Satellite Cybersecurity Reconnaissance: Strategies and their Real-world Evaluation

Johannes Willbold
Chair for Systems Security
Ruhr University Bochum
johannes.willbold@rub.de

Franklyn Sciberras
Department of Computer Science
ETH Zürich
fsciberras@ethz.ch

Martin Strohmeier
Cyber-Defence Campus
armasuisse Science + Technology
martin.strohmeier@armasuisse.ch

Vincent Lenders
Cyber-Defence Campus
armasuisse Science + Technology
vincent.lenders@armasuisse.ch

***Abstract*—The security of satellite and space systems has become a pressing concern in recent years as various high-profile incidents involving satellite-based internet access have been observed in the context of the war in Ukraine. An increase in threat level is partly due to a) rapid advancements in affordable software-defined communications equipment, which have made it easier for attackers to gain communication capabilities with orbital assets and b) the increasing adoption of commercial off-the-shelf hardware and software components in spacecraft enhancing affordability and exposing potential vulnerabilities more easily. A recent study revealed that satellite software typically lacks sufficient protection against unauthorized access. However, it remains unclear how this inherent lack of security is critical to other satellites in orbit as no public work exists on cybersecurity reconnaissance. Hence, to date, identifying non-standard commands and assessing potential vulnerabilities are deemed challenging steps for attackers to perform.**

In light of this current state, this paper analyzes how attackers may conduct reconnaissance on satellites' capabilities without targeting the ground segment. We develop strategies that attackers may employ to evaluate satellites' capabilities using a satellite implementation that adheres to the ECSS-standardized Telecommand on top of a CCSDS protocol stack. Our considered strategies encompass enumeration methods to identify the subset of the standard implemented, including non-standardized functionalities. Additionally, we present strategies to analyze implementation-specific aspects of CCSDS' Space Data Link Security (SDLS) Protocol, which serves as the primary security protocol within the CCSDS protocol family. Our strategies test the potential for timing side channels, fault-message-based enumeration, and payload length enumeration testing. To evaluate the effectiveness of our strategies, we apply them to a real-world satellite and measure their success rate.

TABLE OF CONTENTS

979-8-3503-0462-6/24/$31.00 ©2024 IEEE

1. INTRODUCTION

The advent of novel mega-constellations in Low Earth Orbit (LEO), exemplified by Starlink, OneWeb, and others, is set to introduce an unprecedented number of satellites into space, potentially exceeding 100,000 in the coming years. These constellations will serve crucial communication functions, ranging from global navigation and positioning systems to consumer-oriented phone connections and imaging data services, playing an increasingly vital role in modern society's infrastructure. However, this growing reliance on satellite systems also attracts the attention of cyber attackers, given their critical position in communication and navigation networks. In the Ukraine war, incidents such as the major attack on the ViaSat network [1], [2], during the early stages and the reported Dozor-Teleport hack in 2023 [3] have received global attention. Ongoing disruptions of Global Navigation Satellite Systems (GNSS) worldwide illustrate the vulnerabilities and consequences of satellite cyber attacks [4].

Despite public awareness of satellite system vulnerabilities dating back to the mid-2000s, recent evidence suggests that cybersecurity remains severely behind in legacy and novel deployments. Recently, discussions and publications at academic and hacker conferences raised alarms regarding these vulnerabilities, which persist on a much larger scale today. The computer security communities have developed valuable insights that have not yet been applied to space systems. For instance, work by Pavur et al. [5] and Willbold et al. [6] on satellite communication vulnerabilities highlights the need for further exploration and mitigation of risks in satellite systems. Although the viability of certain attacks on satellite systems and, more specifically, against the space segment of a satellite mission has been shown in previous research, they often assume well-informed attacker models with detailed insights into a system. While this assumption is useful to model powerful attackers, it does not always reflect real-world scenarios well, where attackers must collect information before a successful attack. This phase is usually referred to as the *reconnaissance* phase, the initial phase of the cyberattack lifecycle in the well-known MITRE ATT&CK framework [7]. Threat actors need to gather intelligence about satellite infrastructures, communication protocols, and potential vulnerabilities prior to planning their attack strategies. By understanding the satellite landscape through reconnaissance, attackers can tailor their strategies to exploit specific weaknesses, leading to more effective and targeted attacks.

Unlike traditional network security, satellite reconnaissance requires an understanding of the space environment, including orbital mechanics and satellite communication protocols,

making it a specialized domain within the broader cyber security landscape. With the space industry increasingly adopting commercial off-the-shelf (COTS) hardware and software and moving away from custom designs, the attack surface of satellites becomes more accessible to adversaries. This shift to standardized components may lead to an increase in shared vulnerabilities across multiple satellite systems. Additionally, the advent of software-defined radios allows attackers to reverse engineer satellite protocols and exploit wireless vulnerabilities remotely and stealthily.

Despite the critical importance of satellite systems in many applications, satellite reconnaissance remains largely neglected. The unique challenges, complex architectures, and evolving technologies in the space domain demand focused attention. The lack of academic research and comprehensive studies on satellite reconnaissance hinders the development of hardened satellite systems, effective countermeasures, and threat detection strategies. As the satellite industry expands, the need for robust security practices becomes even more apparent.

Our contributions are three-fold:

- We are the first to enumerate reconnaissance goals that attackers may need to prepare and execute practical cyber attacks against satellite systems.
- We identify strategies to successfully obtain the required reconnaissance information on such systems.
- Finally, we apply and evaluate these strategies against a real satellite system.

The remainder of this paper is organized as follows. Section 2 describes the necessary background on satellite systems before Section 3 discusses practical attacker models. Section 4 enumerates the reconnaissance goals, Section 5 the reconnaissance strategies. Section 6 presents a practical evaluation of the identified strategies. Section 7 discusses our results and Section 8 the related work before Section 9 concludes.

2. BACKGROUND

This section introduces the necessary background on satellite systems to understand the remainder of the paper.

Space Mission Context

Space missions consist of a space segment of one or more satellites, whereby multiple satellites are usually referred to as a *constellation*. The space segment is directed and controlled by the ground segment, consisting of one or multiple Ground Stations (GSs). The satellite operators controlling the satellite use the GS to send command-and-control traffic in the form of telecommands to the space segment. The satellite answers with Telemetry (TM) data. The combination of both is referred to as Telemetry and Telecommand (TMTC) or Tracking Telemetry and Control (TT&C). Tracking information is communicated using the Two-Line Element (TLE), which denotes the parameter necessary to map the current position of a satellite relative to Earth and to predict future orbital positions.

Satellite Radio Background

Spacecraft deploy either a *directional antenna* that uses a directed beam to transfer the radio signal and hence requires a specific point to send it to the right location and a specific physical location to receive it. Contrary to this, *omnidirectional antennas*, as often used for Ultra High Frequency (UHF) or Very High Frequency (VHF), send in all directions and thus can be received from all directions and angles similarly.

Importantly, satellites have antenna beams that differ in size by orders of magnitude depending on the constellation and orbit. A typical GEO-stationary satellite downlink communication with its users, which is not directed, can be received by anyone in a continent-sized area of millions of square kilometers [8]. The uplink communication between satellite and the ground station, on the other hand, is highly directional and can only be received with sensitive equipment near the nominal ground station target. LEO satellites have typical spot beam diameters of 40 km for Starlink to 400 km (Iridium).

Satellite Communication Protocols

On board the spacecraft, the traffic is processed in the Communication Module (COM), where usually multiple layers of network protocols are decoded and processed. Each layer manages one specific task to provide abstraction for higher layers. For space vehicles, the Consultative Committee for Space Data Systems (CCSDS) family of protocols has been established as the de facto standard, however, CCSDS does not define specific Telecommands (TCs). Instead, specific commands are defined by the European Cooperation for Space Standardization (ECSS) Packet Utilization Standard (PUS) commonly used, especially in Europe. In the following, we will briefly introduce the CCSDS protocol parts relevant for this work as well as the ECSS PUS standard for TMTC.

