Wireless Signal Injection Attacks on VSAT Satellite Modems

Robin Bisping, *ETH Zurich;* Johannes Willbold, *Ruhr University Bochum;*
Martin Strohmeier and Vincent Lenders, *Cyber-Defence Campus, armasuisse*

https://www.usenix.org/conference/usenixsecurity24/presentation/bisping

This paper is included in the Proceedings of the
33rd USENIX Security Symposium.

August 14–16, 2024 • Philadelphia, PA, USA

978-1-939133-44-1

Open access to the Proceedings of the
33rd USENIX Security Symposium
is sponsored by USENIX.

Wireless Signal Injection Attacks on VSAT Satellite Modems

Robin Bisping
ETH Zurich

Johannes Willbold
Ruhr University Bochum

Martin Strohmeier
Cyber-Defence Campus, armasuisse

Vincent Lenders
Cyber-Defence Campus, armasuisse

Abstract

This work considers the threat model of wireless signal injection attacks on Very Small Aperture Terminals (VSAT) satellite modems. In particular, we investigate the feasibility to inject malicious wireless signals from a transmitter on the ground in order to compromise and manipulate the control of close-by satellite terminals. Based on a case study with a widely used commercial modem device, we find that VSATs are not designed to withstand simple signal injection attacks. The modems assume that any received signal comes from a legitimate satellite. We show that an attacker equipped with a low-cost software-defined radio (SDR) can inject arbitrary IP traffic into the internal network of the terminal. We explore different attacks that aim to deny service, manipulate the modem's firmware, or gain a remote admin shell. Further, we quantify their probability of success depending on the wireless channel conditions and the placement of the attacker versus the angle of arrival of the signal at the antenna dish of the receiver.

1 Introduction

VSAT satellite communication systems are used for two-way data transfer, including internet connectivity, voice communication, and video transmission. They are commonly deployed to remote areas, maritime environments, crisis regions, and other locations where terrestrial communication infrastructure is limited or unavailable.

Despite the vital importance of VSAT communication for connecting critical infrastructure, it has not received the same attention in security research as other wireless communication systems. Historically, high equipment costs of satellite communication have acted as a barrier to entry for both researchers and attackers. Recently, however, researchers have demonstrated the feasibility to eavesdrop on such communication with less than $300 of widely-available television equipment [2, 24]. Furthermore, the security vulnerabilities of this infrastructure have not gone unnoticed by cyber threat actors. In February 2022, at the beginning of the Russian attack on Ukraine, a major operator of satellite communication services, Viasat, experienced a disruption that resulted in a significant portion of its network becoming inoperable across Europe. According to the company's official statement [39], the attacker used a misconfigured VPN to gain access to the central hub. From there, they issued legitimate commands to the remote endpoints and rendered thousands of them inoperable. SentinelLabs identified a malicious binary called AcidRain that was uploaded to the affected modems and wiped the file system [14]. Recently, in June 2023, another attack on the satellite communication system of the Russian company Dozor-Teleport was reported [27]. This event coincided with the uprising of the mercenary group Wagner in Russia and resulted in the failure of their system. However, little is known about how the attackers proceeded in this case and in general the security dimensions of satellite Internet services remain not well understood. What is becoming clear, though, is the strategic importance such systems have gained in this day and age.

In this work, we aim to better understand the threat of wireless signal injection attacks on VSAT communication. In particular, the main research question we tackle in this paper is how an attacker may compromise the operation of VSAT modems by injecting malicious signals from a close-by transmitter. Signal injection attacks have been considered in various wireless communication systems including the Global Positioning System (GPS) [11, 15, 33, 34], aviation [37, 38] or phone networks [10, 40], but VSAT communications are quite different to these systems in terms of antenna and modem software stack, and the results thus do not generalize to this type of communication. Salkield et al. [29] provide a theoretical, application-independent security analysis of overshadowing attacks on satellite communication, but their analysis remains theoretical in nature and does not address the impact that the signal injection attacks may have on real antennas, modem's software and networking stack.

With the emergence of software-defined radios and readily available, low-cost radio equipment, enabling an attacker to generate and emit arbitrary electromagnetic signals, signal injection attacks have become more viable. Of particular interest to our study is how modems implement conventional defenses like firewalls or authentication and how an attacker on the ground may successfully inject a spoofed signal in a receiver given the high directionality and gain of VSAT satellite antenna dishes, and their orientation towards the sky.

This work makes the following contributions to the current state of research on satellite communication system security:

- We conduct a comprehensive analysis of a VSAT satellite modem, exploring its communication functionality, hardware architecture, and firmware components.
- We introduce a SDR-based attacker implementation capable of executing arbitrary spoofing attacks.
- We demonstrate proof-of-concept signal injection attacks against a satellite modem. We show how to disrupt the modem's operation and gain privileged access.
- We provide an empirical analysis of the conditions that facilitate the successful execution of such attacks.

2 Background

This section introduces several concepts underlying this work. On the one hand, we describe the topology of satellite communication systems and VSATs. On the other hand, we also present the protocol stack used in the evaluated satellite communication system.

2.1 Satellite Communication Systems

Satellite communication systems transmit and receive signals for various types of communication, such as television broadcasts, phone calls, and internet connectivity. On a high level, they consist of three components:

Central Hub: The central hubs or ground stations which are located on Earth serve as communication hubs. They are equipped with large dish antennas that transmit and receive signals to and from the satellites. Central hubs act as a relay between the satellites and terrestrial communication systems, such as the internet, or between several VSAT modems in a mesh network.

Satellite: Satellites are positioned either in geostationary, medium Earth, or low Earth orbit. Geostationary (GEO) satellites remain fixed with respect to a specific location on Earth, while medium (MEO) and low (LEO) Earth orbit satellites move along orbits closer to the Earth's surface. Most satellites only forward the signals from the central hubs to the endpoint or vice versa. They can be compared to a bent pipe, as they do not perform any data processing, routing, or authentication.

Endpoint: The remotely deployed endpoints communicate with the satellites. The received data at the endpoints is then delivered to the intended recipient, which could be a television, a phone, a computer, or any other device capable of decoding and utilizing the information. Depending on the type of satellite, endpoints may also be able to transmit data back to the central hub via the satellite.

These three components communicate with each other over radio signals. The signals are usually transmitted within the Ku band between 12 GHz and 18 GHz as well as within the Ka band between 27 GHz to 40 GHz. Depending on the system, the communication is separated into two channels: a *forward channel* and a *return channel*. The forward channel is the transmission link from a central hub to an endpoint. The return channel is the transmission link from an endpoint to the central hub. Typically, only the central hub transmits on the forward channel, so channel access method is a consideration only on the return channel. Non-interactive systems, such as television broadcasts, have only a forward channel.

Each channel in turn consists of an *uplink* and a *downlink*: The uplink is the communication path from the central hub or user terminal back to the satellite. The downlink is the reverse communication path, namely from the satellite to the central hub or endpoint. These concepts are illustrated in Figure 1. Once the satellite receives the uplink signal, it amplifies and retransmits the signal on another frequency back to Earth using a transponder. It is important to note that this is a simplified explanation of satellite communication systems. There are variations in the technology and protocols used depending on the specific application and satellite network architecture. Additionally, satellite systems may employ multiple satellites in a constellation or employ complex beamforming and frequency reuse techniques to optimize coverage and capacity.

2.2 Very Small Aperture Terminals

VSAT systems are used for two-way data transfer, including internet connectivity, voice communication, and video transmission. They are commonly deployed to remote areas, maritime environments, crisis regions, and other locations where terrestrial communication infrastructure is limited or unavailable. The topology is identical to a generic satellite communication system shown in Figure 1. The endpoints consist of an *outdoor unit* (ODU) and an *indoor unit* (IDU).

The ODU comprises a high-gain dish antenna, typically ranging from 0.75 m to 1 m in diameter that has a directionality, and a transceiver. The dish is used to transmit and receive signals to and from a geostationary satellite. It is installed outdoors to establish a line-of-sight connection with the satellite. Once the dish points at the satellite, it does not need to be readjusted as the satellite does not move relative to Earth.

The indoor unit serves as the interface between the outdoor unit and the user's network. It typically includes a modem, network interface, and other control and monitoring functions. The IDU allows users to configure and manage the VSAT settings. The modem is used to modulate outgoing data signals

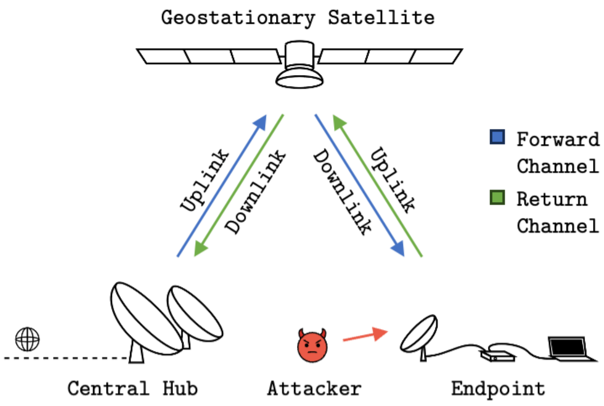


Figure 1: VSAT satellite communication system.

for transmission and demodulate incoming signals for reception. Moreover, it performs functions such as encoding and error correction to ensure accurate and reliable data transfer. The indoor unit is also equipped with a network interface to connect to the user's local network. It may include Ethernet ports, WiFi capabilities, or other networking options to enable connectivity with computers, routers, and other devices within the user's premises.

VSAT operators are typically responsible for maintaining the remotely deployed endpoints. To this end, they maintain a service connection between the central hub and the endpoints, to issue firmware and configuration updates.

