signal using 25 watts of power to the ISS, while the signal from the ISS was received by an RTL-SDR using SDR Sharp software. Figure 3b shows SDR Sharp receiving the signal from the ISS. During the test time the ISS was approaching towards our receiver, so there is an apparent increase in the frequency of the received signal (437.806 MHz instead of 437.800 MHz).

To analyze the received signal from space, we recorded the signal in 8-bit audio wave format at a 48 kHz sampling rate, adhering to the standard for .wav file format. The data rate in COSPAS-Sarsat is 400 bits per second. Therefore, a 48 kHz sampling rate would result in  $48000/400 = 120$  samples/bit. Consequently, including MAC and timestamp, the entire signal would require approximately  $431 \times 120 = 51720$  samples. Figure 5 shows the transmitted and received signals. This signal was recorded when the ISS was only 3 degrees above the horizon from our position, approximately 2,200 km away from us (one-way). Ideally, one-way transmission is sufficient for our test. However, since we do not have access to the onboard satellite data, we had to rely on the repeated signal. So, the signal was exposed to the radio channel for twice the time and distance. As a result, a lot of noise is observable in the received signal. However, if we zoom into the beginning part of the received signal in Figure 5c, it can be seen that after some static noise, approximately every 120 samples exhibit

two levels (high and low) with a transition in the middle, which is the characteristic of a biphase-L Manchester encoded signal.

Approximately fifteen consecutive high-low patterns (i.e., binary 1s) indicate the presence of the preamble. We developed a Python script to retrieve data from the received signal. Our program successfully extracted the message, timestamp, and MAC from the received signal, which were subsequently sent for validation. A time duration threshold must be set to determine the message's time validity. If the encoded timestamp and the time at validation are within the threshold, the message is considered valid; otherwise, it is rejected. This helps to mitigate replay attacks. For both shorter (407 km) and greater (2,200 km) point-to-point distances, we were able to retrieve all information from the received signal, and the signal was successfully validated for all three beacon types, proving the feasibility of our concept.

## VII. DISCUSSION

A robust security mechanism is essential for any satellite system. Adequate security safeguards data integrity, prevents unauthorized access, thwarts signal jamming and spoofing, protects against cyberattacks, and ensures compliance with regulations. This research highlights the lack of proper security measures in the COSPAS-Sarsat system, making it a potential target for malicious actors. Satellite hacking and misusing is a growing concern, and the motivations for such attacks can range from sabotage and destruction to financial gain, competition and rivalry, state-sponsored activity, and hacktivism. We propose, implement, and test a MAC-based security solution for COSPAS-Sarsat beacons. Unlike broadcast or telecommunication systems, COSPAS-Sarsat is not a high-traffic system. This SAR communication is only necessary in emergency situations, so the risk of traffic collisions is very low. Additionally, the system has a very slow data rate of only 400 bits per second, which provides ample time slots for each bit transmission. This means that transmitters and receivers are not subject to time constraints. Even with our inexpensive test setup, we were able to successfully retrieve accurate data from the noisy signal.

Our proposed method facilitates secure message transmission from the beacon to the satellite. However, due to access restrictions on the ISS's onboard data, we had to capture the repeated signal. The signal effectively traveled twice the intended distance (since one-way communication is sufficient to demonstrate the proposed method's effectiveness), resulting in a round-trip distance of approximately 4400 km. This considerable distance closely resembles the operational range of a LEO satellite system. While the ISS's amateur radio module's uplink frequency (145.990 MHz) differs significantly from the targeted frequency (406.025 MHz), its downlink frequency (437.800 MHz) aligns closely with the targeted frequency. Utilizing this downlink frequency, we successfully received the signal from space. So during signal reception, our test conditions were close to the targeted frequency. Moreover, the downlink transmission power from the ISS is approximately 5 watts, which closely aligns with the COSPAS-Sarsat beacon's power output. In summary, our test setup encompasses nearly all aspects of the COSPAS-Sarsat's operational environment.

The proposed method appends the timestamp and MAC to the message while leaving the main message untouched. Dur-

ing testing, it was found that adding this additional information does not affect the existing system; the EPIRB plotter was able to decode the message successfully while disregarding the timestamp and MAC. Therefore, the proposed method can be used in a backward-compatible manner. New generations of the beacon and satellite system should have a shared key and key updating functionality. This will allow for secure communication to be established, and all previous versions of the beacons will still be able to work properly. After the old generation beacons reach the end of their lifespan, they should be replaced with the secure version.