CCSDS

The CCSDS is a committee consisting of several space agencies worldwide to agree upon commonly used protocols for space applications. Besides other protocols, they standardized the commonly used Space Data Link Protocols (SDLP) for the data link layer [9]. The protocol is meant for data frame transmission over a point-to-point link.

Space Data Link Security (SDLS)—To secure various protocols of the data link layer, including SDLP, the CCSDS has introduced the SDLS protocol to secure the data link layer against tampering and eavesdropping. The protocol implements a security header consisting of a Security Parameter Index (SPI) field to determine the security association, i.e., which stateful session should be used for the specific packet. The session also determines which suite of cryptographic protections should be selected, i.e., which authentication and encryption algorithms [10]. Further fields in the protocol are optional but include a sequence counter to defend against replay attacks. The protocol also includes a trailer, which is located after the data frame body when considering an entire packet. The trailer consists of an optional Message Authentication Code (MAC). The data between the header and trailer can be encrypted.

ECSS PUS

The ECSS PUS refers to a set of standards concerning telemetry and telecommand packets used in space applications. With ECSS PUS, we are specifically referring to `ECSS-E-70-41A` and the newer version `ECSS-E-ST-70-41C`. The standard sets the requirements for the packet format, contents, and the associated procedures

to use them. Ultimately, the standard defines 16 services with dedicated telecommands and telemetry responses for each service. The standard, however, is more of a guideline for implementing these services, as many fields are optional, and the set of implemented commands can vary depending on the exact implemented capabilities. In addition, due to the large scope and complexity of capabilities that the standard covers, certain TC implementations may have been skipped in the implementation. This leads to the situation that even if it is clear that the standard is used on a spacecraft, an attacker may still have to enumerate which specific packet formats and capabilities have actually been implemented.

3. Attacker Models

We consider two attacker models that differ in their capability to access the reconnaissance target.

External Attacker

An *external attacker* operates from outside the organization that controls the target satellite, lacking specific internal knowledge about the target system. However, the attacker possesses the necessary hardware to communicate with the target satellite, independent of the satellite operators and the ground station infrastructure that is intended to control the satellite.
Moreover, this attacker also possesses Open Source Intelligence (OSINT) capabilities. This means they can utilize publicly available information about the satellite system such as specifications, standard protocols, or relevant technical publications to inform their attack strategies. Through OSINT, they can gather information that, while not classified or internal, can be substantial. Our model also assumes the attacker's familiarity with widely used protocol stacks such as CCSDS and ECSS. This knowledge allows them to make educated guesses about the target's protocol stacks and potentially identify vulnerabilities or misconfigurations within those protocol implementations. While this attacker lacks specific technical insights into the target, they have substantial knowledge of cryptographic protections and can exploit weaknesses therein. This includes exploring potential timing side channels, taking advantage of replay vulnerabilities, or identifying and exploiting insecure protection methods.
In sum, the external attacker describes any sufficiently educated and motivated hobbyist attacker without access to insider information.

Privileged Attacker

The second model, termed a *Privileged Attacker* or *Internal Attacker*, encompasses all capabilities of the *external attacker* but with the key addition that they possess the secret key required to sign and encrypt Telecommands (TCs). The attackers might have gained this key through a document leak, leaked source code, or previous cryptographic attacks that have compromised the secret key. In this model, we also include the possibility that the target satellite's cryptographic protections are either temporarily disabled, e.g., due to ongoing recovery attempts by the operators, or permanently absent due to missing protections in the first place. Hence, we only assume that the attacker can issue valid TCs and receive and read TM, regardless of the technical reasons or requirements to do so.
Despite this advantage, the attackers lack detailed knowledge about the available TCs. This necessitates reliance on educated guesses or enumeration strategies to identify potential TCs. Further, the attacker is not privy to detailed information about the satellite's internal mechanics, such as a firmware image or exhaustive technical documentation. This limits the understanding of the satellite's behavior in response to certain commands, possibly impeding the development of an effective attack strategy.
Hence, the attacker's primary option to learn more technical details about the target spacecraft is through the reconnaissance techniques detailed in the following.

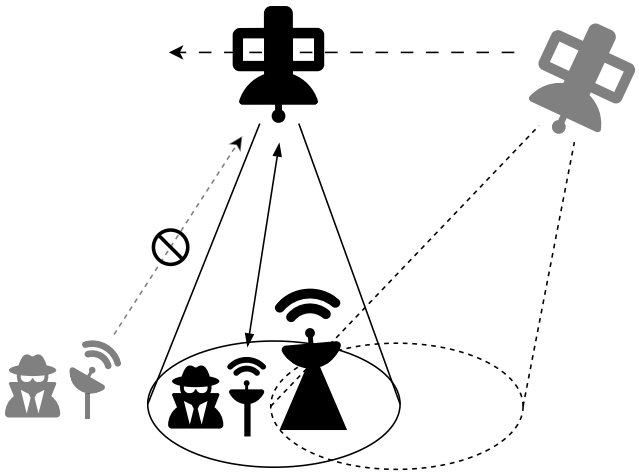

Figure 1. **Satellite and GS Pointing**: Directed Antennas on a satellite might impose geographical location constraints on attackers

4. Reconnaissance Goals

To successfully compromise any systems, attackers need to conduct a reconnaissance phase. This phase aims to achieve a series of reconnaissance goals, e.g., information required to conduct the impending attack. To this end, we formulate a series of *reconnaissance goals* for satellite systems. For each high-level goal, we discuss a list of subgoals. These subgoals are by nature non-exhaustive, as an arbitrarily complex system may have any number of parameters that need to be considered; instead, we focus on common goals that likely apply to a majority of systems.

Spacecraft Tracking & Operations

A successful attack requires the attacker to know where the target spacecraft is at a given time, which can be deducted from the TLE (see Section 2). In addition to the satellite's position, the satellite's attitude can also be relevant depending on the antenna type. Figure 1 shows how a satellite with a bidirectional antenna is in attitude towards the ground station of the legitimate operators. The directional nature of the antenna prevents attackers from having their malicious traffic received by satellite if the satellite is not attituded towards the attacker's GS. Attackers must account for such cases, as it restraints the physical locations a satellite is reachable from, even if it technically passes over it. In the following, we assume that an attacker initially only knows the name of a target satellite.

Space Object ID—Space Situational Awareness (SSA) systems provide the capability to track objects in orbit, but do not distinguish between debris and active satellites. Inher-

ently, they only follow specific trajectories and monitor space objects' locations and velocities. They do not inherently have the capability to determine these objects' operational status or intent. For a more comprehensive understanding of the space environment, additional analysis and data fusion are required. Specifically, this means that for an SSA system, there is no connection between the ID of the object they are tracking and, for example, the satellite's name. Hence, it is necessary to map a satellite's name to some *space object ID* tracked by an SSA system.

Tracking (TLE)—The tracking information of the orbital object is described using the TLE (ref. Section 2). Obtaining the TLE for a target satellite is crucial to predict its future orbital positions, which is necessary for various subgoals such as the antenna pointing direction or to determine the time window in which a satellite will pass over a ground station.

Ground Station (GS) Location—The location of the GS used by the legitimate satellite operators can yield valuable insights, such as determining the satellite's attitude, communication time windows, and geographical constraints to receive or send traffic from (GS pointing). All three of these points are discussed in their own reconnaissance goals.

GS Pointing—Pointing is required if the satellite communicates via (semi)-directed signals. However, even with precise pointing information, the satellite may deploy a spatial isolation mechanism to identify the signal source to prevent attackers from sending from arbitrary locations to the satellite. In these cases, attackers also have to identify the *Ground Station Location*. On the contrary, TMTC signals often utilize UHF or VHF signals, which are usually omnidirectional, skipping the need for precise pointing information.

Satellite Attitude—The satellite's attitude describes the satellite pointing direction, i.e., to which specific direction the antenna is pointed towards. This becomes relevant for an attacker, if the antenna is directed (see Figure 1). Bidirectional antennas are often used in bands such as the S-Band and higher frequencies. In contrast, bands such as VHF and UHF usually use omnidirectional antennas, where the satellite's pointing is less relevant.