2.3 DVB-RCS Protocol Stack

Digital Video Broadcasting - Return Channel via Satellite or short DVB-RCS [18] is a standard for two-way satellite communication systems published by the European Telecommunications Standards Institute (ETSI) in 2000 and at the heart of the wireless injections shown in this paper. It defines a protocol stack for both the forward and return channel that enables communication between a central hub and endpoints. On the forward channel, a signal is broadcast from a central hub to all endpoints. In this way, each endpoint receives the same signal, demodulates and demultiplexes it, but only processes data addressed to it. The return channel, on the other hand, uses Multiple Frequency Time Division Multiple Access (MF-TDMA) to coordinate access to the shared medium and prevent the endpoints from interfering with each other. This requires synchronization among endpoints.

This work focuses solely on the forward channel because injecting a signal into this channel suffices to attack an endpoint. The forward channel protocol layers in DVB-RCS, shown in Figure 2, closely align with layers in the OSI model. Since DVB-RCS supports classical IP-based communication for higher layers, we only describe here the satellite-specific physical and data link layers.

Physical Layer: The physical layer is responsible for transmitting and receiving raw bits over the satellite link. In DVB-RCS, it is specified by the DVB-S2 standard. It supports multiple modulation schemes and adds error correction codes. The standard specifies packetized and continuous data streams.

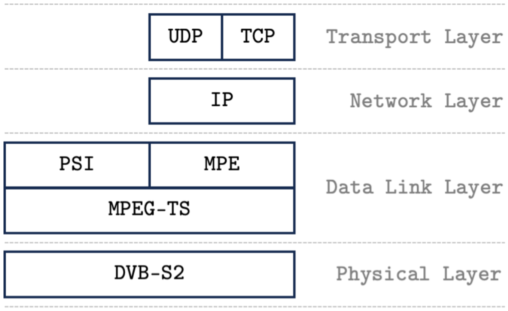


Figure 2: Protocol stack of the DVB-RCS forward channel.

Data Link Layer: In DVB-RCS, the data link layer uses MPEG-TS, which multiplexes multiple data streams intended for different recipients into one transmission. In addition, DVB-RCS supports Multiprotocol Encapsulation (MPE) for encapsulating different types of data packets, such as IP packets. MPEG-TS specifies two types of communication: unaddressed and addressed communication. In unaddressed communication, the data is transmitted to any receiver that receives the signal. In addressed communication, the transmitted data is intended only for reception and processing by a subset of receivers, namely those specified as receivers.

3 Threat Model

The threat model we consider in this work is an attacker that injects wireless signals on the downlink of the forward channel in order to deceive or disrupt the satellite-based communication at the VSAT endpoint. Similar to recent work [29], we define 'signal injection' in this paper as an attacker sending signals from the Earth's surface to the receiver's dish, which otherwise receives its signals only from the intended satellite beam.

This can be considered a significant change from normal operating procedures, whereby the satellite dish and receiver are pointed at a satellite in the sky and receive signals coming exclusively from this direction. In this work, we show that this inherent physical security assumption is not effective in defending against ground-based attackers, also on higher layers of the network stack.

If the legitimate satellite signal is being sent at the same time as the attacker's, this additionally constitutes so-called overshadowing of the legitimate signal by the attacker signal [29].[1]

For example, a spoofing attacker might be able to trick an endpoint into accepting forged configuration updates or com-

[1] For legal and ethics reasons, we did not test this option.

mands. To do this, we assume that the attacker has intimate knowledge of all the protocols in order to forge legitimate communication, and that they have radio equipment to send arbitrary signal waveforms on the frequency of the downlink channel. For example, they may use a software-defined radio such as a USRP or a HackRF together with an upconverter and an antenna to send in the Ku or Ka bands. With this knowledge and equipment, they are able to create arbitrary packets and transmit them to an endpoint.

The attacker must not necessarily be in the direct line between the endpoint's antenna dish and the satellite, but they must have a line-of-sight connection to the antenna of the endpoint and be able to transmit with such power that its signals are stronger than the signals of the legitimate satellite. This is not difficult to achieve since the attacker is on Earth and thus much closer to the endpoint than the satellite in orbit, but high power and close proximity may aid potential detection.

The attacker's goal is to exploit any protocol or software vulnerabilities on the endpoint's side in order to deny service, compromise the integrity of the device, or gain privileged access.

Here, we assume that the attacker knows the modem of a victim (or opportunistically targets any vulnerable device) and can acquire similar hard- and software in order to create a laboratory setup for the initial reverse engineering and vulnerability research similar to what we did in this work. This is a straightforward assumption as satellite modems are generally commercial-off-the-shelf hardware with standard Linux systems (we exclude proprietary military or governmental hardware from our analysis). They can be sourced either directly from a satellite provider/reseller together with a working connection or be bought off second-hand platforms such as eBay.

4 Case Study: Newtec MDM2200

In order to better understand the security of real-world VSAT satellite communication, we analyze the software and hardware architecture of the popular Newtec MDM2200 satellite modem [22] as a typical VSAT endpoint device in the non-governmental/military market. The information in this section is based on knowledge we extracted from publicly available product briefs and technical sheets [22] as well as our own reverse engineering efforts of the modem software's version *2.2.6.19* from *Oct 2014*, which corresponded to the most current version provided to us from the service provider in *March 2021*. The MDM2200 device was introduced by Newtec in 2012 as part of its Dialog modem series. The company was acquired by iDirect in 2019. iDirect as well as Newtec are mainly active in the business-to-business sector. They sell their modems and central hubs to resellers who in turn sell satellite communication services to their customers.

A diverse mix of iDirect systems are in use worldwide. By their own account, in 2022 they held the greatest market share in the maritime, commercial airspace, as well as media and broadcast market [16]. Moreover, they claim to be relied on by 19 NATO member states. The MDM2200 is aimed particularly at consumers, enterprises, and government agencies [22]. Considering the widespread use as well as the criticality of communication systems, it needs to be considered critical infrastructure that require increased protection.

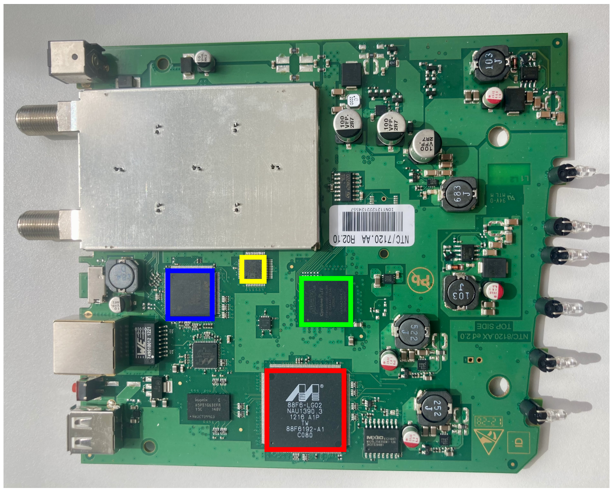


Figure 3: The modem incorporates dedicated hardware components for signal processing, including a microcontroller (red), FPGA (green), demodulator (blue), and DAC (yellow).

Examining the security of this modem not only provides insight into this device, but into an entire product line of one of the largest satellite communication system vendors. The company continues to develop the same product line, possibly drawing on earlier generations and adopting the same vulnerabilities. Therefore, our findings conceptually apply beyond this modem.

In the following section, we rely extensively on the modem control messages and log analysis. This was necessary to reverse engineer the system. However, once the attacker knows how the general system works from their own laboratory setup, it is not necessary to access the logs of a victim setup to perform the attacks against the modem.

4.1 Signal Processing Pipeline

The Newtec MDM2200 IP satellite modem receives and transmits radio signals via two separate coaxial F connectors. These are connected to an ODU consisting of a frequency converter which in turn is connected to a feed horn pointing at the satellite dish. The incoming analog signal is demodulated and decoded by several dedicated hardware components in the modem, shown in Figure 3. These convert the signal into IP packets that are output to the LAN or forwarded to internal processes running on the microcontroller. In detail, this is done as depicted in Figure 4.

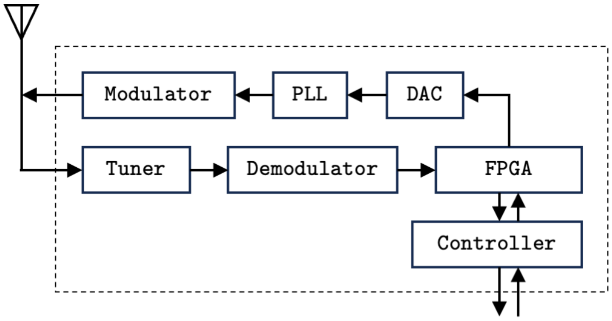


Figure 4: Signal processing pipeline of the Newtec MDM2200 IP satellite modem.

A satellite tuner IC, an STV6111, is used to tune to the correct frequency. It supports an L band frequency input in the range 950 MHz to 2150 MHz. The frequency of the first carrier is configured statically in the web interface, which is usually done by the vendor before sending the modem out to its customers. The parameters of the subsequent carriers are signaled via program-specific information and service information.

The signal is then passed to a demodulator IC, an STV0900A. This chip contains an analog-to-digital converter and supports both the DVB-S and DVB-S2 standards with all their modulation and coding schemes.

The demodulated byte stream is passed to a Field-programmable Gate Array (FPGA), an Altera Cyclone IV EP4CGX30BF14C8N. From reverse engineering the driver, we concluded that the FPGA decodes and demultiplexes the MPEG transport stream. It supports several filters that specify which elementary streams are to be demultiplexed. This allows filtering out all those elementary streams that are not destined for that modem. More specifically, MPEG packets can be filtered by packet identifier and by MAC address. It is also possible to specify whether only the MPEG-TS packet header should be removed or also the transport section header.