The proposed method requires a longer transmission time compared to the current system. Currently, the message occupies 360 milliseconds; following our method, this will increase to 1.0775 seconds. The extended transmission duration may increase the probability of message collisions. Repetition of the same signal from the same beacon may also be affected. However, according to the performance requirements [26], there should be a 99.9% probability of detecting at least one valid beacon message within 30 seconds. Within this timeframe, for a single beacon, the secured signal can be repeated approximately 27 times without collision. Thus, it is highly likely that at least one message will be successfully received by the satellite. On the other hand, considering multiple beacon scenarios, within 30 seconds, at an optimal setup, the secure signal from 27 beacons can be accommodated without collision. At this setting, if multiple beacons transmit the signal at the same time, there will be a collision, but it is less probable that 27 beacons experience a distress situation at the same time and place (e.g., within the footprint of the same satellite). Certainly, longer transmission will increase the probability of transmission collision; however, our proposed method is well-fitted within the performance requirements.

Our experiment demonstrates the effectiveness of the proposed method in enhancing the system's security. However, implementing a new strategy may necessitate certain changes on both the beacon and satellite sides, which could sometimes be burdensome. For instance, COSPAS is a member of the global SAR system, and occasionally, members of this system cooperatively share a few transponders of their satellites to extend services or coverage. In such cases, COSPAS does not own the satellite but rather some transponders, making the process of change implementation potentially delayed. Although we tested the system with an amateur radio system (which is far less effective than a dedicated satellite), the proposed method still needs to be tested on dedicated satellites to gain a better understanding of the final outcomes. Second-generation beacons (SGBs) are planned to include new features such as confirmation to the user regarding message reception, cancellation of distress alerts by the user, and displaying elapsed time since activation. In the future, we intend to further explore these new features of SGBs and conduct experiments involving a geostationary satellite.

## VIII. CONCLUSION

This paper investigates the security vulnerabilities of COSPAS-Sarsat beacons, highlighting the absence of an active security mechanism in this SAR system. We first demonstrate potential attacks on the system and then propose a MAC-based security solution. Our testing was conducted in a real-satellite

environment, encompassing various realistic scenarios. We believe our study will enhance the understanding of COSPAS-Sarsat uplink security and contribute to the implementation of a robust solution. In summary, the proposed solution represents a significant step towards improving the security of COSPAS-Sarsat beacon uplink communication, making it more resilient to cyberattacks and ensuring the timely and accurate delivery of distress signals to SAR authorities.

#### ACKNOWLEDGEMENTS

This work was supported by the Center for Cyber Security at New York University Abu Dhabi (NYUAD).