Operational Time Frame—Satellites and their corresponding ground stations typically operate on schedules, conducting specific tasks at predetermined times. These schedules may include communication sessions, data transfer periods, maintenance activities, or scientific data-gathering phases. Understanding these operational timings can reveal periods where the satellite is either more vulnerable to attacks or when such activities would be less likely to be detected.

For instance, during periods of high data traffic, an attack might blend in with the legitimate data flow, making it less noticeable. Similarly, if the ground station is less staffed or monitored during certain hours, it may offer an optimal time window for an attack.

Time Window—Successful communication with a spacecraft requires identifying the appropriate time window for effective radio communication when the satellite is in a reachable orbit. Satellites, especially those in Low Earth Orbit (LEO), follow a specific path around the Earth, which results in the satellite's position relative to a ground station constantly changing. While the *time window* is primarily dictated by the satellite's orbital position, which is identified through it the TLE, other factors such as the satellite's attitude and operational time frames can be relevant.
Furthermore, other factors such as atmospheric conditions, ionospheric disturbances, or radio frequency interference could also impact the available time window for effective communication with the satellite.

Radio Communication Parameters

Establishing signal reception is the first step to communicating with wirelessly connected systems, followed by demodulation and decoding. Thus, we formulate the next reconnaissance goal as *Radio Communication Parameters*. This encompasses knowledge about the radio parameters, signal processing protocol, and the spacecraft's location. We emphasize that this section does not intend to be a signal-processing guide but rather collects several common parameters. Common subgoals include:

Signal Strength—Signal strength is fundamentally a measure of the power level received by the satellite from the attacker's transmission. A weak signal may not be recognized by the satellite's communication system. Moreover, the requisite signal strength is not static and depends on various factors, including the satellite's altitude, the distance between the satellite and the attacker's transmission equipment, atmospheric conditions, and potential interference from other radio sources.
It is also important to note that many modern satellite systems employ mechanisms such as automatic gain control (AGC) to adapt to fluctuations in received signal strength. Attackers need to consider these mechanisms in their planning, as these could potentially counteract their attempts to blend in with normal traffic by adjusting their signal strength.

Frequency—Each satellite communication system operates on designated frequency bands, such as S-band, C-band, X-band, or K_a-band. International bodies, like the International Telecommunication Union (ITU), typically standardize and regulate these frequencies. While the band itself can often be deducted from the mission type, i.e., modern internet services usually utilize K_u- or K_a-band, a more precise frequency value, such as the *center frequency* and *bandwidth* is required to establish a connection.

Synchronization Methods—An attacker must consider several synchronization methods for different parts of the radio link de/encoding. Frame synchronization ensures that the boundaries between successive data frames are correctly identified. Time synchronization ensures that the clocks of the transmitter and receiver are aligned.

Modulation— Satellite systems may use various types of demodulation schemes, such as Phase-Shift Keying (PSK), Frequency-Shift Keying (FSK), or Amplitude-Shift Keying (ASK). The attacker needs to correctly identify and implement the same demodulation scheme as the target satellite to interpret its signals accurately. Furthermore, the demodulation process often requires exact or estimated knowledge of several parameters such as carrier frequency, symbol rate, and phase reference. Therefore, an attacker would also need to identify these parameters accurately.

Error Correction—Error correction ensures the integrity of communicated data by identifying and rectifying errors introduced during transmission. Various error correction codes exist, such as Convolutional Codes, Reed-Solomon Codes, Turbo Codes, and LDPC (Low-Density Parity-Check) Codes. Each of these methods has its error-correction parameters and

implementation specifics. Understanding the target's error correction becomes primarily interesting in case the attacker actively sends traffic to the target. On the other hand, if an attacker aims to record telemetry, long-term error correction likely increases the yield of correctly decoded telemetry or tracking packets.

Network Protocol Stacks

After establishing a radio link, attackers can send and receive traffic, which is done in the form of network packets. Hence, we formulate the third main attacker goal as identifying the *Network Protocol Stacks*. This goal again consists of multiple subgoals, of which attackers might only need a subset.

Point-to-Point Protocols—Point-to-point protocols facilitate direct communication between two nodes in a network, i.e., between a ground station and a satellite. They are also referred to as "*data link layer*" [9]. Understanding these protocols is essential for an attacker as they contain information such as packet length and the packet's next address or identification. This information was not yet available through radio communication parameters and hence delivers insights that allow for effective traffic eavesdropping even with the upper layers not fully reconcilable. In some cases, such as CCSDS' SDLP protocol, there are separate but similar protocols for TC and TM traffic, which requires understanding two separate protocol stacks to establish a bidirectional communication [11].

Vendor-specific Implementation Details— Spacecraft often utilize specialized radio hardware that was intentionally designed for space systems. To set themselves apart from other manufacturers, companies often implement vendor-specific details like a certain set of telemetry submitted repeatedly as part of a beacon, security features as part of a black-box security product, or specialized fault-resistant transmission protocols. Depending on the nature of the protocol, an attacker must understand these aspects to establish a radio link and actively send traffic. Further, due to being vendor-specific, the protocol or the implementation might not have seen extensive testing, making it an interesting attack surface.

Cryptographic Communications Protection—The communication of a spacecraft might be secured using cryptographic primitives to prevent attackers from injecting malicious commands or eavesdropping on legitimate communication with the ground station. An attacker needs to understand if there is such a protection. A recent paper by Willbold et al. has shown a severe lack of these essential countermeasures [6]. Nonetheless, even if present, they might be vulnerable to attacks such as replay, forging, or side-channel attacks, making it necessary for an attacker to understand and analyze them in detail.

Network Protocols and Routing— Network layer protocols handle routing and thus enable communication between network nodes without a direct path but through other nodes. This is relevant for spacecrafts as multiple components, such as the power supply or the attitude control, can often receive telecommands through the COM. Hence, for an attacker that wants to exploit vulnerabilities on any of these components, it is paramount to understand the routing mechanism in the protocol stack.

TMTC Protocols

With the ability to exchange well-formatted network packets, an attacker can start analyzing the TMTC protocol stack. Generally, the TMTC stacks are the most customized part of the network protocol stack, as it sits on a high protocol layer and requires the most customized parts, such as mission-specific functionalities or operator preferences on the command set.

Telecommand Set—The TC set of a TC protocol stack describes the list of available telecommands. If an attacker can determine which commands are available without reverse engineering or analysis of all of them, it allows to focus efforts on potentially security-critical commands. For example, identifying a memory modification command or a software patching command can be enough to achieve an attacker's goals without analyzing other TCs.

TMTC Formats—Understanding the specific byte-exact formats of individual TCs and TM messages is crucial for at least a subset. Retrieving the format of TM messages can help in understanding the response to a potentially faulty or just partially faulty TC, which allows an attacker to reiterate. Subsequently, an attacker must retrieve the format of at least one (but likely multiple) TCs to perform a successful attack on a satellite.

5. RECONNAISSANCE STRATEGIES

We now describe a series of reconnaissance strategies that aim to recover the information needed to fulfill the reconnaissance goals listed in Section 4. We thereby categorize the strategies into the five main categories: *OSINT*, *common options*, *Commercial off-the-shelf (COTS) acquisition*, *passive analysis*, and *active enumeration*. We list several strategies and information sources and document which information they can provide. Since the categories of passive analysis and active enumeration can have any number of strategies, we focus on strategies that target the widely used CCSDS and ECSS protocols as they are the most likely encountered protocols for attackers.

Open Databases

Open Source Intelligence leverages open resources such as media reports, public databases, forums, and other openly accessible digital platforms to develop insights. In this category, we track open databases and public regulator filings. Since the potential sources of public information on specific satellites are too extensive to be listed here exhaustively, we focus on the following key sources of information.

Spacecraft Tracking—The *Combined Force Space Component Command (CFSCC)* provides public SSA based on information provided by the *18th Space Defense Squadron (18 SDS)*. This SSA is published on the `space-track`[2] website and is publicly accessible.