Finally, the decoded MPEG-TS section is forwarded to the microcontroller, a Marvell 88F6-LG02, running an embedded Linux operating system. It configures the entire modem with all its components. For this, it parses the received program-specific information (PSI) and service information (SI) sections of the MPEG-TS and applies their changes. The signal pipeline is abstracted behind a network interface: the received IP packets are fed into the IP network stack. Depending on their destination, they are either output to the LAN via the RJ-45 Ethernet port or routed to an internal process.

4.2 Control Software and Networking

A microcontroller acts as a link between the DVB-RCS-based communication and the IP-based communication. It runs an embedded Linux operating system based on BusyBox, which contains user-defined drivers for interaction with the various hardware components. In addition, it listens for configuration updates and applies them accordingly.

The Linux kernel running on the microcontroller contains four kernel modules that configure and communicate with the hardware-based signal processing pipeline: `modem.ko`, `fpga.ko`, `stv0900-demod.ko`, and `s3p.ko`. As the names suggest, the first three initialize the various hardware components, whereas the last one, `s3p.ko`, initializes the overall communication system Sat3Play (S3P), Newtec's proprietary implementation of the DVB-RCS protocol.

Several processes are further running in user space of the operating system. Two of them are of particular interest: `modem_controller` and `swdownload`. `modem_controller` listens for configuration updates on UDP ports 49153 and 49154 as well as on TCP port 49157. It supports multiple configuration message types, both in the form of protocol buffers and in other formats, such as ACM messages. Messages are generally neither authenticated nor encrypted. However, there is optional support for the encryption of some message types. `swdownload` listens for firmware updates on UDP port 49152. These are neither encrypted nor authenticated.

The modem also contains other software that can modify local configurations specific to the satellite communication system, such as `modem_controller_client` and `s3p`. These are CLI tools for interacting with other firmware components, in this case `modem_controller` and `s3p.ko`, respectively. They can be used to apply configuration changes directly from a shell on the modem.

Apart from the PSI and SI, data is transmitted in IP packets. This also applies to configuration updates processed by user space processes. The IP packets are encapsulated within MPE packets, which are addressed with a destination MAC address. For the modem to demultiplex such a stream, the filters on the FPGA must be initialized accordingly. After successful forward channel initialization, the modem sets up three additional filters for receiving elementary data streams: one unicast and two multicast filters. The unicast filter requires that the MPE packets be addressed with the modem's MAC address. On the other hand, the multicast filters necessitate the addressing of packets with one of the standardized multicast MAC addresses. These consist of a multicast IPv4 address with the standardised prefix `01:00:5E`. The 24 bits following this prefix must be in the range from `00:00:00` to `7F:FF:FF` and set to the low 23 bits of the multicast IPv4 address [1, 8].[2] Thus, if the packets are addressed with the MAC addresses `01:00:5e:00:00:01` or `01:00:5e:01:00:01`, they are accepted by all modems receiving the signal.

After being demultiplexed by the FPGA and forwarded to the microcontroller, the encapsulated IP packets are ingested into the network stack and routed according to their destination IP address. For example, if it is set to `192.168.1.1`, they are routed internally within the modem.

[2] See [21] for a detailed explanation.

5 SDR-based Signal Injection Transmitter

In this section, we introduce our SDR-based signal injection transmitter. It was developed with the purpose of demonstrating the practicability and assessing the impact of signal injection attacks on VSAT modems.

5.1 Software and Hardware

For a signal injection attack to work, an attacker needs to replicate authentic central hub transmissions. Signals must be generated in such a way that they deceive the modem into performing demodulation, decoding, and subsequent signal processing. In the case of the MDM2200 modem, this necessitates the development of an S3P-compatible transmitter. Operators of central hubs have specialized equipment for signal generation, including the S3P protocol. However, such devices are both expensive and have limited availability. In addition, dedicated radio equipment typically offers little flexibility, which limits its adaptability to different use cases. To address these cost, availability, and flexibility constraints, we chose to implement our transmitter entirely in software. SDRs provide an ideal platform due to their relatively low cost and ease of reconfiguration.

Our SDR platforms are the Ettus Research USRP B200 and USRP B200mini radios, which cost about $1300 each. We chose them primarily because of their availability and because they meet our frequency requirements: They are capable of transmitting in the range 70 MHz to 6 GHz, which includes the receive frequency range 950 MHz to 2150 MHz supported by the modem. This is beneficial when connecting the SDR directly to the modem via a coaxial cable, which is useful during development and testing of the transmitter. In principle, however, all SDRs that meet these requirements are suitable. For example, our implementation is also compatible with the HackRF One, which is available at a lower price of about $350.

The generation of valid S3P signals is implemented in software on a general purpose computer. In doing so, we focus on a minimal version that fulfills the requirements of initializing the modem and transmitting arbitrary data packets, rather than implementing the entire protocol. To accomplish this, we leverage GNU Radio [28], an open source framework for building SDR systems and signal processing applications.

For our transmitter pipeline, we use a combination of existing and self-developed blocks. The UDP, IP, DVB-RCS, MPE, and MPEG-TS layers are implemented in a new C++ OOT module called `gr-dvbrcs`. This module adds a single source block to GNU Radio, which generates the bitstream for initializing the modem and also sending attack packets. Its architecture is depicted in Figure 5. The bitstream is then modulated by the DVB-S2 layer, which is implemented using existing GNU Radio blocks from the `gr-dtv` module. The resulting IQ samples are forwarded through a sink block to the SDR, from where they are transmitted.

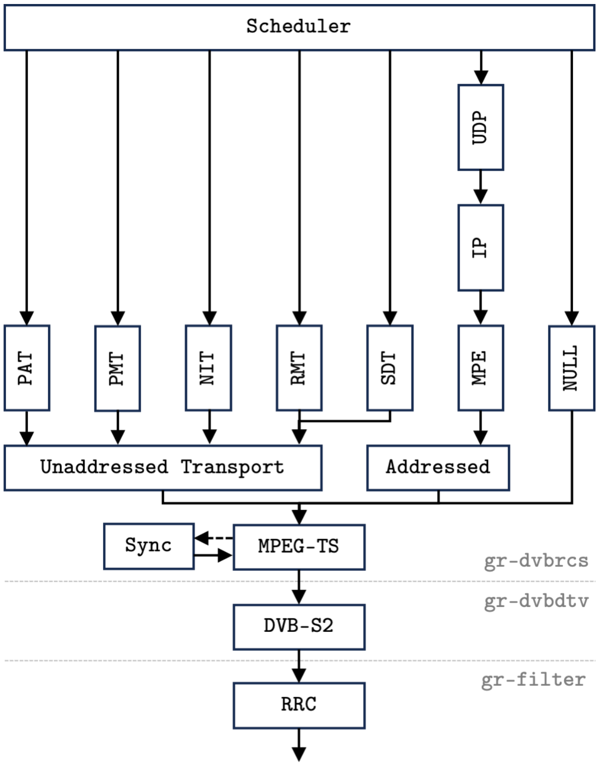


Figure 5: Proof-of-concept transmitter software architecture.

5.2 Forward Channel Establishement

When the modem is started, only the parameters related to the initial carrier are preconfigured, such as its frequency and symbol rate. These settings can be adjusted via the modem's web interface, although they are usually preset by the vendor. We experimentally verified that data transmission on this carrier is not possible. The modem simply ignores it. Therefore, the main challenge of the transmitter is to put the modem in a state where it accepts and processes data packets.

Since S3P is an implementation of the DVB-RCS protocol, the initialization procedure can be derived from the public specification [18]. However, it should be noted that S3P uses a specific configuration of DVB-RCS that does not support all options and subprotocols, and in some cases even deviates from this standard. Through the collection of information from the standard, the reverse engineering of the parsing logic in the kernel modules, and the analysis of log messages, we iteratively reconstructed the required initialization procedure.

5.2.1 Carrier Lookup

To give network operators flexibility in assigning forward channels, especially because the first carrier must be manually configured on the modem, the modem uses a multi-stage carrier lookup process. Consequently, this first carrier is used exclusively to transmit information pointing to the subsequent carrier. The discovery is then performed in several steps: The modem must tune to three different carriers, one after another, on each of which a transport stream is transmitted. After parsing the packets from the first carrier, the receiver learns the parameters to tune into the second carrier, from which it learns about the third carrier. Only on the third and final carrier, it finds the elementary data stream.

This setup presents a challenge for our transmitter. If it does not switch the transmitted carriers at the same time as the modem, the modem tunes to an empty carrier and thus aborts the initialization process. Since we receive no feedback from the modem about such an event, it is impossible for our transmitter to determine the exact time. Consequently, we are forced to transmit on all three carriers simultaneously, which increases the requirements of our attacker.

To bypass this carrier lookup procedure and prevent unnecessary restrictions on our attacker, we leverage the versatility of MPEG transport streams, particularly their ability to accommodate multiple programs and elementary streams. Specifically, we initiate the communication so that each carrier points to itself, albeit with a distinct program number. This is achieved through the proper configuration of the NIT and RMT, communicating the frequency and symbol rate of the initial carrier. This effectively results in a single transport stream containing all the necessary information from the beginning. Consequently, the modem consistently tunes to the same carrier during initialization but demultiplexes a different program from the transport stream each time. It processes only the elementary streams it is expected to receive, disregarding the rest. This approach greatly simplifies the implementation of the transmitter.

