Don’t Look Up: There Are Sensitive Internal Links in the Clear on GEO Satellites

Wenyi Morty Zhang
wez049@ucsd.edu
University of California, San Diego
La Jolla, USA

Annie Dai
anniedai@umd.edu
University of Maryland
College Park, USA

Keegan Ryan
kryan@ucsd.edu
University of California, San Diego
La Jolla, USA

Dave Levin
dml@umd.edu
University of Maryland
College Park, USA

Nadia Heninger
nadiah@ucsd.edu
University of California, San Diego
La Jolla, USA

Aaron Schulman
schulman@ucsd.edu
University of California, San Diego
La Jolla, USA

Abstract

Geosynchronous (GEO) satellite links provide IP backhaul to remote critical infrastructure for utilities, telecom, government, military, and commercial users. To date, academic studies of GEO infrastructure have focused on a handful of satellites and specific use cases. We perform the first broad scan of IP traffic on 39 GEO satellites across 25 distinct longitudes with 411 transponders using consumer-grade equipment. We overcome the poor signal quality plaguing prior work and build the first general parser that can handle the diverse protocols in use by heterogeneous endpoints. We found 50% of GEO links contained cleartext IP traffic; while link-layer encryption has been standard practice in satellite TV for decades, IP links typically lacked encryption at both the link and network layers. This gives us a unique view into the internal network security practices of these organizations. We observed unencrypted cellular backhaul traffic from several providers including cleartext call and text contents, job scheduling and industrial control systems for utility infrastructure, military asset tracking, inventory management for global retail stores, and in-flight wifi.

Paper Version:
Publication date of this paper: October 13, 2025.
This is an expanded version of the conference publication
Wenyi Morty Zhang, Annie Dai, Keegan Ryan, Dave Levin, Nadia Heninger, and Aaron Schulman. 2025. Don’t Look Up: There Are Sensitive Internal Links in the Clear on GEO Satellites. In *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security (CCS ’25), October 13–17, 2025, Taipei, Taiwan.* ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3719027.3765198

1 Introduction

For decades, geostationary (GEO) satellites have been the primary means of delivering reliable high-speed communication to remote sites. GEO links serve a variety of uses including television and Internet Protocol (IP) communication, including Internet access for in-flight WiFi and residential Internet [32, 69], as well as backhaul for private internal networks for sensitive remote commercial and military equipment [53, 58, 59].

There are thousands of GEO network links in operation today, carried by 590 GEO satellites orbiting Earth [68]. Each satellite may carry traffic for dozens of independent networks through an array of on-board transponders, each covering a diameter of thousands of kilometers (at most a third of Earth’s surface) [36, 54]. GEO IP links are established by leasing time on a transponder and aiming dishes for Earth-based terminals and hubs at that transponder [16]. The ecosystem of equipment to support IP-based GEO links is mature and heterogeneous: at least 10 different vendors sell terminal and hub systems that each use their own proprietary protocol stacks to provide GEO networking [60].

Unfortunately, GEO satellites have been shown to be particularly susceptible to interception attacks [5, 39–41]. Consumer-grade satellite dishes and passive terminals are *Commercially Available Off-The-Shelf* (COTS) for hundreds of US dollars; a network of online enthusiasts publishes open databases of satellite coordinates and transponders [57], and the popularity of satellite television has given rise to high-quality free software for finding and decoding GEO satellite signals [12]. Given that any individual with a clear view of the sky and US$600 can set up their own GEO interception station from Earth, one would expect that GEO satellite links carrying sensitive commercial and government network traffic would use standardized link and/or network layer encryption to prevent eavesdroppers [10, 61]. Indeed, encryption has been routinely deployed for the last four decades to protect paid satellite television services from piracy [54]; a succession of satellite

Industry	Cleartext Data
Cell Backhaul	IMS (Call Audio and SMS), Encryption Keys, IMSIs
Telecom	Call Audio and Metadata
Military	Vessel Tracker, Call Metadata
Retail	Inventory, Internal Communications
Power grid	Repairs, Grid Monitoring
Banking	LDAP, ATM Traffic
Aviation	Entertainment Audio, Tail Numbers

Table 1: Unencrypted data we observed on GEO satellites.

CCS ’25, Taipei, Taiwan

ACM ISBN 979-8-4007-1525-9/2025/10
https://doi.org/10.1145/3719027.3765198

content-scrambling algorithms has been subjected to significant real-world scrutiny [11, 17, 20, 25, 42, 56, 70].

Prior work has found sensitive communication in the clear over a select handful of GEO satellites and transponders [5, 39, 40], particularly in specific segments of the GEO satellite market such as marine [39] and in-flight WiFi [28]. However, these prior studies focused on restricted protocols and use cases of the GEO ecosystem, and provide a limited view into the threat posed by GEO network interception. Indeed, prior work even stated that "generalizations applicable to the entire VSAT industry are difficult if not impossible" because "service operators use a wide range of protocols, many proprietary and undocumented" [39].

Threat Model. In this work, we demonstrate the feasibility of an attacker whose goal is to observe satellite traffic visible from their position by *passively scanning as many GEO transmissions from a single vantage point on Earth as possible.* This form of wide-scale interception has previously been assumed to only be feasible with state actor-grade equipment and software [29]. More precisely, we demonstrate that a *low-resource* attacker, using COTS, low-cost equipment can reliably intercept and decode hundreds of links from a single vantage point.

Our Work. We undertake what we believe is the most comprehensive study of GEO satellites, transponders, protocol stacks, encryption use, and application domains carried out to date in the open research community. We observe that while content scrambling is standard for satellite TV, it is *surprisingly unlikely* to be used for private networks using GEO satellite to backhaul IP network traffic from remote areas. Many organizations appear to treat satellite as any other internal link in their private networks. Our study provides concrete evidence that network-layer encryption protocols like IPSec are far from standard on internal networks, unlike on the Internet where TLS is default [44], a finding that has been until now essentially impossible for external researchers to legally measure.

Findings. As a consequence, we observe significant amounts of highly sensitive internal network traffic being broadcast unencrypted to large portions of North America. The severity of our findings suggest that these organizations do not routinely monitor the security of their own satellite communication links. Despite the availability of COTS satellite hardware, scanning and capturing arbitrary GEO network traffic is technically challenging.

Challenges. Capturing traffic across the broad range of GEO satellites and users requires overcoming three challenges: (1) Dish alignment: We require a passive ground station that can be automatically aimed at dozens of satellites with sufficient accuracy and signal quality to ensure correct datastreams [34]. (2) Protocol diversity: Assessing the GEO ecosystem requires understanding a heterogeneous puzzle of satellite protocol layers and proprietary implementation quirks in order to re-assemble data streams into IP packets and accurately parse higher-layer protocols for traffic analysis and attribution [8, 14, 18, 19, 19, 38]. (3) Link churn: A given transponder may lease service for many different links over time [16, 54]. In order to accurately audit traffic flows, we must account for unpredictable user changeover during our scans.

Our contributions. We overcome these challenges to build a universal GEO satellite scanner from low-cost COTS satellite equipment that can scan the sky for visible satellites, scan each for available transponders, then accurately decode IP packets from each transponder. Our technical contributions include:

(1) We introduce a new method to self-align a motorized dish to improve signal quality. Specifically, we could receive IP traffic from *14.3% of all global Ku-band satellites* from a single location with high signal quality and low error rate.
(2) We developed a general GEO traffic parser that can blindly decode IP packets from seven different protocol stacks that we observed in our scans. Five of these stacks have never been reported in any public research we are aware of.

Using our tool, we then carried out an empirical study of the current state of encryption in the GEO satellite ecosystem.

(3) We collected comprehensive scans across a seven-month period. There is enough stability in transponder usage across a 24-hour time interval that we can reliably scan each of the 411 transponders that were visible from our test location and collect a 3-minute data capture from each over the course of a scan that took approximately 24 hours.
(4) Our scanning dataset uncovered a surprising number of private IP networks with cleartext traffic from industry, government, and critical infrastructure. Table 1 summarizes some of the types of traffic we observed and disclosed.

Our findings illustrate ongoing structural disincentives for deploying encryption, even in apparently security-critical use cases. Source code and an up to date full version of this paper [71] can be found at:

`https://satcom.sysnet.ucsd.edu`

1.1 Ethical Considerations

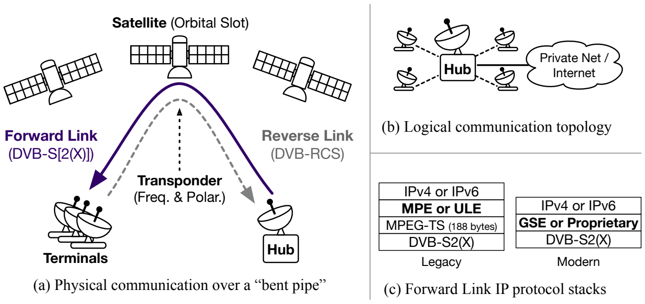
We cleared our experimental design and potential legal concerns with our organization's legal counsel. Our IRB has determined that this project is exempt from IRB review and human subjects consent because it is secondary research on existing public data.

We stored data on a protected machine. We separately encrypted the data files that contain unencrypted communications and have deleted sensitive data at the request of vendors.

When we unexpectedly discovered unencrypted voice and SMS communications in our data, we ceased collection on those transponders, encrypted the relevant data, and consulted again with our lawyers, who helped facilitate disclosure with affected vendors.

1.2 Disclosure

We undertook an extensive, best-effort disclosure process that included guessing security contacts, exercising our professional and LinkedIn networks, and declining bug bounties with nondisclosure agreements. We disclosed to T-Mobile on December 19, 2024. The vulnerability that we found does not affect T-Mobile's new Low-Earth Orbit Starlink deployment. We disclosed to the US Military in December, 2024. We disclosed to Walmart-Mexico on January 14, 2025 and had in-depth conversations with them. We disclosed to AT&T on February 10, 2025. We disclosed the vulnerabilities that affected the Mexico government, TelMex, Grupo Santander



(a) Physical communication over a "bent pipe"

(b) Logical communication topology

(c) Forward Link IP protocol stacks

Figure 1: Overview of the GEO satellite data ecosystem.