The internal process to track a spacecraft starts at launch. Each launch receives an ID in the format of `YYYY-XXX`, where the first four digits describe the year of the launch and the last three digits are a continuous counter over the year. Then each object ejected during the launch receives an *International Designator (INTLDES)*, also called COSPAR (UN Committee on Space Research) ID. This COSPAR ID is based on the launch ID extended with a letter (i.e., `YYYY-XXXA`), where the last letter increases for every ejected object. For example, the *James Webb Space Telescope* has the COSPAR ID *2021-130A*, as it was launched in 2021,

[2] `space-track.org`

Table 1. **Reconnaissance Goal to Strategy Mapping**: Shows which information can be recovered using which strategy. Strategies can achieve a reconnaissance goal in most, if not all cases ('x'), in some cases ('∼'), or are unfit for a specific goal ('-').

Goal / Strategy	Open Database	Regulator Filings	Common Option	COTS Analysis	Traffic Analysis	Active Enumeration
Space Object ID	x	-	-	-	-	-
Tracking (TLE)	x	-	-	-	-	-
GS Location	-	∼	x	-	∼	-
Time Window	-	∼	-	-	x	-
TT&C Channels	∼	x	-	x	x	-
TT&C Modulation	∼	∼	x	x	x	-
TT&C Synchronization	-	-	x	∼	x	-
TT&C P2P Protocols	-	-	x	∼	x	-
Vendor-Specific Prot.	-	-	-	x	x	-
TT&C Crypto. Prot.	-	-	∼	∼	x	∼
TT&C Network Layer	-	-	-	∼	x	-
Telecommand Set	-	-	-	∼	x	x
TMTC Formats	-	-	-	∼	x	x

on the 130th launch that year, and was the first (A) ejected satellite of that launch.

The US Space Surveillance Network (SSN) then tracks the objects ejected during lunch using radar and calculates a TLE. These tracked objects are assigned a Satellite Catalog (SATCAT) Number, also called NORAD (North American Aerospace Defense) ID, as the US NORAD was historically tasked with tracking space objects. However, at this time, there is no connection between a satellite's name and its TLE or NORAD ID. The exact association or identification process can vary. The 18 SDS offers forms for satellite operators to fill out ahead of launch to identify their satellite after launch. If this succeeds, then the NORAD-ID and COSPAR ID are associated with a satellite's name and their TLE is published on the `space-track` website. These TLE also get updated in case the satellite performs orbital corrections after launch, i.e., to reach a target orbit. Although this process, to us, appears not mandatory, unless satellite operators provide their own SSA capability, they have to provide this information to receive the TLE for their satellite. Additional information, such as the aforementioned forms, is only required if there is ambiguity regarding the satellite.

Interestingly, essentially all of this tracking information is provided publicly, allowing attackers to utilize this information to retrieve the real-time position of a target satellite. Hence, attackers only have to look up a target satellite's name in `space-track` to find out their TLE.

Public Regulator Filings

Satellites combine a number of technologies that are regulated by both international agencies and local government bodies. Satellite operators typically register with local authorities in their country of operation, i.e., where customers are present and where the satellite itself is operated from. Using the US as an example, the FAA licenses rocket launches, and the FCC coordinates wireless spectrum use. Additionally, these organizations maintain public databases, rendering them a rich resource for OSINT. In the following, we focus on utilizing public regulatory filings to identify TT&C frequencies.

FCC Filings—Satellites operating wireless links to and from the US are typically registered with the FCC, using form 312 (*Application For Satellite Space And Earth Station Authorizations*). This process involves the disclosure of key parameters of the satellite system. The most accessible and information-rich resource, if available, are *Applications to Launch and Operate (SAT-LOA)* documents filed with the FCC. These documents detail the mission and technical parameters as well as company details (e.g., major shareholders); the appendix *Schedule S* contains detailed radio parameters. This applies both when the satellite is operated from the US or serves users in the US. In addition, the FCC publishes these reports, making them available for OSINT. Generally, these reports include the general system design of a satellite and its radio communication system. Specifically, since the FCC is tasked with allocating the radio frequency spectrum across the US for interference-free communications, the reports include details about which channels, i.e., center frequency and bandwidth, are allocated to the satellite and, more specifically, to which subsystem. Hence, there are details about the exact channels used for TT&C for a given satellite.

Amateur Radio Filings—Universities and academic satellites often cannot afford costly licenses to dedicated frequency bands as allocated by government bodies. Instead, they may rely on amateur radio. Amateur radio comes with several requirements regarding the accessibility of data. Specifically, all data must be readable, and decoding formats must be readily available. The only exception to this rule is command-and-control uplink, which may be encrypted and authenticated [12]. Hence, satellite operators must publish telemetry formats for their satellites, making it trivial to decode this information. Due to the openness of information, the *SatNOGS*[3] project tracks the telemetry of satellites worldwide using receivers around the globe provided by hobbyists. This information is collected in the *SatNOGS* database and is publicly available.

Common Options

In many cases, there are either only a limited number of options or some of the options are significantly more likely to be used. Hence, individuals familiar with the matter can make an educated guess as to which options or parameters are being used. This becomes easier and more accessible as more resources about a topic exist. In the following, we describe how, especially for radio parameters, there is a plethora of tools and resources available. While the same cannot be said for the protocol stacks used in these space applications, there are only very few common choices there, making educated guessing a valid strategy.

GS Location Inference—There are multiple ways to determine the location of a ground station. If the communication time windows between the operators and the satellite are known, and assuming that the satellite's TLE is known, then the GS location can be infered. Another method is FCC filings, which sometimes include either rough GS locations like cities or even precise locations with an actual address. Further, large satellite operators likely deploy a large ground station, which can often be spotted from satellite or aerial images. In addition, for large vendors, there are only so many GS

[3] `satnogs.org`

locations, i.e., *Amazon Web Services (AWS)*.

COTS Analysis

Commercial off-the-shelf (COTS) components are an essential aspect of the *New Space* era as they help to reduce the cost of new spacecraft by centrally developing components and selling them to multiple projects. However, COTS components are also an interesting source of information for attackers as they can purchase the same components used on a target spacecraft to analyze them for non-public technical details or to identify vulnerabilities.

Vendor Documentation—Oftentimes, the requirements technical details are disclosed in the documentation accompanying a COTS component. For example, for COM component solutions, there are often vendor-specific protocols as described in our reconnaissance goals (ref. Section 4). Their documentation then provides detailed insights into these protocols, as satellite operators potentially have to implement a GS counterpart to emit or receive the protocol. For other components on a satellite, the documentation may disclose precise interface information and how to talk to the device, for example, after a satellite has been compromised.

Software Reverse Engineering—In cases where either no documentation is provided or COTS have been obtained through non-official channels, there is still the option to reverse engineer parts of the software, for example, by dumping the program code on the COTS first and performing software reverse engineering afterward [13], [14], [15].

Passive Traffic Analysis

We subsume all techniques under traffic analysis that attempt to passively collect information about the target using eavesdropping. Using these techniques, it is generally possible to learn much about the TT&C signal and the invoked protocols. However, listening to TT&C traffic is usually only feasible close to a target GS. TT&C are usually only emitted from (one of) the satellite's ground station(s), and the satellite often only emits telemetry data if either above such a GS or when specifically asked to do so. Hence, applying these techniques generally requires having already determined the location of the satellite's GS, which we formulated as a reconnaissance goal (ref. Section 4). However, since this strategy is constrained through physical locations and involves the physical action for attackers to relocate close to GS, it is not always feasible.

Signal Analysis Tools—The radio community has developed a plethora of radio communication analysis open-source tools to help signal analysis and to automate the process. In general, the advent of flexible and affordable software-defined radios and the *GNURadio* framework [16] in the 2000s has enabled many users to receive, analyse and process radio signals without requiring specialized hardware and advanced electrical engineering knowledge.

One recent example is provided by *FISSURE* (Frequency Independent SDR-based Signal Understanding and Reverse Engineering [17]), an open-source RF and reverse engineering framework that contains hooks for detection, classification, protocol discovery, attack execution and vulnerability analysis. It offers AI/ML-based tools, which help the user to automate reverse engineering and recon against RF communication, including those of satellites.