5.2.2 Forward Channel Initialization

The DVB-RCS standard defines the syntax of the various PSI and SI sections used for modem initialization, each of which may contain multiple descriptors. However, it often does not specify which descriptors are required for particular sections and their required repetition rates, but only provides a list of compatible descriptors. This may increase flexibility for vendors, but makes it difficult to develop an attacker implementation. As a result, we often had to rely on information provided by the modem and guessing to identify what information it was expecting at what times. In this regard, several resources proved valuable, such as the log errors of firmware components, decompiling `s3p.ko`, and the CLIs `s3p` and `modem_controller_client`, from which the internal modem driver state, including the received section values, could be retrieved.

Our analysis revealed multiple inconsistencies. In two cases, we found that the implementation of the modem's initialization protocol deviates from the standard. In another case, the modem uses a private byte to signal a proprietary feature. In total, the transmitter must support three adjustments, otherwise the forward channel cannot be initialized. Firstly, on the final carrier, the modem expects the Service Description Table (SDT) before the Program Association Table (PAT) and Program Map Table (PMT). As long as it does not receive the SDT, it does not attempt to parse other tables. This order differs from the standard [19]. Secondly, instead of using a Stream Identifier Descriptor within the PMT to identify the data elementary stream configured in the SDT using its component tag, the modem expects an RCS Content Descriptor to identify it by its table ID `0x3E`. The use of this descriptor also differs from the standard [19]. Finally, the private bytes field in the Satellite Return Link Descriptor of the RCS map table [18] is used to configure a proprietary and undocumented transmission mode, also referred to as Mode of Operation (MoO), a designation used by the developers of the modem. A value of `0x00` configures the modem to use ATM/GMSK on the return channel, whereas a value of `0x01` configures it to use GSE/4CPM. Since the Newtec MDM2200 satellite modem only supports GSE/4CPM, the RMT private byte must always be set to `0x01`.

5.3 Data Transmission

After the initialization is completed, the modem is configured to receive and process IP packets. IP packets must be encapsulated within MPE sections, which can be addressed using either unicast or multicast MAC addresses. Besides their increased efficiency for the provider, multicast destination addresses are also more efficient for attacks, as an attacker does not need to enumerate all possible addresses to find the correct one. Instead, the packets are processed by any modem receiving the transmission.

By default, the MDM2200 demultiplexes both unicast and multicast packets. Consequently, it becomes possible to target a modem regardless of whether the attacker knows its MAC address.

In implementing the data transmission logic, we encountered two main challenges. Firstly, we required a method to send payloads that exceeded the maximum length of lower protocol layers, as some of our malicious packets surpassed this limit. Some of the data packets we intended to spoof exceed the fixed MPEG-TS packet length of 188 bytes. To transmit them nonetheless, a mechanism for fragmenting the packets into multiple segments is necessary. DVB-RCS supports fragmentation in various protocol layers to accommodate payloads that surpass the maximum payload length. In our transmitter, we implemented fragmentation exclusively for MPEG-TS packets, as they present the most stringent limitation in terms of packet length. With fragmentation implemented at this layer, the payload can be of any size, provided there are no other constraints imposed by the payload itself. Taking into account the restrictions of the other layers, our implementation is therefore able to transmit arbitrary UDP packets with a length of up to 4073 bytes which is sufficient for our use case.

Secondly, we faced an issue where the modem lost connection after 200 milliseconds due to a lack of synchronization information. The reason for this is that it expects adaptation information containing a program clock reference (PCR) as defined in the MPEG-TS standard [13]. The PCR value must be taken from a 27 MHz clock, which is measured when the fifth byte of the PCR value is transmitted. The maximum allowed jitter is 500 ns. Theoretically, the protocol also supports the transmission of programs without PCR information. However, this does not seem to be supported by the modem, since its clock loses synchronization when signaled that no PCR is present. When it loses synchronization, the modem resets itself to the initial carrier and starts the initialization process over again. Therefore, we implemented synchronization with the goal of performing and evaluating more persistent attacks. Due to the structure of our signal processing pipeline, it is not possible to derive the clock reference from the system clock. In particular, batch processing and recurrent buffers cause irregular delays in the transmission of packets. Therefore, we precompute the PCR based on the number of generated bytes. We count the bytes between two PCRs and derive the value using the byte rate. With this approach, sufficiently accurate PCR values can be generated and transmitted. This suffices to synchronize the clock of the modem.

6 Implemented Attacks

With the signal transmitter from the previous section, we can now perform spoofing attacks and send arbitrary packets wirelessly to any modem in range. This section describes four attacks that target different parts of the modem and vary in terms of goals and sophistication.

It is crucial to reiterate that once the attacker comprehends the fundamental workings of the general system, no access to sensitive information is required for executing attacks against victim modems. The sole prerequisite is knowledge of the parameters of the forward channel.

Scanning the entire RF spectrum is impractical due to its vast size. However, this is unnecessary. Given that the forward channel is a broadcast channel used to transmit data to many modems, commercially acquiring a modem provides the attacker not only with their own parameters but also those of all other modems tuned to the same broadcast channel. Consequently, an attacker is potentially able to target thousands of modems without spectrum scanning at all.

Moreover, the RF spectrum is subject to regulation, with services allocated by the FCC and similar regulatory bodies in other countries. These allocations are publicly accessible. Leveraging open-source intelligence (OSINT), it becomes therefore possible to significantly reduce the search space, rendering spectrum scanning feasible even in the absence of prior information.

6.1 Modem Reset

The first and simplest attack vector aims to reset the modem. For this purpose, we emit a jamming signal consisting of random noise or arbitrary packets at the same frequency as the forward channel. As our objective is to only interfere with the signal rather than overpower it, we establish a jamming-to-signal ratio of $J/S = 1$, where J represents the received jamming signal strength, and S stands for the received legitimate signal power. Because the modem is no longer able to receive the necessary configuration signals to keep the channel active, the modem restarts the channel initialization process, corresponding to an effective reset of the modem. By keeping the jamming signal on, the modem is not able to reinitialize the forward channel. As soon as the jammer is off, the modem reestablishes the link with the central hub again.

6.2 Permanent Channel Disconnect

This second attack aims to permanently disconnect the forward channel from the central hub to the modem. We identified Adaptive Coding and Modulation Messages (ACM) messages as appropriate for this purpose. These messages serve the purpose of signaling the supported modulation and coding schemes (MODCOD), depending on current weather conditions.

By intentionally misconfiguring these values, an attacker can put the modem in a state where it no longer supports the same modulations as communicated by the central hub. This capability can be used to create a discrepancy between the modem and the central hub that affects the availability of the system even after the attack is switched off.

The discovery of this functionality was made through an examination of the modem's data sheet [22]. By eavesdropping on the local communication of an operational modem, we were able to identify the ACM messages. Despite the existence of a standardized ACM protocol [17], our investigation revealed significant differences between the implemented protocol and the established standard. Therefore, we reverse-engineered the parsing logic of the ACM messages to understand their structure and application. Note that this attack is not limited to RCS but effectively applies to the widely-used forward-link of DVB-S2/S2X systems.

We have integrated the generation of such ACM packets into our transmitter. Employing the encapsulation mechanisms described earlier, these UDP packets are encapsulated in IP packets, addressed to all modems, and injected into the MPE elementary stream.

6.3 Malicious Firmware Update

Similar to how network operators provide modems with configuration updates, they also apply firmware updates over the satellite link. On the MDM2200 modem, the update process involves the transmission of an update signalization UDP packet addressed to the internal update management application `swdownload`. Among other information, it contains the version requirements, the size of the update, and instructions on how to get the update. Upon receipt of such a message, the application joins the respective multicast group and starts listening for firmware update blocks on a distinct UDP port, as communicated by the signalization packet. Once the transmission is completed, it checks the CRC checksum of the firmware update and installs it. Crucially, neither the signalization packet nor the update are encrypted or authenticated. Consequently, our attacker transmits a malicious firmware by sending both a signalization packet and a firmware update to the corresponding port. Thus, this represents an attack on the availability, integrity and confidentiality of the system.

6.4 Remote Admin Shell

Finally, the last and most powerful attack vector aims to obtain a remote admin shell in order to execute arbitrary commands on the modem with root privileges. As it is not possible to send shell commands over standard modem features, this requires us to inject a signal that contains an exploit of a vulnerability on the software of the modem. To do so, we rely on a buffer overflow vulnerability in `swdownload` that was previously known and reported but is still unpublished.[3] This vulnerability enables us to execute arbitrary commands with root privileges by sending a malicious UDP packet to the corresponding port.

Specifically, the exploit makes use of a vulnerability in the parsing logic of update signalization messages. There, the `sscanf` function is used to parse and copy a string from the received message into a buffer. Because there is no width specified in the format string of `sscanf`, namely the format string is simply defined as `"%s"`, the function can be misused to overflow the buffer. In this instance, the buffer is allocated with a length of 80 bytes. Thus, by supplying a string longer than 80 bytes, we can overwrite the call stack.

The exploit takes advantage of this vulnerability to alter program execution through return-oriented programming by overwriting the return address of function calls on the stack. The injected sequence of gadgets finally points to the libc function `system`, which executes a shell command passed as an argument. Thus, by transmitting such a crafted update signalization message, an attacker can modify the `swdownload` process to execute an arbitrary shell command. Since `swdownload` is executed with root privileges, the provided shell command is also executed as root.

We inject the exploit by encapsulating the malicious payload within UDP and IP packets, which are then inserted into the MPE elementary stream. These packets are addressed to all modems, which route them internally to the targeted application upon receipt. We establish the reverse shell towards the LAN, although this attack could potentially be extended to connect with any host on the internet, provided that the modem is fully initialized and connected to the internet via satellite.

7 Evaluation

In this section, we evaluate the presented attacks using our SDR-based transmitter. We evaluate not only their impact, but also their feasibility by examining the conditions under which a successful execution is possible depending on the location of the attacker relative to the victim modem.