Mexico, Banjército, and Banorte to CERT-MX on April 4, 2025. We also reached out separately to Grupo Santander on July 10, 2025. We attempted numerous avenues to get in contact with TelMex separately. We disclosed to Intelsat on May 11, 2025. We disclosed to Panasonic Avionics on May 12, 2025. We disclosed to WiBo on July 8, 2025. We disclosed to KPU on July 20, 2025 and had in-depth conversations with them; they are working with affected customers to enable encryption where possible.

2 Background

We provide a brief background primer on GEO satellite networks, focusing on the aspects most relevant to interception.

GEO Satellites. Satellites can be broadly classified by their orbits. For example, Starlink satellites have a Low Earth Orbit (LEO). In this paper, we focus on geostationary (GEO) satellites. Hundreds of GEO satellites orbit Earth around the equator at a fixed altitude and position relative to Earth. Each GEO satellite can be distinguished by its fixed longitudinal position, called its *orbital slot* (Figure 1(a) top). With proper coordination to avoid collisions, multiple GEO satellites can "share" an orbital slot by operating closer than 0.1° apart.

For the most part, GEO satellites act as basic repeaters, receiving signals from the ground, and amplifying and repeating them to broadcast them to a larger coverage area. This basic communication path is often referred to as a "bent pipe" (Figure 1(a)).

Communicating with a satellite requires a directional dish antenna that focuses its beam on one satellite in one orbital slot. When transmitting and receiving, satellites must avoid signal interference. Satellites in distant orbital slots can reuse the same frequency channels, but those in nearby orbital slots must use different frequencies.

Transponders. GEO satellites operate multiple simultaneous amplifying relays, called transponders (Figure 1(a)). Each satellite has dozens of transponders covering different frequencies (GHz), polarizations (Horizontal/Vertical), and coverage areas. GEO transponders are distinguished by the spectrum bands that they transmit over: commonly Ku, Ka, or C. We focus on Ku-band, which is traditionally used for television and Internet applications. Satellite transponders can use beam widths of different sizes in order to transmit to different sized regions on the ground; the area covered by a transponder's beam is called its footprint.

From our single vantage point in La Jolla, CA, we observed 411 independent Ku-band transponders.

Vendor	Phy Layer Protocol	Link Layer Protocol	Link Layer Crypto	Net Layer Crypto
Gilat	DVB-S2(X)	SkyEdge	☒	☒
Hughes	DVB-S2(X)	HX/Jupiter	☒	☒
iDirect	DVB-S2(X)	iDirect	☒	☐
Comtech	DVB-S2(X)	Heights	☒	☐
ND SatCom	DVB-S2(X)	SKYWAN	☒	☐
STM	DVB-S2	SatLink	☒	☐
Newtec	DVB-S2(X)	Dialog	☐	☒
Advantech	DVB-S2(X)	SatNet	☐	☒
Viasat	DVB-S2(X)	ArcLight	☒	☐

Table 2: Popular satellite terminals' protocols and cryptographic capabilities. Despite having cryptographic capabilities, we find that they are not always used.

A customer can set up a satellite link by leasing time on a transponder from the satellite's owner that covers the area where the remote terminals will be deployed. The customer then configures equipment on either end of the link to use that transponder and aligns their antenna to the satellite.

Terminals and Hubs. There are two broad classes of terrestrial end-devices that communicate with GEO satellites. *Terminals* are typically located in remote areas with no access to the Internet except through the satellite link. Terminals that communicate via a single leased transponder are always the same vendor and model. *Hubs* are ground stations that may have access to the Internet or a private network that terminals communicate with via satellite. These devices form a logical star topology, in which all of the terminals' communication is done through their hub (Figure 1(b)).

Forward Link Protocols. Each direction between the terminals and hub is an independent link with its own protocols (Figure 1(a)). The link used to broadcast traffic from the hub to the remote terminals is called the *forward link* (hub-to-terminal). The forward link covers a large area, often a diameter of ∼10,000 kilometers, to send traffic to many remote terminals. The coverage of the *reverse link* (terminal-to-hub) has a comparatively smaller area, just to cover the hub.

Our work intercepts the forward link, so we describe it here. The physical layer protocols in common use today are DVB-S and DVB-S2(X) (Figure 1(c)). DVB-S is a legacy video/audio broadcast protocol that was adapted for use with IP, and DVB-S2(X) is a more modern protocol designed for carrying IP reliably and efficiently.

Different terminal vendors use different link-layer protocols to encapsulate IP packets from frames at the physical layer. Table 2 provides a list of the most popular terminal vendors and their protocols (in the North American market). These include standardized protocols, modified versions of standard protocols, and vendor-specific proprietary protocols. Legacy IP links were designed to operate over terminals that were originally intended primarily for video delivery and thus use MPE encapsulation, while newer IP links operate over newer, more efficient standards like GSE. There is limited public documentation on the proprietary protocols and implementations used by vendors.

Encryption. Over-the-air encryption is supported by most satellite terminal and hub systems, as shown in Table 2. Encryption can

Author (Year)	Use Cases	Protocol	# Sat	# Trans	Challenges	Region
Pavur et al. (2019) [40]	General Internet	MPE	14	13	Poor signal	Europe
Pavur et al. (2020) [39]	Marine Internet	GSE	2	2	No GSE parser, poor signal	Europe
Baselet et al. (2022) [28]	Aviation Internet	GSE	18	34	–	Central Europe
Lin et al. (2023) [47]	No Analysis	GSE	7	–	Low signal	Asia
Our work (2025)	Private Networks	Many	**39**	**411**	Poor signal, many protocols	North America

Table 3: Our methods allow us to scan many more transponders than prior GEO interception papers.

be implemented in terminals at the physical, link, or network layer. Using the terminals' built-in encryption saves bandwidth because the terminal can do header and payload compression. The DVB-S2(X) physical layer also offers a scrambling mode (for wireless transmission integrity) that can be set in the terminal [22, 54].[1]

3 Related Work

To our knowledge, our threat model of using low-cost consumer-grade satellite equipment to comprehensively survey GEO satellite usage has not been explored before in the academic literature.

On the high-resource end, military and intelligence agencies have the capabilities to passively capture huge swaths of the satellite ecosystem. Indeed, there are several companies who publicly sell proprietary GEO satellite survey systems to government customers at relatively high prices. This high-end equipment has the capability of parsing traffic from many terminal/hub ecosystems. One vendor claims to be able to decrypt traffic for the Hughes ecosystem [63].

On the low-resource end of the attacker spectrum, a handful of academic works studied a similar adversary to us but limited their scope to a subset of protocols and marine and aviation applications. Table 3 compares our study to prior work.

Pavur, Moser, Lenders, and Martinovic [40] collected data from 13 DVB-S transponders in Europe in 2019 using consumer-grade equipment. They found unencrypted consumer Internet traffic and utility infrastructure. They focused only on terminals/hubs using the MPE protocol, as that was the only public decoder that was available at the time. In 2020, Pavur, Moser, Strohmeier, Lenders, and Martinovic [39] analyzed marine Internet access traffic for two transponders from two targeted satellites in Europe. For this work, they built a parser to extract IP traffic from DVB-S2 encapsulated GSE streams, focusing on this protocol stack because it was common in maritime applications. Both works struggled with poor signal quality to decode traffic from different transponders.

In 2022, Baselt, Strohmeier, Pavur, Lenders, and Martinovic [28] performed a more comprehensive study of aviation-related traffic using the GSE protocol in Europe. This work provided the first scanning results on the GEO satellite ecosystem. The authors scanned 18 visible satellites and found 34 transponders that had GSE traffic that they could parse that were related to aviation Internet access.

Lin, Cheng, Luo, and Chen introduced a new machine-learning GSE decoding algorithm in 2023 to overcome the challenge of low signal quality in intercepted GEO satellite feeds [47]. They used a $15,000 high-end dual-axis steerable satellite setup, and evaluated their data capture on data from seven satellites in Asia. Their work is complementary to ours as CLExtract could improve our interception of GSE streams. However, even without their low-signal decoding, our comprehensive scanning was able to capture 196 GSE transponders with high rates of successful traffic interception.

Various other threats have been identified in the satellite communications ecosystem, including terminals being vulnerable to injection attacks [9] and fingerprinting [64].

In contrast to these prior works, we focus on carrying out a comprehensive wide-band, many-satellite, long-duration scan of the Ku-band satellite ecosystem and understanding vendor-specific quirks that allow us to accurately reconstruct internal traffic flows that no prior works have documented.

Achieving low-cost, near-comprehensive scans of GEO satellites comes with considerable challenges. Most prior works struggled with low signal quality, which we believe stems from the difficulty of aligning consumer-grade satellite equipment. We overcame this challenge by developing careful alignment methods that allow us to accurately gather raw data from hundreds of transponders. We describe our methods in Section 4. The second challenge is to black-box reverse-engineer and parse the diverse collection of protocols at the physical, link, and network layer, and implementation choices made by vendors. We describe our methods in Section 5. Our more comprehensive viewpoint allows us to accurately reconstruct traffic and gain visibility into satellite ecosystems that were not parseable by the toolchains used in prior work.

4 Scanning Data Collection

This section describes our methodology for scanning and capturing raw data streams from Ku-band GEO satellites with low-cost COTS components (Figure 2). This setup allows scanning many Ku-band satellites (e.g., between 57.2° W and 177.2° W) over a long period of time. We describe two key features—motorized arc traversal, and raw signal capture—in detail.

As noted above, all data collected and analyzed in this work is from the forward link. The narrower beam coverage of reverse links makes them inaccessible from a single vantage point.

4.1 Hardware Setup

110cm Ku-Band Satellite Dish with Mount ($180): A standard offset parabolic dish commonly used for reception of residential satellite TV. The diameter provides sufficient gain for GEO signal reception even at the edge of a transponder's footprint [48].

[1] Reports suggest that the default key of "0" [55] is common. Such contents would have high entropy and appear encrypted. It is possible to brute force scrambling keys, but it is time consuming and could take multiple days per transponder [13].