With such tools, it is generally possible to retrieve most, if not all, radio communication parameters for typical satellite communication systems. The required effort may vary significantly depending on the available information and existing building blocks for signal processing for a specific target.

Protocol Reverse Engineering— Another approach possible after capturing TT&C traffic is the analysis using protocol reverse engineering, attempting to determine protocol fields and potential values from raw messages. Similar to the aforementioned signal analysis tools, there are also open tools and active academic research to aid this process [18]. Compared to the signal analysis tools, they are less capable and still require significant manual effort to reverse engineering protocols. However, especially in lower-layer protocols and with increasing mission complexity and budget, the use of standardized protocols, such as CCSDS SDLP or AOS, become more relevant [6]. Although they are standardized in free and open standards, they still contain optional fields and values, prompting additional analysis requirements to determine the exact details of each protocol layer. For the commonly used Space Data Link Security (SDLS) protocol, we elaborate two enumeration techniques during our discussion of *active enumeration techniques*. *Kaitai Struct*[4] and *scapy*[5] have also been as tools used to successfully reverse engineer unknown satellite communication from the bottom up.

Active Enumeration

Active enumeration strategies attempt to learn about the target's behavior by sending data and observing the reaction. In case the target spacecraft employs *cryptographic access protection* (ref. Section 4), the surface that is actively enumerable for an *external attacker* (ref. Section 3) might be severely limited. In these cases, we assume an *internal attacker* (ref. Section 3) who has access to the cryptographic key(s) to pass the protection.

In the following, we focus on the CCSDS and ECSS protocol stack combination introduced in Section 2. Specifically, the ECSS PUS protocol refers to a pre-defined list of telecommand and telemetry with a roughly defined format. In addition, during the evaluation in Section 6, we will experimentally evaluate these techniques on real-world satellite hardware. We focus on these techniques with significantly greater technical detail, as many other aspects highlighted in this paper have pre-existing research and/or tools, while the enumeration of the following aspects has not been documented yet.

SDLS - Implementation Enumeration—The primary security protocol used in the CCSDS standard is the SDLS protocol. Since satellites are most likely to use this protocol family and, more specifically, SDLP or AOS, they are often used in combination with SDLS as they are compatible and provide a relatively simple way to implement TC protection.

Security Parameter Index (SPI) Enumeration
The only parameter required for the SDLS protocol is the Security Parameter Index (SPI). As previously mentioned, protocols such as SDLS often include optional values or used-defined value ranges. Such is the case for the SPI values, which lets users define a range of $2^{16} - 2$ values for pre-defined crypto-suites. While determining one or a few might be feasible through traffic analysis, the full value space can only be determined through enumeration. While iterating

[4] https://kaitai.io/
[5] https://scapy.net/

Table 2. **ECSS PUS Error Codes**: The error codes to identify which part of the packet processing failed

Error Code	Description
0	Illegal APID
1	Incomplete or invalid length packet
2	Incorrect Checksum
3	Illegal packet type
4	Illegal packet subtype
5	Illegal or inconsistent application data
> 5	Mission-specific codes

$2^{16} - 2$ options in most cases is not a problem, such ranges become a problem for active enumeration on a spacecraft, as the communication time window and bandwidth might be limited. Additionally, operators might notice a stark increase in communication attempts and enact countermeasures. Hence, attackers can likely only iterate a fraction of the full space.

Replay Protection Probing
The security header in the SDLS protocol only defines the SPI field as mandatory [10]. While the standard defines further fields such as an *initialization vector*, a *sequence number*, and a padding field, none of these are mandatory. Hence, without the *sequence number*, replay protection is also not mandatory in the protocol. Attackers can analyze the header fields, and if the header consists of only 2 bytes, the protocol does not provide replay protection. Even if there are more fields, it is not certain that there is a sequence number. Identifying a field as the sequence number can be done by either observing packets and checking if the number ticks up sequentially or by sending the packet manually and checking if only sequential numbers are accepted. If non-sequential numbers are accepted, the field is either not a *sequence number* or does not follow the standard requirements, which allow for replay attacks.

Telecommand Enumeration—Telecommand protocols often denote the type of the incoming telecommand using an integer field that identifies the command, which we will refer to as *TC ID* in the following. This approach is often used in command-and-control protocols and saves bandwidth compared to approaches where, i.e., a string denotes the command type seen in more human-readable protocols. Examples can be seen in the widely used *Cubesat Space Protocol (CSP)* implemented in the open-source library *libCSP*, and the ECSS PUS standard.

In theory, enumerating all possible commands is as simple as exhaustively iterating this TC ID field and evaluating the response for each service. However, even simple protocols employ a two-byte field requiring 2^{16} enumerations, which is usually not feasible for the bus system of satellites in a reasonable time. The reasons are low-bandwidth communication, low-performance processors, and limited power budgets. Hence, even a task as simple as exhausting a 2-byte TC ID field can become challenging for attackers.

For the widely used ECSS PUS standard, attackers can exploit the TC ID bytes being split in a 1-byte *serviceId* and 1-byte *subserviceId* paired with verbose error output to fully enumerate the TC ID space. Specifically, the standards lists *services*, which group TCs by a certain type, such as the *Memory Management Service* or *Event Reporting Service*. For each service, the *subserviceId* denotes a specific TC in the service. Further, several error codes return whether the accessed *serviceId* or *subserviceId* is valid. This allows us to first enumerate the services through the *serviceId* space with 256 options, where the error message returns whether the service exists. Then, for each existing service, we iterate the existing TCs through the *subserviceIds*. The error message reveals if the specific TC exists.

With this approach, attackers can significantly reduce the required iterations to fully enumerate all implemented TCs.

Packet Length Enumeration— In cases where TCs do not follow a well-known standard, identifying the length of the expected TC packets is often a first step to reverse engineering their intended contents. This also applies to the TCs established in the ECSS PUS standard. Curiously, the standard often does not dictate a precise way to implement a TC, but rather lists several optional fields and fields of varying lengths. This leads to multiple implementations of this standard, likely having different TCs where options can be selected. We refer to this as having different *flavors* of the standard. Hence, even in this standardized environment, it is crucial to retrieve the expected length of a TC to, i.e., reason about which of the optional fields are included.

In the case of ECSS PUS, attackers can again exploit verbose error messages (ref. Table 2) to determine the length of TCs. If the payload length is incorrect, the status code 5 is consistently returned. Conversely, if the payload length was correct, but its contents were incorrect, a random status > 5 code is returned, indicating a vendor- or mission-specific error that likely provides developers closer insights. This makes it possible to enumerate the expected packet size for a specific TC, by extending the enumerated command by one or two bytes, until the error message changes to > 5. While expanding by one byte will not miss the correct length, extending by 2 bytes might be a quick success, whereby we can exploit the fact that protocols often only accept even or odd lengths. As such, after learning the length and whether the TC length is consistent, even, odd, or neither, we can speed up the process for further enumeration on that specific target.

It should be noted that TCs can have varying lengths if, i.e., arrays are transmitted, fields are optional and their existence is indicated through a different field. In these cases, a more rigorous protocol reverse engineering approach is needed.

6. STRATEGY EVALUATION

We analyze the feasibility of several strategies discussed in Section 5 using real-world examples. First, we determine the feasibility of extracting TT&C channel information for regulator filings, specifically from FCC filings. Finally, we evaluate the feasibility of several active enumeration strategies using a real-world example. We evaluate the amount of information from freely accessible FCC filings to demonstrate the OSINT capabilities around satellite radio parameters as well as the technical methods required to extract information from commonly used protocols. The latter part has not been demonstrated yet.