7.1 Experimental Setup

Our evaluation is based on three configurations: a transmission over coaxial cables, a transmission over the wireless channel on satellite frequencies in a reflection-free anechoic chamber and a transmission over a wireless multipath channel.

The setup over coaxial cables is used to avoid transmitting electromagnetic signals that could potentially interfere with real systems. The modem receives the signals from the SDR directly over a RF cable without any intermediate antennas and up-/downconverters.

The setup over the wireless channel is used to evaluate our attacks under realistic conditions. For this purpose, we configure the VSAT as if it were in regular operation. Ku band signals are transmitted to the satellite dish where they are downconverted by a low-noise block (LNB) before reaching the modem. Consequently, our transmitter needs to be able to generate and transmit signals on the upconverted satellite frequency. Conventional SDRs are usually limited to a frequency range of up to 6 GHz. This is also true for the Ettus Research USRP B200. However, the endpoint can only receive frequencies in the range of 10.7 GHz to 12.75 GHz, a range allocated to direct broadcast satellite services. Therefore, the SDR signal must be upconverted to this range. To do this, we used a block upconverter (BUC), a UMT-TV BUC-Ku002-10.6 v2.0. This BUC upconverts a signal from the L band (950 to 1950 MHz) to the Ku band (11.55 to 12.55 GHz). Specifically, the SDR is connected to a power injector, a UMT-TV DC Injector IDCI with IP Ctrl, which in turn is connected to the BUC. The power injector provides the necessary power to operate the BUC, enabling us to set the transmission power to a value in the range 9.5 dBm to 19 dBm. Finally, the upconverted signal is passed from the BUC to a directional feed horn, a UMT-TV Offset Feed, that emits it. The feed horn has a gain of 13.9 dB.

[3] We note explicitly that this exploit is not specific to this attack vector. Any similar exploit could be used over the wireless interface via the same signal injection method (e.g. CVE-2016-9494 to 9497). As such the exploit is just a means of illustration, not a contribution of this work.

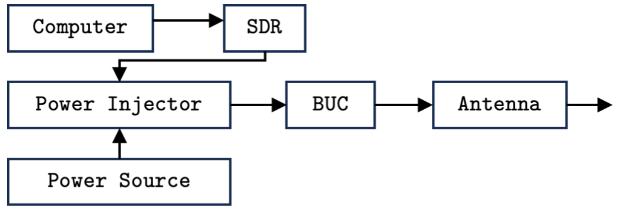


Figure 6: Hardware components for signal transmitter.

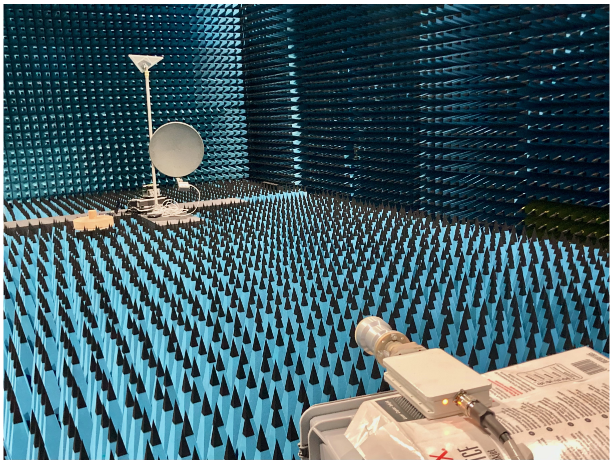


Figure 7: Setup for evaluating signal injection attacks in anechoic chamber.

Overall, the cost of this setup is about $1000, not including the software-defined radios.

On the receiver side, we use the default configuration of the Newtec MDM2200 modem as provided by the vendor. It comprises an iLNB, a Newtec Interactive LNB NTC/2530.AA, and a 75 cm parabolic antenna. The iLNB is connected to the modem and points at the antenna. It downconverts received Ku band signals (10.7 to 12.75 GHz) to the L band (950 to 2150 MHz), a frequency band that is supported by the modem. In addition, it upconverts the transmission signal emitted by the modem to the Ku band (14 to 14.5 GHz). According to the data sheet [22], the gain of the iLNB is 57 to 70 dB.

The experiments over the wireless channel are conducted in two environments: a multipath environment and an anechoic chamber that mitigates multipath reflections. In both environments we set up the receiver and transmitter in a distance of 10 meters with direct line of sight. The multipath environment comprises a corridor that is 3 meters high, 1 meter wide, and 60 meters long, with rooms on both the left and right sides. The transmitter is positioned within the corridor and is oriented towards the transmitter located in one of the room bulges. The walls are constructed of concrete. The anechoic chamber is a rectangular room that is roughly 5 meters high, 10 meters wide, and 15 meters long. The walls, floor, and ceiling are damped with anechoic material.

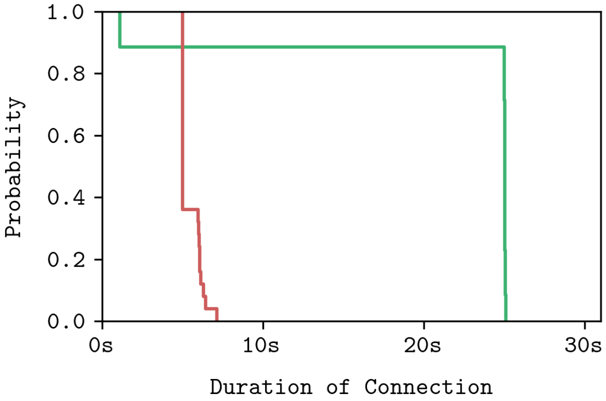


Figure 8: The cumulative distribution of connection duration under jamming (red) and under normal conditions (green).

7.2 Impact of Attacks

7.2.1 Modem Reset

First, we evaluate the effects of interference attacks on the modem, in particular the feasibility of the modem reset attack. For this purpose, two SDRs are connected to the modem via coaxial cable with a T-shaped adapter. One SDR is configured to emulate a legitimate central hub, transmitting the forward channel initialization stream. The other emits a jamming signal consisting of random noise at the same frequency. Given that the SDRs are not calibrated, we adjusted the output to ensure that both emitted signals were received with equal strength at the modem, specifically at -53.4 dBm.

We conduct a comparative analysis between the scenario in which the jammer is present and without the jammer. Specifically, we compare the connection duration, defined as the time interval from when the modem confirmed acquisition of the final carrier signal and commenced synchronization, to when the connection was subsequently lost. The results of the two scenarios are shown in Figure 8. In the absence of a jammer, it can be seen that most connections persisted for about 25 seconds. This corresponds to the time when the modem drops the connection because it does not receive the necessary configuration sections to initialize the return channel. This is intentional, since our transmitter does not send this data. In the scenario with the jammer active, the connections have a much shorter duration, about 5 seconds. The exact reason for the connection losses occurring only after 5 seconds is not entirely clear but most likely related to the time the modem needs to report a loss of connection in the logs. Nevertheless, this evaluation clearly shows the effectiveness of jamming signals to disrupt a connection and reset the modem.

7.2.2 Permanent Channel Disconnect

In evaluating a permanent channel disconnect, we instructed our transmitter to first initialize the modem. We then transmitted the ACM *Client Configuration Message*, *Polling Request Message*, and *PID Mapping Message* either together or individually. We generated these messages with both false and real values.

While we were able to confirm that the spoofed ACM messages are correctly interpreted and applied by examining the modem state, we did not observe any exploitable behavior, such as the ability to change the signal processing configuration, which could be misused to disrupt communication. Despite the modem appropriately configuring the transmitted parameters for switching schemes, and making attempts to adapt coding and modulation schemes based on the received SNR, no other effects were detected.

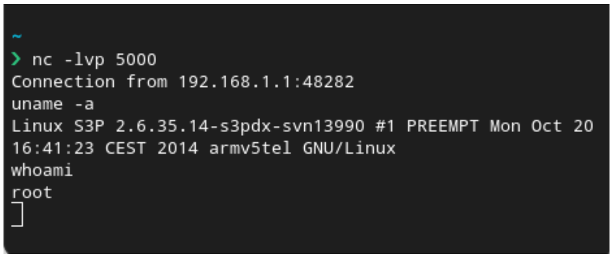
The use of a proprietary ACM protocol by the modem, deviating from the standard, complicates the determination of the attack's failure cause. It is plausible that the demodulator chip is exceptionally robust, capable of demodulating any signal regardless of the configured settings. However, it is also possible that a different encoding is necessary to indicate ACM usage. Unfortunately, this hypothesis could not be tested, as the modem proved incapable of demodulating and decoding the standardized DVB-S2 ACM format.

7.2.3 Malicious Firmware Update

Due to the risk of rendering the modem inoperable by applying a bogus firmware update, we only examined the wireless reception of an update, not its installation. To this end, we sent an update signalization packet to `swdownload` that instructs the modem to listen for update fragments on the specified port if the version is at least 1 and no greater than 9999999. These requirements apply to effectively all modems. We then transmitted an update fragment to the specified port, with an invalid checksum to make sure that the update is not intalled. In our experiments, the modem received it correctly and acknowledged its reception with `SW download completed` while reporting that the checksum was incorrect as intended. This demonstration confirms the feasibility of injecting malicious firmwares of the satellite downlink channel. Since no cryptographic primitives are implemented, the CRC checksum can be easily calculated correctly and trick the modem to apply the firmware update.