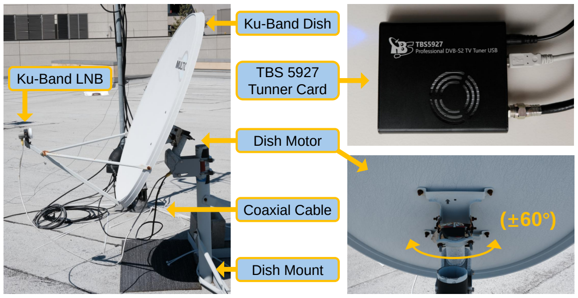


Figure 2: Low-cost GEO satellite scanning hardware setup.

Universal Ku-Band LNB (10.7–12.75 GHz) ($15): A low-noise block downconverter covering the full Ku-band downlink range. It supports H/V linear polarization and has a 0.1 dB noise figure [6].

DiSEqC 1.2-Compatible Dish Motor (± 60°) ($195): A consumer-grade single-axis rotor controlled over coaxial cable via DiSEqC 1.2 commands, for hobbyists desiring access to multiple TV satellites from one dish. With proper alignment (Section 4.3), it allows automated scanning across the GEO plane from a single fixed dish [7].

TBS-5927 DVB-S/S2 USB Tuner Card ($230): A high-performance tuner supporting blind scan, signal locking, and raw capture. It demodulates DVB-S/S2 signals over the 950–2150 MHz IF band, providing full Ku-band coverage when paired with the LNB. The tuner card allows software control of the dish motor [67].

Miscellaneous Components ($30): Coaxial cables, connectors, power inserters, and crimping tools.

4.2 Data Collection

The workflow for scanning with this low-cost setup is as follows:

(1) **Dish Alignment and Aiming**: We set up the dish and perform fine alignment to match the GEO plane, calibrating using specific transponders. We use the DiSEqC 1.2 command [26] to remotely operate the dish motor.

(2) **Blind Scan**: We perform a blind scan across the Ku-band (10.7–12.75 GHz), sweeping symbol rates from 100–70,000 KS/s and both horizontal and vertical polarizations to detect active transponders. We store the frequency, symbol rate, and polarization of detected transponders in a structured database.

(3) **Raw Data Capture**: We iteratively lock onto transponders from the database and record data of the desired duration to a raw data file, bypassing the capture software's automatic filters.

4.3 Dish Alignment and Aiming

Imperfect satellite alignment can result in poor signal quality and signal loss.

Prior to this work, achieving accurate alignment required an expensive dual-axis steerable dish or phased-array steerable antenna [50]. We demonstrate that with hidden alignment data in satellite feeds, even a hobbyist-grade single axis rotor dish can achieve good enough alignment to receive high-quality signal from all visible satellites. There are two components to aligning a single-axis rotor: vertical (elevation) and horizontal (azimuth) alignment.

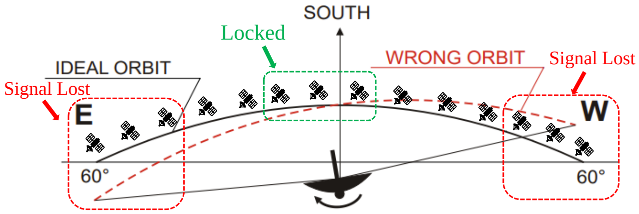


Figure 3: Example of single-axis rotor dish misalignment (adapted from [66] Figure A1).

Vertical alignment is straightforward: any error in elevation will shift the entire elliptical arc above or below the GEO plane. This reduces signal strength uniformly across all satellites. In our experiments a misalignment of just over 1° reduced SNR by more than 5dB, despite using a large 110 cm dish.

Horizontal (azimuth) alignment is significantly harder. The motor's 0° reference must be precisely oriented toward the user's geographic longitude, but magnetic compasses and mobile apps are unreliable in urban environments. The primary limitation of a single-axis rotor is that the dish must sweep across the satellites in the GEO plane along an elliptical arc. Even small azimuth misalignments can lead to partial coverage: satellites near the center of the arc may still lock, but those at the edges are lost (Figure 3). We found manual alignment via compass-based methods to be effectively impossible for our needs.

Instead, we used ground-truth satellite positions to calibrate alignment. Although it is possible to detect a satellite via a spike in signal power, identifying said satellite without access to the owner's proprietary operational parameters is more difficult.

Hidden ground-truth alignment references. We identified two sources of ground truth to calibrate our azimuth alignment: (1) some transponders broadcast satellite-specific metadata such as the orbital position (e.g., 99° W), satellite name (e.g., G16 for Galaxy 16), and transponder ID (e.g., BEAM 0067) within the Network Information Table (NIT) and Service Description Table (SDT) fields of DVB-SI tables [23]; (2) public databases of free-to-air television (e.g., LyngSat [45]) identify known satellite transponders. These anchors span both edge and central positions in the GEO plane, enabling precise alignment across the dish's full motor range.

Evaluation of Alignment. In our capture location, with a manually misaligned dish, we observed that despite strong signal reception from satellites between 129.0° W and 87.1° W (a span of 15°), all satellites east of 87.1° W (10°) failed to lock due to arc misalignment. We identified twelve transponders across eight longitudes ranging from 65.0° W to 129.0° W (see Appendix A) that broadcast recognizable identifiers. After adjusting the horizontal alignment of the motor to agree with the ground truth latitude of the satellites, we gained 10° of visibility, adding 14 new satellites and 110 transponders to our scan.

4.4 Blind Scanning

We use the freely available EBSPro [12] (Easy BlindScan Pro) software to blind scan for active transponders. After alignment, the scanner sweeps the full Ku-band frequency range from 10700 to 12750 MHz across both horizontal and vertical polarizations. We set the symbol rate range from 100 to 70,000 KS/s, constrained by the capabilities of the STMicroelectronics STV0910 demodulator used in the TBS-5927 tuner card. To ensure comprehensive coverage and avoid missing any signals, we set the frequency step to 1 MHz and the symbol rate step to 1 KS/s during the scan. A transponder is detected when the signal exceeds the minimum SNR threshold required for lock-in by the TBS 5927 tuner card.

For each detected transponder, we record EBSPro's detected physical-layer attributes such as frequency, symbol rate, polarization, forward error correction (FEC), and roll-off factor.

4.5 Raw Data Capture

We automated raw data capture with `dvbv5-zap` [1]—a Linux tool compatible with our TBS tuner card's Linux driver—to iteratively lock onto transponders from the list obtained above in Section 4.4. For each transponder, the tool needs to capture the raw bits decoded from the DVB-S/S2(X) protocol into a `.ts` file so our tools can then parse the vendor specific stack to recover IP packets.

However, Linux's DVB-S support was developed for and largely used by satellite television enthusiasts, so it does not naively support recording raw data streams. We modified the tuner card's driver to record a raw transport stream at the DVB-S/S2(X) protocol layer and above instead of an MPEG-TS video/audio stream. MPEG-TS frames are prefixed with a synchronization byte `0x47` [14, 15]; when this byte is not observed, the driver assumes it has lost synchronization and discards bytes until the next available `0x47`. This approach works for an MPEG-TS video/audio stream temporarily corrupted by noise, but it fails for non-MPEG-TS streams, such as raw DVB-S2(X), which do not use `0x47` for frame synchronization.

To capture raw streams of DVB-S/S2(X) bytes, we modified and recompiled the tuner card's Linux driver to disable this MPEG-TS internal filtering and forward the raw demodulated data directly to the `demux0` device. Our modified driver is open source to support future satellite research [71].

In EBSpro, this filtering can be similarly circumvented by enabling the *Raw data handling (do not use Internal EBSPro's filter)* option before performing a manual capture. In earlier versions (prior to 18.0.0.2 RC), this feature was unavailable.

4.6 Dataset Summary

From August 16 to 23, 2024, we conducted a systematic scan of all 39 satellites visible to our receiver from our position in western North America. We captured 3–10 minutes of data from each satellite transponder, varying based on data rates.

Beyond this initial period, we have continued periodic selective scans, as well as longer recordings on specific satellites for more in-depth analysis. In total, we collected over 3.7 TB of raw data.

4.6.1 Ku-Band Satellite Coverage and Scanning Results. According to the open source satellite database SatBeams, there are 273 GEO satellites operating in the Ku-band [57]. By cross-referencing this data with the coordinates of our receiver, we limit the scope of our data collection to satellites between 55.0° W and 172.0° E. Because our motor was only capable of ±60° of freedom, we further limited our collection to satellites between 57.2° W and 177.2° W.

As shown in Figure 4, our scan successfully captured signals from satellites positioned between 61.0° W and 129.0° W, covering 39 satellites across 25 distinct longitudes. The lack of detected satellites in the western sky is not due to scanning limitations, but rather the sparse customer base over the Pacific Ocean, which leads to fewer active Ku-band satellites in that region. In total, we identified and successfully locked onto 411 Ku-band transponders, each carrying distinct services and traffic data. A detailed analysis of the captured transponder traffic is provided in Section 5.

Apparent transponder counts can be inflated by orbital slot overlap. For instance, at 105° W, we observed what appears to be an unusually high number of transponders. However, closer inspection reveals that multiple satellites share nearby orbital slots (e.g., within 0.1–0.5° separation), causing their transponders to be grouped under a single longitude. Such slot-sharing is common in GEO deployments and reflects coordinated co-location strategies [37, 54].

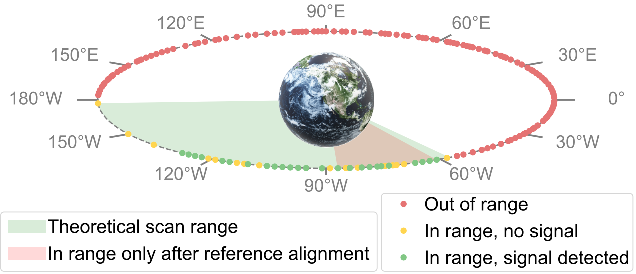


Figure 4: Coverage of Ku-Band GEO Satellites from Our Scanning Location. Each dot represents a unique GEO satellite. The green shaded region shows the theoretically achievable scanning arc; pink highlights coverage enabled by precise reference-based alignment.