TT&C Channels from FCC Filings

We perform case studies on selected satellites and constellations to assess the effectiveness of using FCC Filings to determine crucial TT&C channel and GS location param-

Table 3. **Results from FCC Filings**: Check if FCC Filings contain TT&C channels and GS information

Type	Satellite	Reference	TT&C	GS
VSAT	ViaSat-3	SAT-LOA-2019061700048	14,000.3, 14,001.0, 14,498.5, 14,499.0 MHz	-
VSAT	EutelSat 133WA	SAT-MPL-2018090800068	2085.688 and 13,750.6 MHz	x
VSAT	OneWeb	SAT-MPL-20200526-00062	19,265–19,300 MHz	x
VSAT	Starlink	SAT-MOD-2018110800083	13,875.0, 13,925.0, 13,975.0 MHz	$\sim$
SA	HawkEye 360	SAT-LOA-20190102-00001	2063–2065 MHz, 432–438 MHz	x
EO	PlanetLabs Pelican	SAT-MOD-2022042100042312	2056, 2066, 2086, 2096 MHz	x

eters, which we both defined as reconnaissance goals (ref. Section 4). We selected satellites or constellations from 6 established US-based companies for our experiment. The most accessible and information-rich resource, if available, are *Applications to Launch and Operate (SAT-LOA)* documents filed with the FCC. These documents detail the mission and technical parameters, as well as the appendix "*Schedule S*", which contains detailed radio parameters.

Table 3 shows columns with the satellite's or constellation's mission (VSAT, Situational Awareness, or Earth Observation), the name, whether the FCC filing contains at least one TT&C band, and if the filing contains the location of the GS used to operate the satellite. In the following, we discuss the results of the six satellites/constellations.

ViaSat-3 is a planned constellation of three Geostationary Orbit (GEO) K_a-band Very Small Aperture Terminal (VSAT) satellites, where the first was launched in 2023 [19]. The satellite contains the payloads V*iaSat-89W* and -*US89* with separate FCC filings (SAT-LOA-2019061700048) [20]. The satellite utilizes a portion of the conventional K_u-band for TT&C and conducts TT&C operation from outside the US. However, the precise GS location was not part of the application. The center frequencies of the TT&C channel were filed as 14,000.3 MHz, 14,001.0 MHz, 14,498.5 MHz and 14,499.0 MHz.

EutelSat 133WA is a decommissioned satellite for EutelSat. The FCC filing SAT-MPL-2018090800068 documents that EutelSat requested US market access for their satellite [21]. The satellite utilizes two K_u-band telemetry channels with center frequencies of 11,451.091 MHz and 11,452.570 MHz and a bandwidth of 300 kHz. Further, the satellite utilizes one K_u-band command channel, with a center frequency of 13,750.6 MHz and 600 kHz bandwidth. The TT&C operations will be conducted from earth station facilities in Mexico, listed with an address. Notably, the written information differs from the channel list appended to the application, which lists 13,750.6 MHz (600 kHz bandwidth), 2085.688 MHz (400 kHz bandwidth), and a wide-beam, nearly omnidirectional, backup antenna.

OneWeb requested in its *Schedule S* report the TT&C uplink to be 27,500–27,600 MHz with BPSK modulation, the download control TT&C at 19,700–19,770 MHz, and the downloading payload control at 19,265–19,300 MHz with QPSK modulation. The two ground stations are in *Tysons, Virgina, US* and in the UK, respectively. Interestingly, we found several ESA documents detailing the exact technical details for OneWeb's payload testing equipment [22]. Such insights might be interesting for potential attackers.

StarLink requested at least nine TT&C channels in their SAT-MOD-2018110800083 filing [23], three of which are at 13,875.0 MHz, 13,925.0 MHz, and 13,975.0 MHz with 50 MHz channels each. While there is a vibrant community tracking StarLink gateway locations [24], these are likely not used for TT&C as the FCC filing states that they only plan to deploy two TT&C ground station in the US, one on the West coast and one on the East coast.

HawkEye provides in their SAT-LOA-20190102-00001 filing [25] rather detailed information, even compared to other filings. The TT&C channel is at 2063–2065 MHz and a backup TT&C channel at 432–438 MHz. There also exists a plethora of open information, including technical details for their radio setup, in a 2018 conference presentation [26].

PlanetLabs filed SAT-MOD-2022042100042312 for their Pelican constellation, which also is comparably detailed with the TT&C channels at 2056 MHz, 2066 MHz, 2086 MHz and 2096 MHz.

SDLS Analysis using Active Enumeration

We evaluate the SDLS-related active enumeration strategies highlighted in Section 5 by evaluating how many SPI field enumerations are feasible in a given time frame and how to determine if the given SDLS implementation is susceptible to replay attacks.

For our evaluation, we use the flatsat model of a real-world satellite, which we cannot disclose the name of due to a Non-Disclosure Agreement (NDA). The satellite uses an S-band antenna for the TT&C channel to receive TCs and send telemetry. We interacted directly with the S-band processing board by sending the raw bytes that would have been decoded from the radio signal. The satellite then processed the incoming signal, and if it was a valid TC, executed the associated TC handling routines. The satellites implement an SDLP-based stack with SDLS and ECSS PUS on top of TMTC. The satellite contained a pre-exchanged key used internally in the SDLS implementation. For our SDLS experiments, we assume an *external attacker* that has no access to the pre-exchanged key or any other cryptographic material and has only obtained information through OSINT (ref. Section 3).

We disclosed all of our findings in a coordinated and responsible fashion to the satellite vendor and have received feedback acknowledging the findings and follow-up plans for a more detailed exchange.

SPI Field Enumeration—The only mandatory parameter of SDLS is the Security Parameter Index (SPI) field (ref. Section 2), which identified the security association sued for this packet, which in turn identified a cryptographic session that describes how to handle the packet regarding decryption and authentication. As already assumed during the strategy discussion in Section 5, fully enumerating the 2-byte field turned out to be not feasible. In our experiments, we iterated $\sim$ 3500 SPI values, which only covered around 5.3% of the

full space. hence, it can be assumed that for an attacker it is likely also not feasible to iterate the entire possible SPI space, and an attacker might have to resort to eavesdropping to learn about valid SPI values.

Replay Protection Analysis— During the reconnaissance strategies, we discussed that replay protection is not a mandatory part of the SDLS protocol, as the required fields are optional. The field meant for replay protection is a sequence counter, where the counter must be increased to prevent resending old packets. To determine if there is such a field, an attacker with no inherent detailed knowledge of the protocol stack can eaesdrop on multiple packets or actively send packets with various sequence values.

In our case, since we were provided with scripts to build valid packets, we noticed that a 6-byte time value was used instead of a sequence counter. This does not conform with the SDLS standard since a sequence counter has to be increasing *without gap* [10], which is hardly possible with a timestamp field. However, it also shows another approach to replay protection based on timing. The problem with timestamps against replay is that to still accept valid packets, there needs to be a grace period since timers on the GS and satellite are unlikely to be in perfect sync. Hence, we probed the grace period by sending a packet that requests current housekeeping telemetry with increasing older timestamps and future timestamps. In our analysis, the satellite accepted packets with a $[-60\,sec,\ +60\,sec]$ time window, resulting in a worst-case grade period of 2 minutes.

We especially find the observation interesting that a non-standard compliant timing field is used, which is also not defined in the older version of the standard [27].

TT&C Set and Format Analysis

Using the same experimental setup that was already used for the analysis of the SDLS protocol, we evaluate the feasibility of probing TCs implemented as part of the ECSS PUS standard. While we previously utilized our external attacker model, we are now considering the internal attacker (ref. Section 3), which has access to cryptographic key material to craft valid signed messages but lacks detailed knowledge of the internal attack surface of the satellite.

The ECSS PUS standard describes a series of services and, for each service, a set of TC and TM protocol fields. However, due to the complexity and the way the messages are structured, there is room for either additional custom messages for each of the services or to skip certain messages. Thus, even if it is known that the standard is used, the exact set of TC implements is not certain, nor are the exact formats of either the TC or the TM, as both have many optional fields. The satellite used during our evaluation uses version `ECSS-E-70-41A` of the standard.

TC Capability Enumeration— As described during the reconnaissance strategies (ref. Section 5), the exact (sub)set of telecommands implemented from the standard can be iterated without exhausting the 2-byte search space by exploiting the verbose error messages. We tested this strategy on our target satellite to evaluate in which time frame we could exhaust all options, which issues we might encounter, and to which degree custom messages are implemented.