7.2.4 Remote Admin Shell

To evaluate the remote admin shell attack, we first initialized the forward channel on the modem. Subsequently, we transmitted the malicious update signalization message containing commands opening a reverse shell to a command and control host. We directly obtained a remote admin shell. The established shell is shown in Figure 9. Thus, this represents an exploitable vulnerability that allows arbitrary commands with root privileges to be executed remotely through wireless transmissions. Given the nature of the vulnerability, which leverages a buffer overflow and subsequently manipulates the call stack, it leaves it in a corrupted state, allowing the attack to be executed only once.



```
~
❯ nc -lvp 5000
Connection from 192.168.1.1:48282
uname -a
Linux S3P 2.6.35.14-s3pdx-svn13990 #1 PREEMPT Mon Oct 20
16:41:23 CEST 2014 armv5tel GNU/Linux
whoami
root
```

Figure 9: Established remote root shell on the modem.

7.3 Evaluation of Channel Conditions

In the following, we evaluate the conditions that facilitate successful signal injection attacks by measuring the received signal characteristics when attacking the complete receiver setup including the low-noise block (LNB) which points at the satellite dish. We performed these experiments at different angles between transmitter and receiver in both the horizontal and vertical planes as it is not clear under which angle a signal injection attack may work. In the horizontal plane, we conducted measurements on one side only, given the antenna's symmetrical construction. The results are thus identical regardless of the side from which the signal is transmitted. Therefore, we mirrored them to the other half accordingly.

In this analysis, we mainly focus on three metrics as they were reported by the modem:

- The *Signal-to-Noise Ratio* (SNR) provides a measure of the signal strength relative to the noise floor.
- In addition, the *Signal Strength* offers an absolute value of the received signal at the specific frequency, encompassing the sum of noise and signal.
- The *Success Rate* quantifies the likelihood that signal injections can be effectively performed from a given angle. In this context, we define a successful signal injection as a transmission where the demodulator successfully locks on and reports a positive signal-to-noise ratio.

Figures 10 and 13 show the results in the anechoic chamber for the signal characteristics of the horizontal and vertical planes, respectively, with a 9.5 dBm signal transmitter. The signal characteristics of the horizontal plane reveal a peak of approx. 40 degrees in width at the center, signifying that this beam is the most suitable for mounting an attack. Additionally, we observe a slight increase in both SNR and signal strength with increasing angles. We attribute this phenomenon to the fact that above a certain angle, the signals can be received directly by the feed horn, bypassing the indirection through the satellite dish. In the vertical plane, we observe comparable outcomes. A peak with an approximate width of 30 degrees, centered around 25 degrees, exhibits optimal reception characteristics. As in the horizontal plane, we observe an elevation in both SNR and signal strength with increasing angle. Beyond an angle of roughly 100 degrees, the transmitter maintains an unobstructed line of sight to the feed horn, facilitating direct transmission before encountering obstruction at higher angles by the satellite dish. Lastly, the success rate shows that signal injection attacks have the highest chance of success across a range of angles, especially when transmitted from frontal positions or directly at the feed horn.

In the multipath environment, we conducted measurements using two distinct transmission powers. Given that the BUC can only be configured to transmit at a minimum of 9.5 dBm, we employed damping materials to attenuate the emitted signal strength. As a result, we conducted experiments utilizing both this lower, damped transmission power and the undamped transmission power of 9.5 dBm. When employing the lower power setting, we observe a pattern similar to that seen in the anechoic environment. The signal characteristics shown in Figures 11 and 14 exhibit a peak when transmitted from frontal positions. Furthermore, we note a marginal increase in both SNR and signal strength with increasing angles. This can be attributed to the phenomenon where, beyond a certain angle, signals are received directly by the feed horn, circumventing the indirect path through the satellite dish. The success rate measurements, depicted in Figure 11c and 14c, support these observations. However, if we look at the measurements performed at a transmission power of 9.5 dBm, we observe a different phenomenon. Instead of individual angles showing favorable reception, we can measure high reception values over the entire range. Remarkably, both the signal characteristics as well as the success rate shown in Figure 12 and 15 exhibit high values for each angle. This result shows that attacks can be carried out successfully from any angle given sufficient transmission power.

8 Discussion

On a first glance, the attacks discussed in this work may appear specific to one device. However, we argue that the underlying problem is systemic and arises from a neglect of security measures in both standards and security lifecycle management. While any specific software vulnerability can be addressed with a software patch, the fact that it is possible to inject arbitrary packets into the network stack, specifically the control plane, exposes a wide attack surface that is hard to control in the present architecture of VSAT systems.

Many traditional and professional users of satellite communication (i.e, military and government) use their own proprietary solutions to secure their systems. While these may work, it is clear that this leaves any other less sophisticated user of the same underlying VSAT architecture vulnerable.

8.1 Impact

The underlying DVB-RCS standard has been developed in the 1990s by, among others, Newtec and the European Space Agency [3]. Importantly, many VSAT modems support and have been using the DVB-RCS standard and are thus fundamentally vulnerable to receiving wireless signal injections. Examples include the entire product line of Newtec that operates on their Sat3Play technology (e.g. NTC2210 and NTC2215). Attacks against these modems are highly likely to succeed without requiring any adaption, given their shared implementation. Some of these modems have subsequently been incorporated into iDirect's brand following the acquisition.

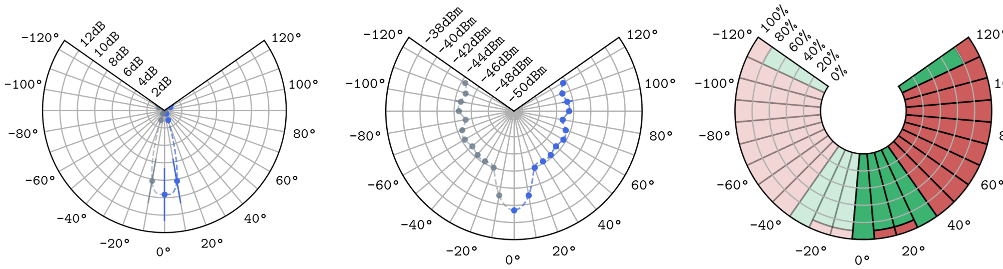
In addition, other vendors have established satellite networks based on DVB-RCS, with prominent examples being Viasat's Linkstar product line (e.g., Linkstar S2 and Linkstar S2A) and Hughes' HN series (e.g., HN7000S and HN9000). Since all devices implement the same standard, we believe that the results of this study can be transferred to these devices with only minimal adjustments. While current numbers on the VSAT market are difficult to find, the Comsys VSAT report [5] lists further standards-based DVB-RCS system vendors (for example Advantech and Satlink). Together with a study by the European Union Agency for Railways [12], both reports put the market share of such solutions around 10-20% out of millions of deployed VSAT terminals.

A designated successor, RCS2, was developed with improved security features [4] but so far has found limited publicly visible adoption. Its concepts have found their way into proprietary solutions used e.g. by providers for government and military and recently the standard has been extended to high capacity beam hopping.

As is known from recent research into vulnerabilities in critical infrastructures and embedded systems, the focus on stability, uptime and verified safety combined with difficulty of access leads to many obsolete software versions with known and unknown vulnerabilities being in use for many years. [6,7] In summary, we believe our approach and our case study transfer to many other satellite modems in current usage with only relatively small modifications, specifically any which do not implement data link layer protection measures, e.g. DVB-RCS or DVB-RCS2 without optional protection or proprietary signaling protocols without link layer security. This attack vector was simply ignored at the time of development of DVB-RCS, likely because it predated the rise of SDRs, and there has been no (public) awareness about it since.

8.2 Mitigations

To mitigate the discussed vulnerabilities, mechanisms to authenticate and encrypt the communication are needed. Such protections can be put in place on multiple layers.

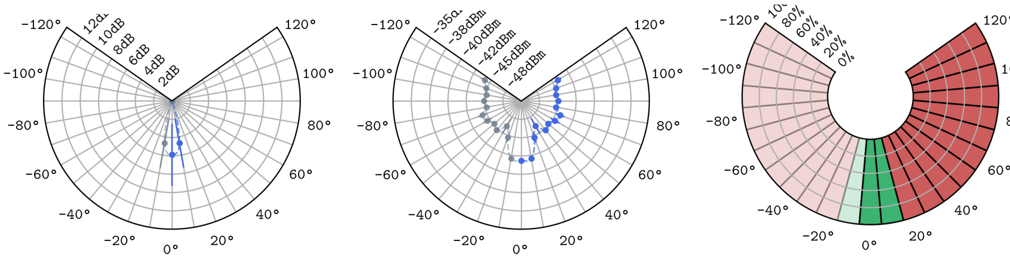


(a) Signal-to-noise ratio.

(b) Signal strength.

(c) Success rate.

Figure 10: Signal injection attack at **9.5 dBm** in the **anechoic environment** from different angles in the **horizontal plane**.

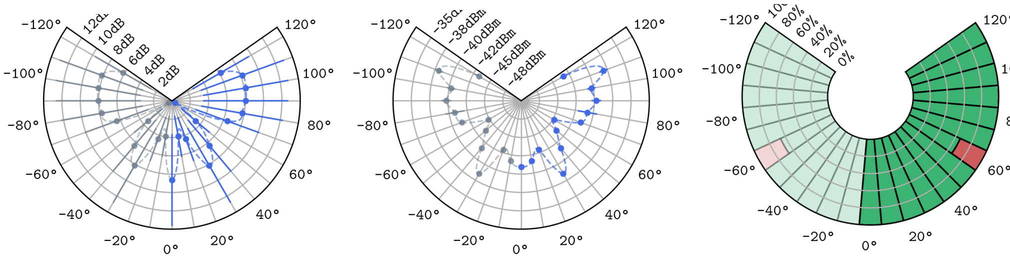


(a) Signal-to-noise ratio.

(b) Signal strength.