4.7 Transponder Churn

Satellite transponders change over time because of expiring and activating leases. We conducted two full-band scans in August 2024 in February 2025 and compared the results. Figure 5 presents a histogram of transponder changes by longitude between scans.

The results highlight the heterogeneous and dynamic nature of real-world satellite transponder configurations. The high turnover for some satellites suggests that certain operators dynamically lease and reallocate transponder capacity in response to service contracts, demand shifts, or regional coverage strategies [21].

Interestingly, transponder churn does not always correspond with changes in physical-layer parameters. In some cases, the transmission configuration remained constant (same frequency, polarization, symbol rate), yet the protocol content changed entirely. For instance, one transponder on 67.0° W previously carried DVB-S2 → GSE → IP traffic, but was later observed transmitting an unidentified binary data format, indicating a full-service replacement while retaining the same modulation settings. This reinforces the importance of layered traffic inspection, as services cannot be inferred from transmission parameters alone.

These findings demonstrate the need for longitudinal, periodic scanning of satellite systems to maintain visibility into spectrum

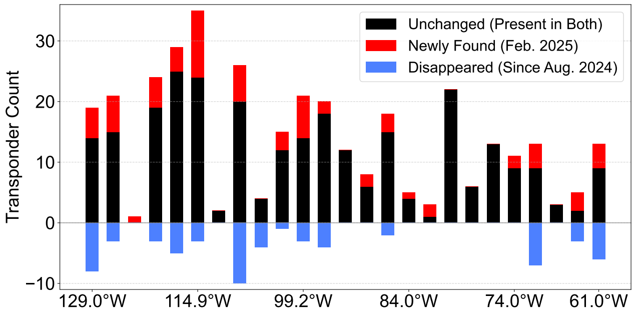


Figure 5: Per-longitude transponder changes between August 2024 and February 2025. Black bars show stable transponders present in both scans, red shows newly detected transponders, and blue shows transponders that disappeared.

usage, service rotation, and emergent threats. A static snapshot of satellite spectrum is insufficient; instead, satellite security research requires a continuous and systematic monitoring paradigm.

5 Parsing Data Captures

For each transponder identified in our scan, our tuner card decodes the physical layer and captures a raw stream of bytes. We ended up with 411 captures, each corresponding to a unique active GEO transponder visible between August 2024 and February 2025. 65 of these captures contained satellite TV; we exclude them from further analysis. This leaves 346 captures. For the parsing numbers below, we sampled 1 MB of each capture to characterize the protocol stack.

As discussed in Section 2, IP packets sent over satellite are typically encoded and encapsulated in several layers, where the protocols, their ordering, and implementation quirks may all be vendor-specific and proprietary, with scarce public documentation.

In order to empirically understand this implementation universe, we black-box reverse engineered as many of the protocol stacks as we could, developing heuristics for manual parsing. Prior work has assumed that the universe of terminals and protocols is too diverse to study the security of the ecosystem as a whole [39]. However, we show that this problem is more tractable than believed, allowing us to analyze a large fraction of the Ku-band GEO satellite ecosystem. Figure 6 provides an overview of our general GEO satellite IP packet parser and shows how it compares to parsers in prior work.

5.1 DVB-S/S2(X)

The public standardized encodings for satellite data are DVB-S, DVB-S2, and DVB-S2X [14, 19, 22]. The physical-layer decoding for these standards is performed by our tuner card, yielding a raw bytestream. In some cases, there is additional framing that must be parsed in software. For the legacy DVB-S (Digital Video Broadcasting–Satellite) standard, no further processing is required. For the DVB-S2 (DVB-S Second Generation) and DVB-S2X (DVB-S2 Extension) standards, there is a framing-layer encoding which is not handled by our tuner card and must be processed in software. The process of decoding this framing layer is complicated by implementations that deviate from the published standard [22].

In both the standard and our empirical observations, DVB-S2(X) frames consist of a 10-byte Base-Band (BB) header followed by an arbitrary length data field, which together comprise a BB frame. The BB header format is a two-byte Mode Adaptation Type (MATYPE) field, a two-byte User Packet Length (UPL) field, a two-byte Data Field Length (DFL) field, three bytes related to synchronizing decoding of user packets, and a one-byte CRC-8 checksum over the first nine bytes of the header [19].

Because a raw capture may begin in the middle of a DVB-S2 frame, our parser must be able to distinguish the beginning of valid BB headers. The CRC-8 checksum of the first nine bytes helps eliminate many candidate headers, but the short checksum length leads to many false positives. We heuristically eliminate candidates by assuming the UPL and DPL length fields are multiples of 8 bits and $0 < \text{DFL} \leq 58{,}112$ bits, the maximum specified length of the data field [19]. Next, if a candidate BB header passes these checks, we expect that it is followed by DFL bits of data field, some padding, and the next BB header. If the next BB header passes the heuristic checks, we have probably identified a true BB header and continue parsing. If it does not, the header was a false positive; we discard bytes until we heuristically identify another candidate header.

This synchronization approach is fast and reliable. Since our parser discards bytes that do not pass the heuristic checks, we can use the rate of discarded bytes to identify which captures use DVB-S2(X) framing. If our parser discards more than 10% of the capture, that capture is either not DVB-S2(X) or there is too much transmission noise to decode properly. The remaining captures are classified as DVB-S2(X). Of the 346 captures we analyzed, we identified 290 as using DVB-S2(X) framing. Of these, the median rate of discarded bytes was 0.1% and the mean was 0.2%.

5.1.1 Comparison with prior approaches. Pavur et al.'s GSExtract tool [39] is designed to parse DVB-S2 satellite traffic for maritime applications. They observed that the MATYPE often had the value 0x4200, so their tool scans for this 16-bit value to find the beginning of DVB-S2 frames. They also observed that the value was occasionally 0x4300, although their code does not scan for it.

While our study confirms that 0x4200 is the most commonly observed MATYPE (accounting for 90.5% of our recovered MATYPE values), we observe that their parsing approach is fundamentally flawed for a large number of transponders. This is because GSExtract implicitly assumes that all DVB-S2 frames within a transponder share the same MATYPE, and hardcodes this value to 0x4200. In reality, only 185 (64%) of our 290 DVB-S2(X) captures satisfy these assumptions. We also observed that there were 63 captures that used more than one MATYPE, meaning that even if the source code of GSExtract were modified to scan for a different 16-bit value, it would still miss DVB-S2 packets for 22% of the transponders. This demonstrates that our more flexible DVB-S2(X) parsing is necessary for proper recovery of real-world data.

5.1.2 Deviations from Standard DVB-S2. Our raw captures differed from the DVB-S2(X) standards in several ways.

Padding. The DVB-S2 standard describes how individual frames should be followed by 0-valued padding bits between the end of one frame's data field and the beginning of the next frame's header [19, Section 5.2.1]. This is to ensure that successive BB headers are

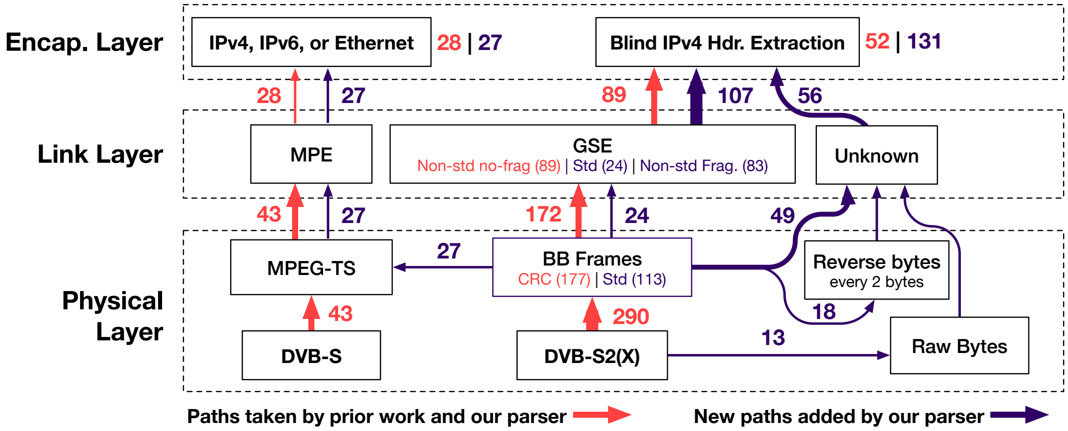


Figure 6: Comparison of our general GEO satellite parser with specific parsers from prior work. Each node represents a protocol being parsed, including different variants. Edges are annotated with the number of transponders we observed on that specific path. Values in orange represent parser flows (and transponder data) that could be handled by prior work, and values in purple represent flows that are unique to our work. In the end, our enhanced parsing leads to the full understanding and extraction of IP packets from 27 new transponders, and partial understanding and extraction of IP packets from 131 new transponders. This is a significant addition to the 28 and 52 transponders that could be fully and partially parsed by prior work.

separated by a fixed number of bits. Instead, we observe that our captures either include no padding bits between frames or a single padding byte with value 0. We noticed that there is a padding byte only when the byte-length of the data field is odd, but an odd byte-length does not always imply the presence of a padding byte.

CRC-32. Of the 296 captures with DVB-S2 as the first layer framing, 177 (61%) employ an additional CRC-32 checksum. This checksum is computed over the 80-bit header and the first DFL − 32 bits of the data field, and it is located in the final 32 bits of the data field. We reverse engineered the CRC parameters and found that the CRC polynomial is `0x104c11db7` and the initial value is `0x00000000`. These parameters are the same as the CRC-32 defined for GSE, except the initial value in that application is `0xffffffff` [18]. When available, the presence of this checksum allows us to have even greater confidence that we correctly identify DVB-S2 frame boundaries and retrieve the correct data field contents.

Pavur et al. also observed this checksum, and their GSExtract tool assumes that the last four bytes of the DVB-S2 frame are always a CRC-32 [4], but it discards the value without validating it. This means that their parser may include corrupted bytes, which validating the checksum would have prevented, or excluded valid data bytes in the case where no CRC-32 checksum is present.