During the process, we noticed the issue that there is no trivial way to tell if a TM packet belongs to the most recently emitted TC or to a previous TC. While some TM packets specifically identify the TC that emitted them, that is not the case for all. To counteract this, we sent a well-known *marker* command between every enumerated TC. For this marker command, we used the `Generate Housekeeping Report` TC which would emit some housekeeping TM parameters. Hence, we always first sent the marker then the enumerated TC then again a marker. We would thus receive the marker response, then potentially the enumerated TC response, and again the marker response. If we receive a marker response before the TC response, we know that there is no response, which can be the case. In addition, we always added a one-second wait period after every command, so as not to overwhelm the microcontroller.

In total, we found that our target satellite implemented nine of the 16 standard services. On top of this, four custom services were implemented for a total of 13 services. Across these 13 services, we identified 75 TC handlers, of which only 36 refer to handlers proposed by the standard, while the other 39 handlers are custom and would require reverse engineering to understand their meaning. Interestingly, this means that over half of all functionalities are non-standard. Further, we encountered a TC that put the satellite in a state of not responding anymore, which required us to restart the setup. After checking with the satellite developers, the satellite does turn off the UART board connection used for our testing under that specific command. This prompts the issues of telecommands that can put the satellite in a non-responding state in the wild. This breaks the enumeration process of the attackers but it might also be interesting for performing a Denial-of-Service attack.

While the total time to complete this enumeration will vary across setups, it took us roughly 30 mins to enumerate all possible TCs.

TC Payload Length—Since many of the TCs contain optional fields or varying length fields, it is a good starting point in the reversing process to determine the expected length of each TC. As described previously, the expected size of a TC can be enumerated by increasing the TC size with zeros and checking when the error messages change. It should be considered that some TC might have multiple valid lengths, for example, depending on some other field, which in our case would be zero. We did not account for such cases.

We were able to determine the size of 69 of 75 TC handlers. Out of these 39 commands, 7 had an empty payload. These were functions such as resetting the command schedule, a ping command, or requesting an event list. For the commands that we could not determine the size, it could be the case that they are missing the minimum size check.

7. DISCUSSION

After evaluating the information that can be obtained through OSINT such as the FCC filings, the space-tracking capabilities, and further documents mentioned during our evaluation, we were surprised by the amount of accessible information. However, as also mentioned, not all countries and jurisdictions report information as openly as the US does. Although we primarily portray this information in the hands of malicious actors in this work, we think that treating information this open should be standard as these hurdles first and foremost hinder legitimate actors such as researchers. The prevailing sentiment is that attackers will always find out this information, for example, at conferences when talking to the

people working with the information, in leaked documents, or from previous work experience. Hence, many aspects of security-by-obscurity practiced, especially, in the space and satellite systems area do not hold against the ease of accessing such information. For example, the recently published survey among space system engineers by Willbold et al. [6] mentioned that in many cases the community assumes that frequencies are secret and thus act as a security measure to prevent attackers from communicating with the victim satellites. However, in this work, we have shown that this information can often be accessed in official online databases. In a similar fashion, it should also be considered whether the FCC is required to publish information about security measures, such as the protocol used to secure the TT&C traffic. Such measures would likely have a large impact on the security-by-obscurity sentiment, and thus on the satellite cybersecurity field as a whole.

8. Related Work

General Cybersecurity Intelligence and Reconnaissance

The main framework in the area of cybersecurity reconnaissance is the MITRE ATT&CK framework. The reconnaissance phase framework involves the adversary's systematic gathering of information to identify and assess potential targets. This initial stage encompasses both passive and active methodologies, including OSINT collection and network scanning. By comprehensively mapping the target environment and understanding its vulnerabilities, adversaries lay the groundwork for subsequent stages of the cyber attack lifecycle.

Satellite Security

Security and privacy in satellite networks have recently enjoyed a renaissance in the academic computer security community. This was sparked by Pavur et al. in 2020, who outlined the ease of eavesdropping on the downlink contents of unencrypted legacy geostationary satellite systems [5]. It illustrated that it is feasible to identify and analyse the traffic of a large number of users in the same satellite footprint. Willbold et al. [6] conducted a first public experimental security assessment of three real-world satellite firmware images, finding several critical vulnerabilities. An additional survey of 19 satellite software developments indicated a lack of state-of-the-art security-focused software testing. In the same vein, Scharnowski et al. [28] apply modern embedded firmware analysis techniques to satellite payload data handling systems, discovering new bugs in several firmware images.

Previously, several works [29], [30] have pointed out that security in space and terrestrial security differ in key aspects. Falco et al. also introduced a framework to analyze the threats against CubeSats [31]. Further, Manulis et al. have discussed aspects of the *New Space Era* that must be considered in future security research [32]. Additionally, two reports on space asset threats by Harrison et al. have pointed out the dangers posed by attacks [33], [34]. Ultimately, there is a plethora of additional research on this topic, albeit typically of theoretic nature [35], [36], [37], [38] or involving simulations [8]. In 2020, the US Air Force Research Laboratory started the yearly Hack-A-Sat competition at the DEF CON conference, culminating in a successful live satellite hacking demonstration in the 2023 edition.[6]

[6] https://hackasat.com

Satellite Reconnaissance—Recent efforts in satellite security reconnaissance focus on creating frameworks to categorize space-related cybersecurity threats. Two popular frameworks are the Space Attack Research & Tactic Analysis (SPARTA) framework by the Aerospace Corporation [39] and the *SPACE-SHIELD* framework published by European Space Agency (ESA) [40].

Both frameworks list reconnaissance and information-gathering phases but focus on the abstract level. SPARTA in particular considers the collection of information on spacecraft design, descriptors, communication, and flight software. They list potential options and mitigations, which we explore in significant detail in this paper.

9. Conclusion

In this work, we have discussed which pieces of information are crucial to satellite cybersecurity reconnaissance as well as the strategies to obtain this information. We tested some of these strategies on real-world targets. The paper concludes that a large amount of required information is freely accessible or has a common option that can be correctly guessed with reasonable effort. Only the final steps to determine exact protocols and commands require eavesdropping on actual traffic. This eavesdropping in turn might be geographically restrained. Finally, we present some active enumeration strategies and test them on a real-world satellite and conclude that verbose error messages allow attackers to reasonably enumerate the available set of commands on a satellite.
Ultimately, the paper suggests that security-by-obscurity is not advisable in the space domain. Through the enormous amounts of information an attacker can obtain from open sources alone, not even considering minimal effort analysis, attackers can easily collect all required information to be able to launch a cyberattack against satellites.

Acknowledgments

The work was partially supported by the MKW-NRW research training group SecHuman and the Cyber-Defence Campus of armasuisse Science and Technology.

References

[1] N. Boschetti, N. G. Gordon, and G. Falco, "Space cybersecurity lessons learned from the viasat cyberattack," in *ASCEND 2022*, 2022.

[2] R. Santamarta. (2022) Viasat incident: from speculation to technical details. [Online]. Available: https://www.reversemode.com/2022/03/viasat-incident-from-speculation-to.html

[3] J. Menn, "Cyberattack knocks out satellite communications for russian military," *Washington Post*, 2023. [Online]. Available: https://www.washingtonpost.com/technology/2023/06/30/satellite-hacked-russian-military/

[4] M. Felux, B. Figuet, M. Waltert, P. Fol, M. Strohmeier, and X. Olive, "Analysis of gnss disruptions in european airspace," in *Proceedings of the 2023 International Technical Meeting of The Institute of Navigation*, 2023, pp. 315–326.

[5] J. Pavur, D. Moser, M. Strohmeier, V. Lenders, and I. Martinovic, "A Tale of Sea and Sky on the Security of

Maritime VSAT Communications," in *IEEE Symposium on Security and Privacy (S&P)*. IEEE, 2020.

[6] J. Willbold, M. Schloegel, M. Vögele, M. Gerhardt, T. Holz, and A. Abbasi, "Space odyssey: An experimental software security analysis of satellites," in *IEEE Symposium on Security and Privacy*, 2023.

[7] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "Mitre att&ck: Design and philosophy," in *Technical report*. The MITRE Corporation, 2018.