(c) Success rate.

Figure 11: Signal injection attack at **low power** in the **multipath environment** from different angles in the **horizontal plane**.

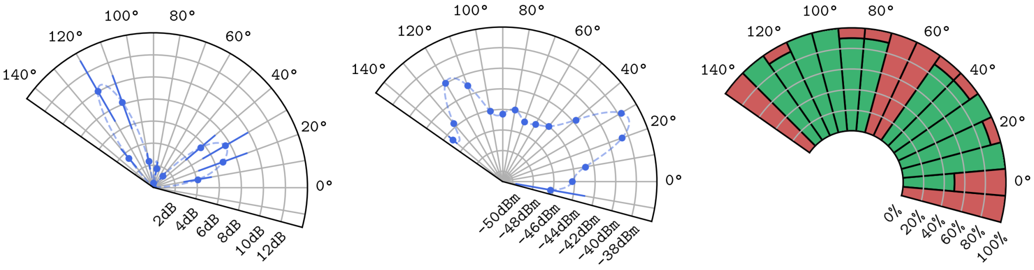


(a) Signal-to-noise ratio.

(b) Signal strength.

(c) Success rate.

Figure 12: Signal injection attack at **9.5 dBm** in the **multipath environment** from different angles in the **horizontal plane**.

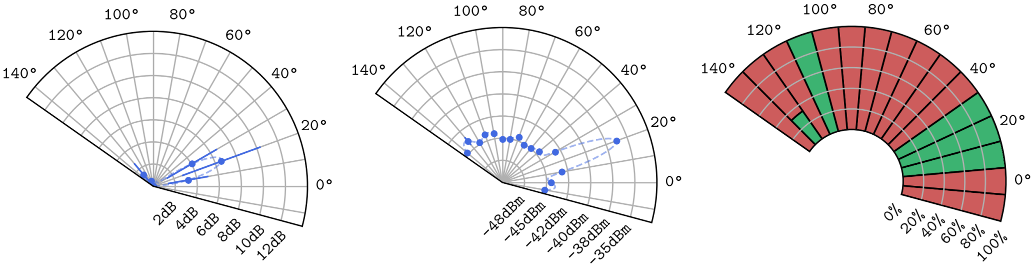


(a) Signal-to-noise ratio.

(b) Signal strength.

(c) Success rate.

Figure 13: Signal injection attack at **9.5 dBm** in the **anechoic environment** from different angles in the **vertical plane**.

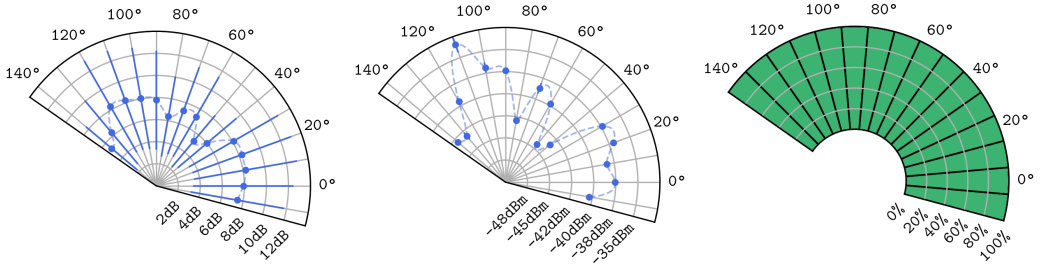


(a) Signal-to-noise ratio.

(b) Signal strength.

(c) Success rate.

Figure 14: Signal injection attack at **low power** in the **multipath environment** from different angles in the **vertical plane**.



(a) Signal-to-noise ratio.

(b) Signal strength.

(c) Success rate.

Figure 15: Signal injection attack at **9.5 dBm** in the **multipath environment** from different angles in the **vertical plane**.

8.2.1 Physical Layer

Several strategies have been proposed to detect spoofed signals at the physical layer. For instance, Jedermann et al. [20] advocate for the utilization of time difference of arrival measurements obtained from multiple receivers to compute the satellite's position and compare it with the satellite's actual position. In cases where these do not align, it can be concluded that the signal is spoofed.

In addition, Smailes et al. [36] advocate for employing fingerprints, which are derived from distinctive signal characteristics exhibited by the transmitter. To implement this, they propose an architecture for transforming satellite signals into unique fingerprints. If these deviate significantly from pre-recorded reference fingerprints, the system categorizes the signal as spoofed and rejects it.

While these approaches provide robust results, their implementation requires additional hardware, either in the form of multiple receivers for multilateration or integrated circuits that provide the necessary signal characteristics for fingerprint computation.

8.2.2 Data Link Layer

The DVB-RCS [18] standard outlines encryption and authentication mechanisms that cryptographically scramble the transmission. However, the adoption of these measures is optional and contingent upon the specific implementation. The data necessary for the modem to discern if a packet is intended for it is retained in unauthenticated plaintext. Furthermore, these protective measures are only activated following the exchange of keying material. Consequently, they do not provide protection against spoofing during modem initialization, unless they keying material has previously been exchanged offline.

8.2.3 Network Layer

To provide supplementary protection, the DVB-RCS standard advocates for the implementation of network layer security mechanisms. For instance, IPsec [9] could be leveraged to encrypt and authenticate the communication between the endpoint and central hub. Consequently, user communication is encrypted and authenticated, regardless of the implementation of lower layers. However, protection measures at this layer do not protect against spoofing attacks that use lower-layer protocols. Attacks that exploit vulnerabilities in the data link layer therefore remain a potential threat against all modems not implementing a proprietary layer 2 encryption authentication scheme.

9 Related Work

Although the security of satellite communication systems has so far received little attention by the research community, there is a small but growing body of work in this domain. Pavur and Martinovic [23] have compiled a comprehensive list of incidents from the past six decades, underscoring the critical importance of satellite security and its vulnerability to potential attacks.

9.1 Security of VSAT Systems

After decades dominated by closed and proprietary satellite systems, the European Telecommunication Standards Institute (ETSI) published the DVB-RCS protocol in 2000, nowadays still an important protocol in VSAT systems. Skinnemoen et al. [35] present a comprehensive study of this protocol, with their main focus on performance evaluation. The lack of security adoption in DVB-RCS is evident in a number of papers dealing with forward channel communication security. For example, Adelsbach and Greveler [2] and Pavur et al. [24] analyzed DVB-S broadcast communication. They found that a significant portion is unencrypted, allowing an attacker with a budget of $300 or less to intercept confidential communication. In following work, Pavur et al. [25] come to a similar conclusion when analyzing DVB-S2 traffic. They find that much of the communication is unencrypted, revealing confidential information, and should be secured with new countermeasures such as secure proxies [26]. Although these papers do not analyze DVB-RCS specifically, their results show that many protocols supported by RCS — and their users — do not adopt any security measures.

While these papers targeted vendor-independent communication, others have analyzed the security of specific hardware implementations. For example, Santamarta [30] performed an analysis of several satellite communication modems. In each of them, he uncovered exploitable vulnerabilities, such as backdoors, hardcoded credentials, and insecure protocols. In subsequent work [31], Santamarta analyzed the impact of accessing such systems in the aerospace, maritime, military and space industries, ranging from attacking onboard communication to disrupting it altogether.

This field of research has gained practical relevance in light of the ViaSat incident [39], the largest publicly known attack on satellite communication systems, which disrupted a significant portion of its network. Following the incident, Guerrero-Saade and Amerongen [14] identified the wiper malware uploaded to the affected modems. Subsequently, Santamarta [32] confirmed these findings and identified the mechanism used to upload the malware to the modems. In contrast, to our work, this attack was performed by infiltrating the central hub, while we study the threat model of a wireless signal injection from a device in the neighborhood of a VSAT receiver.

9.2 RF Signal Injection Attacks

Salkield et al. [29] provide an general security analysis of satellite downlink spoofing. Their study addresses the factors that influence the success of such attacks, including modulation scheme, antenna directionality, and signal strength.

A particular topic that has attracted a lot of attention is the security of Global Navigation Satellite Systems (GNSS). Several studies addressed the characterization and mitigation of GPS jamming [34]. For example, Ferreira et al. [11] assessed the utilization of low-cost SDRs for implementing a jammer capable of generating an effective interfering signal targeting an unmanned aerial vehicle. Horton and Ranganathan [15] introduced a low-cost implementation of a GPS spoofer.

Finally, there are many demonstrated RF attacks on terrestrial systems, particularly within mobile communication. Both Yang et al. [40] and Erni et al. [10] implemented overshadowing attacks against LTE, showing that it is possible to inject arbitrary messages and achieve a denial of service.

To the best of our knowledge, no previous study considered an active wireless attacker model on the communication stack of VSAT endpoints. This study fills this gap by analyzing signal injection attacks on a real-world VSAT system.

10 Conclusions

Satellite communication systems serve as a crucial link often connecting critical infrastructure to the internet. Consequently, many of these satellite systems may qualify as critical infrastructure themselves, a fact not lost on threat actors, as evidenced by recent attacks targeting such networks. However, these systems have received only limited attention from the research community so far.

This study is a first step towards examining satellite endpoint security against signal injection attacks. We focused on the Newtec MDM2200, a modem of a leading VSAT vendor. To assess the feasibility of these novel attacks, we constructed a proof-of-concept transmitter that can wirelessly initialize the modem and transmit arbitrary data packets to it. This enables the remote exploitation of protocol and software vulnerabilities in the modem. To understand the conditions required for successful execution, we evaluated our attacker implementation in both an anechoic chamber and a multipath environment. This enabled us to determine the positions from which signal injection attacks have the highest probability of success. We find that if the transmission power is high enough, an attack can be carried out from any angle.