SYNC Byte Discrepancy. For packetized transport or generic stream input, the SYNC field is meant to be a copy of the User Packet Sync Byte [19]. For MPEG-TS stream input encoded using DVB-S2(X), this value should be `0x47` [15]. In 27 (9%) of the transponders, despite using MPEG-TS user packets, the SYNC field was set to `0xBB`. According to the standard, this value is reserved for "user private" signaling [19].

Byte Order Reversal. In some transponders, we observed a swapped byte order within 16-bit words in the data field. We manually identified such captures in order to further decode their data fields. We observe this byte order in 18 out of the 290 DVB-S2 captures (6%).

5.2 MPEG-TS

The MPEG-TS (MPEG Transport Stream) standard describes a common method for encoding arbitrary data over an unreliable physical connection. This data encoding scheme is used with the legacy DVB-S physical framing and can also be used on top of DVB-S2(X).

MPEG-TS data consists of a series of 188-byte packets, and the first byte of the packet is a synchronization byte with fixed value `0x47` [15, 38]. Since this fixed value appears every 188 bytes, it is easy to detect MPEG-TS streams. We identify that 43 of our 346 raw captures (12%) use standard MPEG-TS encoding. In addition, we find that of the 290 captures which use DVB-S2(X) framing, 27 of these (9%) use MPEG-TS encoding at the next parsing layer.

Once we have a valid and packet-aligned MPEG-TS stream, we use existing tools like TSDuck and TShark to parse its contents [2, 3]. These tools allow us to calculate statistics about the stream and extract network packets which are encoded within the packets using multiprotocol encapsulation (MPE).

5.3 GSE

Generic Stream Encapsulation (GSE) is a common encapsulation protocol inside DVB-S2 captures [18]. Of the 290 captures with DVB-S2 as the first layer framing, 196 (68%) of them use some variant of GSE. However, these implementations of GSE all deviate from the published standard, complicating GSE parsing.

GSE encapsulates a series of variable-length packets, called protocol data units (PDUs), into a single stream of bytes [18]. GSE supports fragmentation, so a single PDU may be split across multiple GSE packets. Each GSE packet includes a 1-bit start indicator (set if the packet contents are the start of a PDU), a 1-bit end indicator (set if the contents are the end of a PDU), a 2-bit label type, and a 12-bit GSE length. When both the start and end indicators are set, the packet contains the entire PDU and no fragmentation occurred. If there is fragmentation, a 1-byte fragment identifier is used to aid in reassembling the PDU. GSE also includes a 2-byte protocol type for each PDU and a CRC-32 checksum for each fragmented PDU to validate that it was reassembled properly.

Although GSE is common among our DVB-S2 captures, we identify two major variants.

Standard GSE. Of the 290 DVB-S2 captures, 24 (8%) appear to closely follow the published GSE standard [18]. However, one important distinction is that the published CRC-32 parameters do not validate the defragmented PDUs. This meant we were unable to use the checksum to verify that defragmentation was successful, but we could still validate that the reassembled length matched the expected length and avoid improper defragmentation.

Non-standard GSE. Of the 290 DVB-S2 captures, 172 (59%) appear to follow a modified version of the standard which differs in several ways. First, the 12-bit length field *includes* the first two bytes when calculating the length, while the standard version excludes these, so all length values are off by two. Second, the fragment identifiers follow a different convention than in the standard. Instead of all 8 bits identifying the PDU, only the first 6 bits identify the PDU, and the final 2 bits are a counter representing that fragment's offset in the reassembled PDU. We call this 6/2 fragmentation. Third, GSE streams of this type include the optional DVB-S2 CRC-32 in Section 5.1, while standard GSE streams do not. Fourth, the GSE CRC-32 checksum used to validate reassembled PDUs uses the same parameters as the reverse engineered DVB-S2 CRC-32. This means that we can validate the checksum of non-standard fragmented GSE packets and are confident that our 6/2 fragmentation hypothesis is correct. Fifth, the two-byte protocol type field has a different meaning, which we discuss below. Finally, in 21 cases, it appears that several GSE PDU fragments are not received in the capture. We hypothesize this is due to channel bonding, and since our setup can only monitor a single transponder at a time, we discard any PDUs with missing fragments.

GSExtract [4] appears to decode this non-standard GSE scheme and fails to decode standard GSE. In particular, their code interprets the length field as off-by-2 and discards the last four bytes of DVB-S2 frames (although they do not validate this checksum value). It does not handle defragmentation properly, so the nonstandard 6/2 fragmentation ID, CRC parameters, or missing fragments are irrelevant to their approach.

Protocol type. Each PDU includes a two-byte protocol type. According to the standard, the protocol type either indicates the presence of optional headers as specified by the IANA registry [30] or an EtherType which indicates the type of the PDU (for example, IPv4 or IPv6) [18]. In the non-standard implementation, we observed protocol types between 0x0001 and 0x0008.

Although we do not fully understand what these protocol types mean, it appears that protocol 2 sometimes includes network traffic, protocols 1, 3, 4, 6, and 7 are too short or too repetitive to include interesting data, and protocols 5 and 8 are potentially encrypted. Although we do not know for certain that these PDUs are encrypted, protocol 5 (and 8) PDUs include 1 predictable byte (17 respectively) followed by a multiple of 16 bytes of high-entropy data. Because these PDU lengths consistently differ by multiples of the AES block cipher size, it is unlikely that the high-entropy contents are simply compressed data. Since we are focused on interpreting network traffic, we focus on protocol type 2.

5.4 Unidentified and Proprietary Protocols

Although we successfully identified most of the encapsulation formats from our captured data, there were still a number of examples where we did not fully understand the protocol stack. This includes 13 (4%) of the 346 raw captures which were not identified as DVB-S2 or MPEG-TS, 67 (23%) of the 290 DVB-S2 captures which did not use GSE, and the format of the protocol 2 (network) PDUs in GSE. Some of this may be caused by noisy collection, but in many cases the unknown bytes reflect some as-yet unidentified protocol.

For some protocol layers, we were able to identify unencrypted IP traffic within the unidentified encapsulating protocol using heuristic assumptions. If the entire IP packet appears uninterrupted and unfragmented in the mystery stream, then finding a valid IP header will also recover the IP body of that packet; any bytes not belonging to IP packets are ignored. Although this blind IP header recovery could be applied to any byte stream, we parse MPE and GSE first, since both schemes support IP packet fragmentation [18, 24] that would cause naive application of this approach to fail.

To detect IPv4 and IPv6 packets within a stream of bytes of unknown format, we progressively apply heuristic checks for IP headers. For IPv4, the header must have the correct four-bit version, a reasonable length, and a correct 16-bit checksum. IPv6 headers must have the correct four-bit version, a reasonable length, and a common public or private 16-bit IPv6 address prefix as either the source or destination.

False positives are possible with this approach, but for random data we expect it to be less than 0.001%. If there are errors, it is more likely to be because the IP packet bytes are not continuous. That is, we expect to correctly find almost all of the IP headers, and thus the correct number of IP packets, but there may be some errors when recovering the data. Based on our manual analysis of the resulting IP traffic, these errors appear to be uncommon. We ran the IP extractor on the 80 unidentified DVB-S2 and raw capture traffic, and although the bottom decile recognized less than 1.3% of the bytes as belonging to IP packets, the top decile recognized more than 57.2% of unidentified bytes as belonging to IP packets.

We note that GSExtract also extracts IP packets without understanding all surrounding bytes, but it just assumes the IP packet begins at a fixed offset rather than scanning for headers.

5.5 Full comparison with GSExtract

The GSExtract parsing tool developed by Pavur et al. [39] is the most sophisticated existing public tool for extracting IP traffic from satellite captures. Designed for parsing maritime satellite traffic, it extracts IP packets encapsulated using a specific variant of GSE and DVBS-2 with MATYPE 0x4200. Many of the individual design choices of this tool prevent it from successfully parsing generic traffic. We compared the performance of GSExtract against our parser on the 1MB samples from our 346 captures.

GSExtract only recovered IP packets from 52 of these transponders (15%). Among these captures, GSExtract recovered 26,421 IP packets. In contrast, our multilevel parser recovered IP packets from 238 captures (69%), totalling 192,624 packets, or an increase of over 600%. Different transponders naturally vary in the amount of IP traffic and size of IP packets, but we find that, regardless of the characteristics of the transponder, our parser recovers more IP packets more often. This is depicted by the CDF plot in Figure 7.

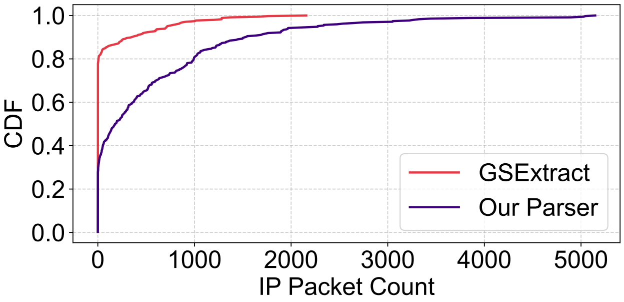


Figure 7: Cumulative distribution function (CDF) of IP packet counts extracted per transponder using our custom parser (orange) versus GSExtract (blue). GSExtract recovered no IP packets from 85% of captures, while our parser recovered IP packets from all but 31% of captures.

5.6 Encrypted and Unencrypted Traffic

After obtaining the raw satellite byte streams, we applied our multistage process to determine if and where encryption was present in the protocol stack. Our custom parser attempted to reconstruct protocol headers from the lowest possible layer upward, identifying the highest unencrypted layer for each capture (Figure 6).

When traffic was unencrypted above the IP layer, the parser successfully reconstructed complete IP packets, which we exported as PCAP files for manual inspection in Wireshark. In such cases, transport- and application-layer protocols (e.g., HTTP, SIP, DNS) were visible in plaintext. Conversely, when network-layer encryption was present, only IPsec or TLS headers were recoverable. Manual PCAP analysis was feasible for our dataset size, though the triage process could be automated for larger collections.