[8] J. Pavur and I. Martinovic, "The Cyber-ASAT: On the Impact of Cyber Weapons in Outer Space," in *International Conference on Cyber Conflict*. NATO CCD COE, 2019.

[9] CCSDS Contributors. (2023) OVERVIEW OF SPACE COMMUNICATIONS PROTOCOLS. [Online]. Available: https://public.ccsds.org/Pubs/130x0g4.pdf

[10] ——. (2022) SPACE DATA LINK SECURITY PROTOCOL. [Online]. Available: https://public.ccsds.org/Pubs/355x0b2.pdf

[11] ——. (2015) SPACE DATA LINK PROTOCOLS—SUMMARY OF CONCEPT AND RATIONALE. [Online]. Available: https://public.ccsds.org/Pubs/130x2g3.pdf

[12] AMSAT - The Radio Amateur Satellite Cooperation. (2018) FCC Part 97 Amateur Radio Licensing for CubeSats. [Online]. Available: https://www.amsat.org/wordpress/wp-content/uploads/2018/04/AMSAST_CubeSat_Licensing.pdf

[13] D. Quarta, M. Pogliani, M. Polino, F. Maggi, A. M. Zanchettin, and S. Zanero, "An experimental security analysis of an industrial robot controller," in *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2017.

[14] S. Wallat, M. Fyrbiak, M. Schlögel, and C. Paar, "A look at the dark side of hardware reverse engineering-a case study," in *2017 IEEE 2nd international verification and security workshop (IVSW)*. IEEE, 2017.

[15] M. Lungu, "Towards reverse engineering software ecosystems," in *2008 IEEE International Conference on Software Maintenance*. IEEE, 2008, pp. 428–431.

[16] E. Blossom, "Gnu radio: tools for exploring the radio frequency spectrum," *Linux journal*, vol. 2004, no. 122, p. 4, 2004.

[17] C. Poore, "Fissure: The rf framework for everyone," in *Proceedings of the GNU Radio Conference*, vol. 7, no. 1, 2022.

[18] J. Chandler, A. Wick, and K. Fisher, "Binaryinferno: A semantic-driven approach to field inference for binary message formats." in *NDSS*, 2023.

[19] ViaSat Inc. (2023) ViaSat-3 is satellite reimagined. [Online]. Available: https://www.viasat.com/about/viasat-3/

[20] JOHN P. JANKA. (2019) SAT-LOA-20190617-00048. [Online]. Available: https://fcc.report/IBFS/SAT-LOA-20190617-00048

[21] Carlos M Nalda . (2019) SAT-MPL-20180908-00068. [Online]. Available: https://fcc.report/IBFS/SAT-MPL-20180908-00068

[22] Steiner, Hans Martin and Wolf, Hermann and Danzer, Hermann and Perschak, Thomas and Micko, Jan and Ali, Mohammed and Neuhaus, Áin. (2019) One Web Power, TCR and Payload test System (PTS) EGSE - Final Report. [Online]. Available: https://connectivity.esa.int/system/files/OW-COM-RP-SIA-0001-FinalReport_1v0.pdf

[23] William M. Wiltshire . (2018) SAT-MOD-2018110800083. [Online]. Available: https://fcc.report/IBFS/SAT-MOD-20181108-00083

[24] StarLink-Insider Team. (2023) Starlink Ground Station Locations: An Overview. [Online]. Available: https://starlinkinsider.com/starlink-gateway-locations/

[25] HawkEye 360, Inc. (2019) SAT-LOA-20190102-00001. [Online]. Available: https://fcc.report/IBFS/SAT-LOA-20190617-00048

[26] Kreiner, EJ and CaJacob, Dan. (2018) GRCon18 - GNU Radio and RFNoC in Space: How Hawkeye 360 uses GNU Radio on Small-Satellites. [Online]. Available: https://www.youtube.com/watch?v=WlXi2UyOL

[27] CCSDS Contributors. (2015) SPACE DATA LINK SECURITY PROTOCOL (Historical Document). [Online]. Available: https://public.ccsds.org/Pubs/355x0b1s.pdf

[28] T. Scharnowski, F. Buchmann, S. Wörner, and T. Holz, "A case study on fuzzing satellite firmware," in *Workshop on the Security of Space and Satellite Systems (SpaceSec)*, 2023.

[29] G. Falco, "The Vacuum of Space Cyber Security," in *AIAA SPACE and Astronautics Forum and Exposition*. American Institute of Aeronautics and Astronautics, 2018.

[30] D. Livingstone and P. Lewis, *Space, the Final Frontier for Cybersecurity?* Chatham House. The Royal Institute of International Affairs, 2016.

[31] G. Falco, A. Viswanathan, and A. Santangelo, "CubeSat Security Attack Tree Analysis," in *IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT)*, 2021.

[32] M. Manulis, C. P. Bridges, R. Harrison, V. Sekar, and A. Davis, "Cyber Security in New Space," *International Journal of Information Security*, 2020.

[33] T. Harrison, K. Johnson, and T. G. Roberts, *Space Threat Assessment 2019*. Center for Strategic & International Studies., 2019.

[34] T. Harrison, K. Johnson, T. G. Roberts, T. Way, and M. Young, *Spacethreat Assessment 2020*. Center for Strategic and International Studies, 2020.

[35] D. P. Fidler, "Cybersecurity and the New Era of Space Activities," *Digital and Cyberspace Policy Program*, 2018.

[36] D. Barnard-Wills and D. Ashenden, "Securing Virtual Space: Cyber War, Cyber Terror, and Risk," *Space and Culture*, 2012.

[37] G. Falco, "Job One for Space Force: Space Asset Cybersecurity," *Belfer Center for Science and International Affairs, Harvard Kennedy School*, 2018.

[38] L. Yang, X. Cao, and J. Li, "A New Cyber Security Risk Eevaluation Method for Oil and Gas SCADA based on Factor State Space," *Chaos, Solitons & Fractals*, 2016.

[39] Aerospace Corporation. (2023) Space Attack Research & Tactic Analysis (SPARTA). [Online]. Available: https://sparta.aerospace.org/

[40] European Space Agency. (2023) SPACE-SHIELD. [Online]. Available: https://spaceshield.esa.int/

BIOGRAPHY

Johannes Willbold *received his B.Sc. and M.Sc. from the Ruhr University Bochum in Germany in 2018 and 2020, respectively. He is currently a doctoral student in the systems security group, working on space and satellite systems security. His work focuses on firmware security aspects of space systems, with a recent research paper at the 44th IEEE Symposium on Security and Privacy (S&P) presenting a security analysis of LEO satellites. As subgroup chair, he is also working on transferring recent academic advances into the IEEE Standard for Space System Cybersecurity (S2CY).*

Franklyn Sciberras *earned his B.Sc. in Computer Science with Summa Cum Laude from the University of Malta in 2019. He went on to cultivate his skills as a Software Engineer in the United States, contributing to the field of Application Security Testing. In 2021, Franklyn pivoted back to academia, where he is currently finishing a joint M.Sc. in Cybersecurity at both ETH Zurich and EPFL. Franklyn's research interests encompass a diverse range of topics, including blockchain technologies, mobile and wireless network security. In his current research he is focusing on malware analysis and its intricate intersections with social vectors.*

Martin Strohmeier *is a Senior Scientific Project Manager at the Cyber-Defence Campus in Switzerland and a Visiting Fellow at the University of Oxford. Before coming to Oxford for his PhD, he received his MSc from TU Kaiserslautern, Germany and worked as a researcher at Lancaster University's InfoLab21 and Lufthansa. He is interested in wireless security, critical infrastructures and adversarial machine learning. Martin is also a co-founder and board member of the OpenSky Network.*

Vincent Lenders *is Director of the Cyber-Defence Campus and Head of the Cyber Security and Data Science Business Unit at the Science and Technology branch of the Swiss Federal Department of Defence. He holds a Masters and a PhD in Electrical Engineering and Information Technologies from ETH Zurich and was Postdoctoral Research Fellow at Princeton University. His research interests lay at the intersection between cyber security, data science, networking, and crowdsourcing. He has published over 150 papers in these areas and is a co-founder of the OpenSky Network.*