Signal injection attacks are possible, because the modem supports communication without encryption and authentication, thus neglecting protection against unauthorized eavesdropping, manipulation, or injection of packets in the lower communication layers. This likely stems from a time when security considerations were not as prevalent. However, in the age of software-defined radios, it is imperative to reevaluate these assumptions. Protection against such attacks must be considered a primary systems requirement.

Ethics / Disclosure

All experiments involving malicious signals injections have been performed against our own modem. Signals that could potentially compromise neighboring modems have been transmitted in a shielded environment or over shielded RF cables to avoid collateral damage.

The attack and all vulnerabilities used in this paper have been reported responsibly to the vendor of the modem. iDirect has not reacted to multiple inquiries over one year. After submission of the paper, we have further disclosed the paper and all used vulnerabilities with the Swiss National Cyber Security Center (National CERT). They are working to obtain a response from the vendor and are preparing CVE numbers where applicable. Further discussions are being conducted regarding other manufacturers using potentially unsecured DVB-RCS systems.

Acknowledgments

We thank Knut Eckstein from the European Space Agency for his helpful feedback.

References

[1] Donald E. Eastlake 3rd and Joe Abley. IANA Considerations and IETF Protocol and Documentation Usage for IEEE 802 Parameters. RFC 7042, October 2013.

[2] Andre Adelsbach and Ulrich Greveler. Satellite communication without privacy - attacker's paradise. pages 257–268, 01 2005.

[3] European Space Agency. Newtec signs multi-million Euro ESA contract, 2001. https://www.esa.int/Applications/Connectivity_and_Secure_Communications/Newtec_signs_multi-million_Euro_ESA_contract, visited 2024-02-28.

[4] Mark Bowyer, Lars Erup, and Hans Peter Lexow. Security in DVB-RCS2. *International Journal of Satellite Communications and Networking*, 31(5):263–276, 2013.

[5] Comsys. The comsys VSAT Report - 14th Edition, 2017. https://www.comsys.co.uk/pdfs/p_vr_bro.pdf, visited 2024-02-28.

[6] Andrei Costin, Jonas Zaddach, Aurélien Francillon, and Davide Balzarotti. A {Large-scale} analysis of the security of embedded firmwares. In *23rd USENIX security symposium (USENIX Security 14)*, pages 95–110, 2014.

[7] Andrei Costin, Apostolis Zarras, and Aurélien Francillon. Automated dynamic firmware analysis at scale: a case study on embedded web interfaces. In *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, pages 437–448, 2016.

[8] Dr. Steve E. Deering. Host extensions for IP multicasting. RFC 1112, August 1989.

[9] Naganand Doraswamy and Dan Harkins. *IPSec: The New Security Standard for the Internet, Intranets, and Virtual Private Networks*. Prentice Hall PTR, USA, 1999.

[10] Simon Erni, Martin Kotuliak, Patrick Leu, Marc Roeschlin, and Srdjan Capkun. Adaptover: Adaptive overshadowing attacks in cellular networks. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*, MobiCom '22, page 743–755, New York, NY, USA, 2022. Association for Computing Machinery.

[11] Renato Ferreira, João Gaspar, Pedro Sebastião, and Nuno Souto. Effective gps jamming techniques for uavs using low-cost sdr platforms. *Wireless Personal Communications*, 115(4):2705–2727, Dec 2020.

[12] European Union Agency for Railways. Study on feasibility of satcom for railway communication - final report, 2016. https://www.era.europa.eu/system/files/2022-11/Study%20on%20feasibility%20of%20satcom%20for%20railway%20applications%20by%20INDRA_ALG_0.pdf, visited 2024-02-28.

[13] International Organization for Standardization. Information technology — generic coding of moving pictures and associated audio information — part 1: Systems. Standard ISO/IEC 13818-1:2022, International Organization for Standardization, 2022.

[14] Juan Andres Guerrero-Saade and Max van Amerongen. Acidrain | a modem wiper rains down on europe, 2022. https://www.sentinelone.com/labs/acidrain-a-modem-wiper-rains-down-on-europe/, visited 2024-02-28.

[15] Eric Horton and Prakash Ranganathan. Development of a GPS spoofing apparatus to attack a DJI Matrice 100 Quadcopter. *The Journal of Global Positioning Systems*, 16(1):9, Jul 2018.

[16] ST Engineering iDirect. Shaping the future of how the world connects through satellite communications, 2022. https://www.idirect.net/wp-content/uploads/2020/01/2022-Corporate-Factsheet-US.pdf, visited 2024-02-28.

[17] European Telecommunications Standards Institute. Digital Video Broadcasting (DVB); DVB-S2 Adaptive Coding and Modulation for Broadband Hybrid Satellite Dialup Applications. Technical Specification ETSI TS 102 441, European Telecommunications Standards Institute, 2005.

[18] European Telecommunications Standards Institute. Digital video broadcasting (dvb); interaction channel for satellite distribution systems. European Standard ETSI EN 301 790, European Telecommunications Standards Institute, 2009.

[19] European Telecommunications Standards Institute. Digital Video Broadcasting (DVB); DVB specification for data broadcasting. European Standard ETSI EN 301 192, European Telecommunications Standards Institute, 2021.

[20] Eric Jedermann, Martin Strohmeier, Matthias Schäfer, Jens Schmitt, and Vincent Lenders. Orbit-Based Authentication Using TDOA Signatures in Satellite Networks. In *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, WiSec '21, page 175–180, New York, NY, USA, 2021. Association for Computing Machinery.

[21] DQ Networks. Multicast addressing, 2023. https://www.dqnetworks.ie/toolsinfo.d/multicastaddressing.html, visited 2024-02-28.

[22] Newtec. Mdm2200 ip satellite modem, 2013. http://web.archive.org/web/20130414212813/http://newproducts.newtec.eu/frontend/files/products/pdfs/mdm2200.pdf, visited 2024-02-28.

[23] James Pavur and Ivan Martinovic. Building a launchpad for satellite cyber-security research: lessons from 60 years of spaceflight. *Journal of Cybersecurity*, 8(1):tyac008, 06 2022.

[24] James Pavur, Daniel Moser, Vincent Lenders, and Ivan Martinovic. Secrets in the sky: On privacy and infrastructure security in dvb-s satellite broadband. In *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks*, WiSec '19, page 277–284, New York, NY, USA, 2019. Association for Computing Machinery.

[25] James Pavur, Daniel Moser, Martin Strohmeier, Vincent Lenders, and Ivan Martinovic. A Tale of Sea and Sky On the Security of Maritime VSAT Communications. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 1384–1400, 2020.

[26] James Pavur, Martin Strohmeier, Vincent Lenders, and Ivan Martinovic. Qpep: An actionable approach to secure and performant broadband from geostationary orbit. In *The Network and Distributed System Security Symposium (NDSS)*, Feb 2021.

[27] The Washington Post. Cyberattack knocks out satellite communications for russian military, 2023. https://www.washingtonpost.com/technology/2023/06/30/satellite-hacked-russian-military/, visited 2024-02-28.

[28] GNU Radio Project. Gnu radio (version 3.10.6.0), 2023. https://www.gnuradio.org/, visited 2024-02-28.

[29] Edd Salkield, Marcell Szakály, Joshua Smailes, Sebastian Köhler, Simon Birnbach, Martin Strohmeier, and Ivan 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*, WiSec '23, page 341–352, New York, NY, USA, 2023. Association for Computing Machinery.

[30] Ruben Santamarta. Satcom terminals: Hacking by air, sea, and land. Technical report, 2014.

[31] Ruben Santamarta. Last call for satcom security. Technical report, 2018.

[32] Ruben Santamarta. Viasat incident: from speculation to technical details, 2022. https://www.reversemode.com/2022/03/viasat-incident-from-speculation-to.html, visited 2024-02-28.

[33] Harshad Sathaye, Martin Strohmeier, Vincent Lenders, and Aanjhan Ranganathan. An Experimental Study of GPS Spoofing and Takeover Attacks on UAVs. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 3503–3520, 2022.

[34] Desmond Schmidt, Kenneth Radke, Seyit Camtepe, Ernest Foo, and Michał Ren. A survey and analysis of the gnss spoofing threat and countermeasures. *ACM Computing Surveys*, 48:1–31, 05 2016.

[35] H. Skinnemoen, R. Leirvik, J. Hetland, H. Fanebust, and V. Paxal. Interactive ip-network via satellite dvb-rcs. *IEEE Journal on Selected Areas in Communications*, 22(3):508–517, 2004.

[36] Joshua Smailes, Sebastian Köhler, Simon Birnbach, Martin Strohmeier, and Ivan Martinovic. Watch this space: Securing satellite communication through resilient transmitter fingerprinting. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security (CCS '23)*, New York, NY, USA, 2023. ACM.

[37] Martin Strohmeier, Ivan Martinovic, and Vincent Lenders. Securing the air–ground link in aviation. *The Security of Critical Infrastructures: Risk, Resilience and Defense*, pages 131–154, 2020.

[38] Martin Strohmeier, Daniel Moser, Matthias Schafer, Vincent Lenders, and Ivan Martinovic. On the applicability of satellite-based air traffic control communication for security. *IEEE Communications Magazine*, 57(9):79–85, 2019.

[39] Viasat. Ka-sat network cyber attack overview, 2022. https://news.viasat.com/blog/corporate/ka-sat-network-cyber-attack-overview, visited 2024-02-28.

[40] Hojoon Yang, Sangwook Bae, Mincheol Son, Hongil Kim, Song Min Kim, and Yongdae Kim. Hiding in plain signal: Physical signal overshadowing attack on LTE. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 55–72, Santa Clara, CA, August 2019. USENIX Association.