For encryption or obfuscation below the IP layer—such as RF scrambling or link-layer compression—we examined protocol flags, payload lengths, and any recoverable fragments of valid IP packets. In partially recoverable cases, we used a custom IP extraction script to reconstruct identifiable headers. In addition, we applied standard tools such as Binwalk to estimate payload entropy and locate unencrypted regions within the raw captures. These steps, combined with iterative reverse engineering of encapsulation layers, allowed us to identify and characterize unencrypted traffic in mixed or proprietary link-layer environments.

6 Case Studies

After parsing the outer network layers and recovering unencrypted IP packets, we manually analyzed the application-layer data to identify use cases and inform vendors where the unencrypted traffic constituted a vulnerability.

6.1 Cellular Network Backhaul

In cellular networks, satellite backhaul is common to connect remote cell towers to the core network, transmitting control plane and user data traffic like voice calls, SMS, and Internet traffic [51].

We observed unencrypted cellular backhaul traffic from multiple telecommunications providers with multiple tower connections per provider. Since the relevant protocols are rarely discussed in public security research, we first provide a primer on what each of these protocols does in a cellular network.

Cellular backhaul takes place over the IPv4-based GPRS Tunneling Protocol (GTP), which is commonly used to transport control and user traffic between cell towers over private network links back to the core network (e.g., PGW, SGW, or UPF in 5G) [52].

The traffic we found flowing unencrypted over these links included the following cellular-specific control and data traffic:

IP Multimedia System (IMS) is a standardized framework for VoIP-based communications over cellular networks that was introduced in 4G. IMS system traffic includes *control plane messages* to register phone numbers with the system, and set up/tear down calls, and *user data* such as VoIP voice calls, GMS SMS messages, and MMS and RCS messages which include photos and videos. IMS relies on SIP (Session Initiation Protocol) for signaling and enabling communication between devices and IMS core network elements. Call audio data is transmitted via RTP (Real-time Transport Protocol).

S1-U tunnels raw user Internet data over private networks between a cell tower and the cellular core network. This traffic includes common protocols like DNS, UDP, QUIC, and TLS.

S1-C carries control plane traffic with signaling protocols between cell towers (i.e., eNodeBs in the Radio Access Network) and the core network (e.g., the MME). All S1-C traffic is generally carried over SCTP (Stream Control Transmission Protocol) that ensures reliable in-order delivery. Specific S1-C protocols include:

S1 Application Protocol (S1AP) is a signaling protocol in 4G networks that facilitates communication between the eNodeB (base station) and the MME (Mobility Management Entity). S1AP messages perform functions such as establishing context about cell phones, reporting protocol errors, paging, carrying encrypted NAS signaling, and releasing UE context from the network.

EUTRAN X2 Application Protocol (X2AP) facilitates communication in 4G between eNodeBs, enabling inter-cell coordination, handovers, and mobility management. X2AP messages can update eNodeB configurations, report radio link failures, and acknowledge handover requests.

UTRAN Iub Interface NBAP Signaling (NBAP) is used in 3G to communicate between the NodeB (base station) and the Radio Network Controller (RNC) over the Iub interface. NBAP messages perform functions such as managing radio link removal, reconfiguration, power control, and various radio link measurements.

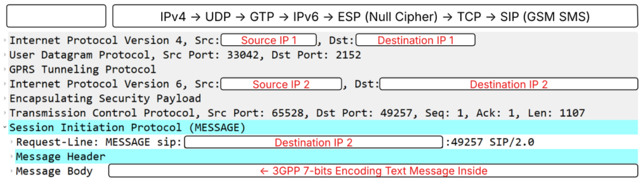
6.1.1 T-Mobile Cellular Backhaul. We observed satellite traffic corresponding to T-Mobile cellular backhaul that included plaintext user SMS and voice call contents, user Internet traffic, and cellular network signaling protocols. After we disclosed the vulnerability, T-Mobile quickly enabled encryption.

We identified three satellite beams carrying unencrypted T-Mobile traffic, with footprints spanning part of North America. All three transponders use a protocol stack consisting of DVB-S2 at the physical layer, followed by a proprietary encapsulation layer, an IP layer, and a GTP tunnel layer.

The observed network links, unencrypted S1-U/C links, transmitted customer data, including Internet traffic and IMS services such as voice and SMS. The unencrypted traffic included unprotected IMS signaling as well as metadata and content of real user SMS messages, call metadata, browsing history, and unencrypted RTP voice streams. The IMS traffic did have an IPsec layer, but it used a *null cipher* for encryption. Figure 8 illustrates an anonymized packet we extracted whose SIP MESSAGE method contains a 3GPP 7-bit encoded GSM SMS payload in plaintext, which is impossible to decode without correct parsing on all upper layer protocols.

From a 9-hour recording, we observed 2,711 users' phone numbers from metadata associated with voice calls and messages. We identified the traffic as belonging to T-Mobile based on the MCC & MNC code of users.

The S1-C control traffic we observed over the SCTP (port 36422) was the S1AP protocol.



```
IPv4 → UDP → GTP → IPv6 → ESP (Null Cipher) → TCP → SIP (GSM SMS)
› Internet Protocol Version 4, Src: Source IP 1 , Dst: Destination IP 1
› User Datagram Protocol, Src Port: 33042, Dst Port: 2152
› GPRS Tunneling Protocol
› Internet Protocol Version 6, Src: Source IP 2 , Dst: Destination IP 2
› Encapsulating Security Payload
› Transmission Control Protocol, Src Port: 65528, Dst Port: 49257, Seq: 1, Ack: 1, Len: 1107
˅ Session Initiation Protocol (MESSAGE)
  › Request-Line: MESSAGE sip: Destination IP 2 :49257 SIP/2.0
  › Message Header
  › Message Body   ← 3GPP 7-bits Encoding Text Message Inside
```

Figure 8: Anonymized SMS message protocol stack recovered from unencrypted T-Mobile satellite backhaul.

6.1.2 AT&T Mexico Cellular Backhaul. We observed unencrypted cellular backhaul traffic that included protocol metadata and cellular network signaling protocols, and raw user Internet traffic. For this transponder, we did not observe any unencrypted call or SMS contents, nor any IPsec tunnels that might contain this data.

Our analysis identified one satellite beam carrying unencrypted AT&T Mexico traffic, which could be received in much of North America. The transponder uses DVB-S2X, with byte-reversed encapsulation of IP packets. The traffic was IP packets with no encryption and no IPsec tunnel.

The unencrypted S1-U/C link exposes control and user plane traffic, revealing core network signaling, authentication data, VoLTE/VoIP signaling, and Internet traffic.

The S1-C signaling traffic we observed included UTRAN Iub Interface NBAP signaling, S1AP, and X2AP. This unencrypted SCTP traffic exposes sensitive information, including secret security keys (KeNB), network identity details (e.g., IMSI, PLMN, eNodeB CellID) session identifiers, (e.g., MME-UE-S1AP-ID, eNB-UE-S1AP-ID) and UE security capabilities, posing significant security risks.

The end-user Internet data we observed was standard S1-U IP-based communication, including HTTP, SSL, TCP, ESP, UDP, QUIC, and TLS packets. Many outer IP source addresses correspond to AT&T domains. Additionally, within these GTP tunnels, we identified inner source IP addresses from major Internet companies, including Facebook, Google, etc.

In a 30-minute recording, we observed 710 users' phone numbers and related control and Internet traffic. We identified the tower as belonging to AT&T based on its MCC & MNC codes.

6.2 VoIP Over Satellite

6.2.1 TelMex VoIP on Satellite Backhaul. We observed unencrypted satellite backhaul traffic that included the plaintext contents of user voice calls, and protocol metadata and cellular signaling protocols.

Our analysis identified three satellite beams carrying unencrypted TelMex VoIP traffic, whose satellite footprints cover part of North America. These transponders follow two distinct protocol stacks: one uses DVB-S2 with a proprietary protocol encapsulating IP, while the other two use a standard DVB-S2 with GSE (with 6/2 fragmentation) protocol to encapsulate IP. There was no IPsec tunnel present; data was sent in plain IP packets.

Our analysis confirmed the presence of unencrypted SIP signaling and RTP voice data. SIP signaling data was transmitted in clear text, revealing metadata that included caller/callee identities (phone numbers), SIP server and domain, call session metadata (Call-ID, CSeq numbers), SDP details (e.g., RTP ports, codec ITU-T G.729), and access network identifiers (e.g., IEEE-802.11 WiFi location data).

The captured data includes source IPs linked to TelMex's network, unencrypted RTP sessions corresponding to different calls, and compressed voice packets (ITU-T G.729 codec) We observed 142 voice conversationsfrom a 2-minute, 167 MB, capture.

6.2.2 KPU Telecommunications. KPU Telecommunications is an Alaskan telecom. We observed unencrypted satellite backhaul traffic carrying plaintext SIP signaling for some endpoints in their VoIP system. This traffic was being carried over a satellite link used by a customer of KPU. This satellite link was a secondary link enabled only while the main link was down. The link operated over a DVB-S2 physical layer, with an unidentified intermediate encapsulation preceding the IP layer. KPU traced the issue to a VPN that unexpectedly terminated at the satellite modem.

6.2.3 WiBo Satellite Links. We identified unencrypted satellite traffic associated with WiBo, the registered brand of Starsatel S.A. de C.V., a Mexican telecommunications provider offering broadband satellite and IP telephony services to remote and rural areas. The observed links used a standard DVB-S2 physical layer with GSE encapsulation (6/2 fragmentation) to carry IP traffic.

Our analysis revealed clear-text SIP signaling and RTP voice traffic within WiBo's network, exposing sensitive user metadata and audio content. SIP packets (e.g., INVITE, ACK, BYE) were transmitted over without encryption, and contained SDP fields revealing caller/callee phone numbers, IP addresses, negotiated codecs, and RTP port assignments. RTP streams carrying G.729 and G.711 audio were also unencrypted, enabling full reconstruction of call audio. While many calls used private-address source and destination IPs indicating internal VoIP traffic, some were routed to public IPs registered to WiBo.