#### REFERENCES

- [1] Cospas-Sarsat, "Participants," <https://www.cospas-sarsat.int/en/about-us/participants>, accessed on November 28, 2023.
- [2] A. Costin, S. Khandker, H. Turtiainen, and T. Hämäläinen, "Cybersecurity of cospas-sarsat and epirb: threat and attacker models, exploits, future research," in *1st Workshop on the Security of Space and Satellite Systems*, 2023.
- [3] H. Jingli. The birdman: Hacking cospas-sarsat satellites. <https://www.youtube.com/watch?v=uVY916u96BY>. Accessed on November 10, 2023.
- [4] Y. Lv, Q. Ding, X. Liu, J. Zhang, and H. Yang, "Interference analysis of the public walkie-talkie on the Cospas-Sarsat system's uplink," in *15th IEEE International Conference on Signal Processing (ICSP)*, vol. 1, 2020, pp. 675–678.
- [5] *Specification for COSPAS-SARSAT 406 MHz distress beacons*, COSPAS-SARSAT, 3 2021, issue 4 – Revision 7.
- [6] M. Strohmeier, M. Schäfer, V. Lenders, and I. Martinovic, "Realities and challenges of nextgen air traffic management: the case of ADS-B," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 111–118, 2014.
- [7] M. Balduzzi, A. Pasta, and K. Wilhoit, "A Security Evaluation of AIS Automated Identification System," in *Proceedings of the 30th Annual Computer Security Applications Conference*, ser. ACSAC '14. New York, USA: ACM, 2014, p. 436–445.
- [8] D. P. Wright and E. A. Ball, "Highly portable, low-cost SDR instrument for RF propagation studies," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 8, pp. 5446–5457, 2019.
- [9] V. Petkauskas, "We breached Russian satellite network, say pro-Ukraine partisans," <https://cybernews.com/cyber-war/we-breached-russian-satellite-network-say-pro-ukraine-partisans>, accessed on October 10, 2023.
- [10] F. Bussoletti, "Ukraine, Team OneFist brings cyber warfare against Russia into Space," <https://www.difesaesicurezza.com/en/cyber-en/ukraine-team-onefist-brings-cyber-warfare-against-russia-into-space/>, accessed on September 26, 2023.
- [11] J. Menn. Cyberattack knocks out satellite communication for Russian military. <https://www.washingtonpost.com/technology/2023/06/30/satellite-hacked-russian-military>. Accessed on November 15, 2023.
- [12] E. Nakashima. Russian hacker group exploits satellites to steal data, hide tracks. [https://www.washingtonpost.com/world/national-security/russian-hacker-group-exploits-satellites-to-steal-data-hide-tracks/2015/09/08/c59fa7cc-5657-11e5-b8c9-944725fcd3b9\\_story.html](https://www.washingtonpost.com/world/national-security/russian-hacker-group-exploits-satellites-to-steal-data-hide-tracks/2015/09/08/c59fa7cc-5657-11e5-b8c9-944725fcd3b9_story.html). Accessed on September 25, 2023.
- [13] P. Paganini. Hacking the Iridium network could be very easy. <https://securityaffairs.co/wordpress/39510/hacking/hacking-iridium-network.html>. Accessed on October 21, 2023.
- [14] S. P. Arteaga, L. A. M. Hernández, G. S. Pérez, A. L. S. Orozco, and L. J. G. Villalba, "Analysis of the GPS spoofing vulnerability in the drone 3DR solo," *IEEE Access*, vol. 7, pp. 51 782–51 789, 2019.
- [15] L. H. Newman. Hackers are building an army of cheap satellite trackers. <https://www.wired.com/story/nyansat-open-source-satellite-tracker>. Accessed on September 19, 2023.
- [16] E. Salkield, M. Szakály, J. Smailes, S. Köhler, S. Birnbach, M. Strohmeier, and I. Martinovic, "Satellite spoofing from a to z: On the requirements of satellite downlink overshadowing attacks," in *Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. New York, NY, USA: Association for Computing Machinery, 2023, p. 341–352.
- [17] P. Yue, J. An, J. Zhang, J. Ye, G. Pan, S. Wang, P. Xiao, and L. Hanzo, "Low earth orbit satellite security and reliability: Issues, solutions, and the road ahead," *IEEE Communications Surveys and Tutorials*, vol. 25, no. 3, pp. 1604–1652, 2023.
- [18] O. Kodheli, E. Lagunas, N. Maturo, S. K. Sharma, B. Shankar, J. F. M. Montoya, J. C. M. Duncan, D. Spano, S. Chatzinotas, S. Kisseleff, J. Querol, L. Lei, T. X. Vu, and G. Goussetis, "Satellite communications in the new space era: A survey and future challenges," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 70–109, 2021.
- [19] K. Bernsmed, G. Bour, P. H. Meland, R. B. Borgeonkar, and E. Wille, "D4. 3 Multi-modal communication-Securing future communication across different sectors and technologies," *SINTEF AS (ISBN starter med 978-82-14-)*, 2021.
- [20] T. Mladenov, D. Evans, and V. Zelenevskiy, "Implementation of a gnu radio-based search and rescue receiver on esa's ops-sat space lab," *IEEE Aerospace and Electronic Systems Magazine*, vol. 37, no. 5, pp. 4–12, 2022.
- [21] EPIRB-Plotter – decodes, logs and plots transmissions from digital EPIRBs operating in the 406 MHz bands. <https://www.coaa.co.uk/epirbplotter.htm>. COAA. Accessed on November 7, 2023.
- [22] GNU Radio Cmpanion. <https://www.gnuradio.org>. GNU radio companion. Accessed on November 7, 2023.
- [23] SDR Sharp. <https://airspy.com/download>. Air Spy. Accessed on November 12, 2023.
- [24] Amateur Satellite. <https://www.amsat.org>. The Radio Amateur Satellite Corporation. Accessed on November 15, 2023.
- [25] A. Walls, "Tutorial on BPSK bursts," <https://lists.gnu.org/archive/html/discuss-gnuradio/2016-03/msg00208.html>, accessed on September 10, 2023.
- [26] *Operational requirements for COSPAS-SARSAT second-generation 406-MHz beacons*, COSPAS-SARSAT, 10 2014, issue 1 – Revision 3.