We also observed large volumes of plaintext DNS traffic. DNS responses were sent to IPs owned by WiBo, likely acting as client resolvers. The data also contained upstream queries to public resolvers, such as Google Public DNS (8.8.8.8). DNS queries suggested typical activity from mobile devices, resolving domains like Apple iCloud, Android OS services, TikTok, and Samsung's app store.

6.3 Government and Military

We observed unencrypted satellite traffic belonging to government and military for multiple countries.

6.3.1 US Military. We observed both unencrypted (DNS, ICMP, SIP, SNMP) and encrypted (IPSec and TLSv1.2) traffic from sea vessels owned by the US military. One transponder encapsulates IP packets with GSE with the 6/2 fragmentation quirk, and the other has an unknown layer before the IP header. We were able to identify names of the vessels from addresses in the plaintext SIP packets. By investigating the names, we determined they were all formerly privately-owned ships that were now owned by the US.

6.3.2 Mexico Government and Military. We observed unencrypted satellite traffic from multiple organizations within the Mexican government, including military, law enforcement, and government agencies. These unencrypted links appear to be used to connect remote command centers, surveillance outposts, and mobile units via commercial satellite backhaul.

The two transponders use unencrypted DVB-S2 at the physical layer. One followed this with GSE encapsulation with the 6/2 fragmentation implementation quirk, followed by unencrypted IP. For the other transponder, it has byte-reversed word encoding of an unknown header followed by an IP header (Section 5).

The traffic that we observed includes DNS requests for hostnames and TLS certificates with Distinguished Name (DN) attributes for a variety of internal and external government hostnames and infrastructure. Among the traffic payloads, we observed large amounts of unencrypted HTTP traffic containing JSON and HTML formatted web application responses from internal systems used for infrastructure, logistics, and administrative management, including:

- References to military terminals, regions, and zones.
- Law enforcement asset inventory, personnel records, and traffic monitoring.
- Incident reporting, case tracking, and evidence documentation by field personnel and administrative staff, including narcotics activity and public gatherings.
- Military asset tracking records for aircraft, sea vessels, armored vehicles, and LIDAR and RADAR. This data included locations, deployments, mission roles, and maintenance logs.
- Real-time military object telemetry with precise geolocation, identifiers, and live telemetry.

6.4 Corporations

6.4.1 Walmart-Mexico Internal Network Traffic. Walmart is a multinational retail company, with thousands of stores in dozens of countries. An internal inventory management tool allows stock tracking, price updates, and store operation management via real-time data synchronization across store locations, distribution centers, and Walmart's central database.

We identified three satellite beams carrying unencrypted Walmart-Mexico internal system traffic that could be received across North America. Among the three transponders, two use DVB-S2 followed by GSE encapsulation with 6/2 fragmentation carrying IP traffic, while the third uses DVB-S2 followed by byte-reversed encapsulation with an unknown header before an IP header.

Notable internal network traffic includes:

- Logins via unencrypted telnet to their inventory management system, including plaintext credentials, terminal UI text, and invoice summaries.
- Inventory records transferred and updated via unencrypted FTP, including UPC and SKU numbers, retail/wholesale/cost numbers, store layouts, and sales data.
- Unencrypted internal corporate emails.
- Unencrypted NetBIOS internal computing equipment usage notices, indicating employee activity monitoring.

6.4.2 Grupo Santander Mexico. Grupo Santander is a major multinational financial institution. We identified unencrypted traffic from internal Grupo Santander Mexico networks being transmitted on a satellite transponder using DVB-S2 followed by GSE encapsulation with 6/2 fragmentation carrying IP traffic. The traffic suggests that Grupo Santander Mexico relies on satellite to support connectivity for remote branches, ATMs, and internal infrastructure.

The resolved packets' IPs fall within TelMex-assigned blocks. This suggests that traffic is routed over commercial ISP networks before reaching satellite uplinks. We observed traffic including

- TLS and IPsec traffic with certificates with DNs, internal CRLs, and OCSP URLs for internal network PKI.
- DNS responses for internal ATM-related domain names.
- Unencrypted LDAP traffic including a segregated AD domain hierarchy dedicated to ATM infrastructure.

6.4.3 Banjército and Banorte. Banjército is a bank affiliated with the Mexican military, and Banorte is a large commercial bank. We identified extensive unencrypted satellite traffic linked to the internal infrastructure of both banks being transmitted on a satellite transponder using DVB-S2 followed by GSE encapsulation with 6/2 fragmentation carrying IP traffic. The destination IP ranges were inside of the internal/private address range used by Banjército.

Plaintext traffic included DNS responses for domains tied to financial operations, ATMs and POS terminals, and CLDAP and LDAP authentication.

6.5 In-Flight WiFi

One of the common consumer applications of GEO satellite is providing airline passengers and crew with internet access (in-flight WiFi). Unencrypted in-flight WiFi traffic was previously observed in satellite transmissions by Baselt et al. [28], but our IP header parsing gives us deeper insights into the magnitude of this problem.

Among our captures, 40 included the string "inflight" in unencrypted data. 35 of these used DVB-S2 as the first layer, and 2 of these 35 included MPEG-TS as the second layer. 30 of the 35 include IP, 2 involve GSE, and 2 use the byte order reversal discussed in Section 5.1. We were able to associate DNS lookups and hostnames with two in-flight WiFi and entertainment providers, Gogo Inc and Panasonic. Unencrypted metadata from these captures included references to flights on at least ten different airlines.

The data included hostnames for the domains used by captive portals that users are redirected to, hostnames and accesses for flight information used by pilots, as well as DNS lookups, HTTPS and QUIC traffic, and IPsec and Wireguard traffic to and from domains that would be expected from normal end-user consumers. According to our conversations with Panasonic, they told us they rely on the ubiquity of TLS to secure end user applications.

For one in-flight transponder, roughly 40% of the IP packets included RTP traffic containing MPEG audio/video streams. Although the video streams were scrambled, the audio was not, and the audio appeared to belong to news programs, sports games, and more. We suspect that these packets are used to transmit live television to in-flight entertainment seatback systems. Roughly 60% of the IP packets include a proprietary protocol. We reverse engineered this protocol, and it appears to be a system which compresses TCP and UDP traffic to reduce bandwidth. This is consistent with a system which encapsulates passenger traffic, and understanding the proprietary protocol allows us to observe passenger traffic.

We also observe a handful of IP packets which include a partial PEM-encoded RSA private key. Although this exact same partial private key was previously documented by Pavur [28], our system gives us deeper insight into its provenance. Pavur observed the partial key by extracting strings from the raw satellite capture, and he concluded that the remaining bytes were lost due to connection quality issues, positing that a more reliable setup could recover the remaining bytes. With our improved parsing, we conclude that this is not the case. We reliably recover IP packets from this transponder with negligible error rate, indicating that signal quality is not an issue. The partial private key appears in UDP packets transmitted by a specific host IP on a specific port; these UDP packets also contain partial SQL statements and log strings.

Instead of attributing the partial key exposure to poor signal quality, we believe it is more likely that a specific device on the private network used for in-flight Wi-Fi has a bug that leads to the device leaking internal memory to the network. Because of our improved parsing, we are confident that improving the collection will not reveal more bytes of the private key. More advanced cryptanalytic techniques are required to recover the private key. We developed these advanced cryptanalytic methods and successfully recover the full private keys used by this device. We detail our methods in Appendix B.

6.6 Utilities and Infrastructure

6.6.1 Comisión Federal de Electricidad . The Comisión Federal de Electricidad (CFE) is a large electric utility in Mexico with 90,000 workers and 46 million customers. We observed one transponder carrying unencrypted CFE internal communications, whose footprint spans large portions of North America. The transponder uses DVB-S2 followed by GSE encapsulation with 6/2 fragmentation carrying IP traffic. Unencrypted internal traffic included:

- DNS responses for internal domains to and from private IPs.
- Structured JSON responses for customer service and maintenance work orders with locations, urgency levels, and customer names, addresses, account numbers, and tariff types.
- Labeled identifiers for power grid provisioning to military zones and government buildings.
- Internal reporting and configuration web forms listing administration and substation infrastructure.
- Internal maintenance systems for infrastructure failures and asset status, such as mechanical failures and safety hazards.

6.6.2 Other Industrial Applications. Pending ongoing disclosure, a future version of this document will contain further details on other unencrypted infrastructure and industrial data we observed, including utilities, maritime vessels, and offshore oil and gas platforms.

7 Discussion and Conclusions

Updating Threat Models. There is a clear mismatch between how satellite customers expect data to be secured and how it is secured in practice; the severity of the vulnerabilities we discovered has certainly revised our own threat models for communications. Cell phone traffic is carefully encrypted at the radio layer between phone and tower to protect it against local eavesdroppers; it is shocking to discover that these private conversations were then broadcast to large portions of the continent, and that these security issues were not limited to isolated mistakes. Similarly, there has been a concerted effort over the past decade or two to encrypt web traffic because of widespread concern about government eavesdropping through tapping fiber-optic cables or placing equipment in Internet exchange points; it is also shocking to discover that this traffic may simply be broadcast to a continent-sized satellite footprint.

Impediments to Encryption. The unencrypted traffic we observed results from a failure to encrypt at multiple levels of the satellite network protocol stack. At the satellite link/transport layer, streams using MPEG encoding have the option to use MPEG scrambling; 2/3 of the TV transponders we observed enabled this but only 10% of the non-TV transponders did. Only 20% of satellite transponders using GSE had encryption enabled. At the network layer, organizations can use IPsec to protect their traffic, but only 6% of the hundreds of transponders from which we recovered IP packets consistently encapsulated further layers using IPsec.

And finally, at the application layer, services can encrypt using TLS; remarkably, nearly all the end-user consumer Internet browsing and app traffic we observed used TLS or QUIC, while we observed extensive unencrypted internal traffic for critical infrastructure being broadcast via satellite.

There are a number of factors contributing to this state of affairs.

Network Visibility. The increased deployment of TLS for web traffic is the outcome of a concerted effort by browser vendors and privacy organizations to make HTTPS the default; TLS is well-studied by academics and the open web is the subject of a constant stream of academic measurement papers stretching back decades.

In contrast, the academic literature on satellite communication security and the security practices of internal networks is relatively scarce. Our work gives us a rare window into the security practices of these internal networks, which appear to be largely open.

Organizations that *do* have visibility into these networks have been raising alarms for some time. A 2021 US Executive Order [27] and a follow-up memo [62] mandate moving US government infrastructure to a "zero-trust" architecture, which includes encrypting data on all internal networks. A 2022 NSA security advisory about GEO satellite links states: "Most of these links are unencrypted, relying on frequency separation or predictable frequency hopping rather than encryption to separate communications" [49].

Abstraction. Satellite network connectivity may be provided via several layers of service providers; the network providers we spoke to told us that typically customers choose whether to enable encryption, but there may be multiple layers of customer relationships. From our conversations with vendors, no auditing tools exist that allow vendors to audit the security of their own satellite backhaul. Our work has identified multiple unintentional misconfigurations among organizations who had intended to enable encryption.

Economic incentives. We can observe from the popularity of encrypted satellite TV feeds that network operators will, in fact, encrypt streams when there is a clear economic incentive to do so [11, 54]. However, it appears that the incentives are aligned in the opposite direction for network traffic: enabling link-layer encryption can require additional license fees to use the crypto subsystem for specific satellite terminals and hubs [21, 35].

Efficiency. Enabling encryption can impact efficiency by incurring additional bandwidth overhead costs and also by requiring increased power consumption for limited-resource off-grid terminals relying on solar power. Panasonic told us that enabling encryption could incur a 20–30% capacity loss. In addition, when using IPsec, ESP and IP headers can introduce 20–30 bytes of overhead, which is nontrivial for small-packet applications like VoIP and video calls.

Reliability. For some of the traffic we observed, such as VoIP for emergency services, the lack of encryption is an intentional choice to maximize service reliability. LEO satellite networks are becoming increasingly popular, but may not provide 100% reliability; in contrast, users who are optimizing for maximal reliability may choose unencrypted GEO connections.

Usability. Usability has been a well-known impediment to cryptographic deployments for decades. The systems we observed where encryption had been inadvertently disabled by misconfigurations or software updates had "failed open" in that the network continued to function correctly with no apparent indication to the operators that no encryption was present. Modern public-key infrastructure still requires significant overhead to deploy and maintain up-to-date certificates on terminals and ground stations [33, 43]. Network providers also described to us how enabling encryption made it impossible to troubleshoot network issues.

Export controls. The documentation and marketing materials for the satellite terminals that we examined included artifacts of US export controls on cryptography, such as listing optional 56-bit key strengths [65]. Export controls have historically resulted in protocols and products being developed with encryption as a separate add-on so that vendors could comply with these regulations.

Future Work. We hope that our study inspires future work providing further visibility into satellite communciations as a critical component of our network infrastructure. Clear future directions include scanning other frequency bands such as Ka and C, developing methodologies for different orbits, and refining our parsing stack to account for further proprietary protocols. Further future work could also include reverse-engineering satellite receivers to understand protocol aspects that are not evident in our passive black-box setting.

Acknowledgments

We are grateful to Weyl Lu and Wenhao Chen for extensive discussions and time spent on the roof under the sun to help align the satellite dish; our facilities manager, Matt Scott, for assisting with the setup of the satellite dish receiver on the rooftop; kc Claffy, Javier Jamison, and Vijay Sarvepalli for assistance with discosure; Sajjad Matin and Dennis Klein for extensive conversations and legal help; to all the vendors who responded to us and remediated the issues and particularly to our contacts at T-Mobile, Panasonic, KPU Telecom, X2nSat, and Viasat for the insights they shared about their industry.

This research was supported by NSF grants no. CNS-1943240, CNS-2239163, and MPS-2232481 and a gift from Qualcomm.

References

[1] [n. d.]. Dvbv5-Zap. https://linuxtv.org/wiki/index.php/Dvbv5-zap.

[2] [n. d.]. TSDuck. https://tsduck.io/.

[3] [n. d.]. Tshark(1). https://www.wireshark.org/docs/man-pages/tshark.html.

[4] 2025. Ssloxford/Gsextract. SSL Oxford. https://github.com/ssloxford/gsextract.

[5] André Adelsbach and Ulrich Greveler. 2005. Satellite Communication Without Privacy – Attacker's Paradise. In *Sicherheit*. 257–268.

[6] Amazon.com. [n. d.]. FTA Universal Ku Band LNB, Single, 0.1dB Satellite Dish LNBF, Linear, Polarized (0.1 dB, Single) : Electronics. https://www.amazon.com/Universal-Single-Satellite-Linear-Polarized/dp/B01N1W636Q?th=1.

[7] Angel Electronics. 2024. STAB HH90 Satellite Dish Motor. https://angelelectronics.ca/products/stab-hh90-satellite-dish-motor.

[8] ATSC. [n. d.]. ATSC. https://www.atsc.org/documents/.

[9] Robin Bisping, Johannes Willbold, Martin Strohmeier, and Vincent Lenders. 2024. Wireless Signal Injection Attacks on VSAT Satellite Modems. *USENIX Security Symposium*.

[10] CCSDS. 2008. *Encryption Algorithm Trade Survey*. Technical Report. CCSDS.

[11] Neil Chenoweth. 2013. *Murdoch's Pirates: Before the Phone Hacking, There Was Rupert's Pay-TV Skullduggery*.

[12] cjcr. [n. d.]. EBSpro – The DXer Dream Come True! http://ebspro.cjcr-soft.com/.

[13] deeptho. 2022. tbs6904SE. https://www.satellitescommunity.de/forum/index.php?thread/2654-tbs6904se/.

[14] DVB. 1997. Digital Video Broadcasting (DVB); Framing Structure, Channel Coding and Modulation for 11/12 GHz Satellite Services. https://www.etsi.org/deliver/etsi_en/300400_300499/300421/01.01.02_60/en_300421v010102p.pdf.

[15] DVB. 2007. Digital Video Broadcasting (DVB); Specification for the Use of Video and Audio Coding in Broadcasting Applications Based on the MPEG-2 Transport Stream.

[16] DVB. 2009. Digital Video Broadcasting (DVB); Interaction Channel for Satellite Distribution Systems. https://www.etsi.org/deliver/etsi_en/301700_301799/301790/01.05.01_60/en_301790v010501p.pdf.

[17] DVB. 2013. DVB-CSA for DVB-IPTV (Common Scrambling Algorithm). https://www.etsi.org/deliver/etsi_ts/103100_103199/103127/01.01.01_60/ts_103127v010101p.pdf.

[18] DVB. 2014. Digital Video Broadcasting (DVB); Generic Stream Encapsulation (GSE); Part 1: Protocol. https://www.etsi.org/deliver/etsi_TS/102600_102699/10260601/01.02.01_60/ts_10260601v010201p.pdf.

[19] DVB. 2014. Digital Video Broadcasting (DVB); Second Generation Framing Structure, Channel Coding and Modulation Systems for Broadcasting, Interactive Services, News Gathering and Other Broadband Satellite Applications; Part 1: DVB-S2. https://www.etsi.org/deliver/etsi_en/302300_302399/30230701/01.04.01_60/en_30230701v010401p.pdf.

[20] DVB. 2014. DVB-CSA3 (Third Generation Common Scrambling Algorithm). https://www.etsi.org/deliver/etsi_ts/100200_100299/100289/01.02.01_60/ts_100289v010201p.pdf.

[21] DVB. 2020. Second Generation DVB Interactive Satellite System (DVB-RCS2); Part 1. https://www.etsi.org/deliver/etsi_en/301700_301799/301790/01.05.01_60/en_301790v010501p.pdf.

[22] DVB. 2024. Digital Video Broadcasting (DVB); Second Generation Framing Structure, Channel Coding and Modulation Systems for Broadcasting, Interactive Services, News Gathering and Other Broadband Satellite Applications; Part 2: DVB-S2 Extensions (DVB-S2X). https://www.etsi.org/deliver/etsi_en/302300_302399/30230702/01.04.01_60/en_30230702v010401p.pdf.

[23] DVB. 2024. Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB Systems. https://dvb.org/wp-content/uploads/2024/09/A038r17_Specification-for-Service-Information-SI-in-DVB-Systems_Draft_EN_300-468-v1-19-1_September-2024.pdf.

[24] DVB. 2025. Digital Video Broadcasting (DVB); DVB Specification for Data Broadcasting. https://www.etsi.org/deliver/etsi_en/301100_301199/301192/01.08.00_20/en_301192v010800a.pdf.

[25] Erik Tews, Julian Wälde, and Michael Weiner. 2011. Breaking DVB-CSA. 45–61. https://doi.org/10.1007/978-3-642-34159-5_4.

[26] Eutelsat. 1998. DiSEqC Bus Functional Specification Part 2: Version 1.2. https://www.eutelsat.com/files/contributed/satellites/pdf/Diseqc/associated%20docs/bus_spec_part2_v1_2.pdf.

[27] Executive Office of the President. 2021. Improving the Nation's Cybersecurity. https://www.federalregister.gov/documents/2021/05/17/2021-10460/improving-the-nations-cybersecurity.

[28] Georg Baselt, Martin Strohmeier, James Pavur, Vincent Lenders, and Ivan Martinovic. 2022. Security and Privacy Issues of Satellite Communication in the Aviation Domain. In *2022 14th International Conference on Cyber Conflict: Keep Moving! (CyCon)*, Vol. 700. 285–307.

[29] Glyn Moody. 2016. New Snowden Leaks Reveal "Collect It All" Surveillance Was Born in the UK. https://arstechnica.com/tech-policy/2016/09/snowden-leaks-collect-all-signals-surveillance-born-in-uk/.

[30] Gorry Fairhurst. [n. d.]. Unidirectional Lightweight Encapsulation (ULE) Next-Header Registry. https://www.iana.org/assignments/ule-next-headers/ule-next-headers.xhtml.

[31] Nadia Heninger and Hovav Shacham. 2009. Reconstructing RSA Private Keys from Random Key Bits. In *Advances in Cryptology - CRYPTO 2009*, Shai Halevi (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 1–17.

[32] Hughes. [n. d.]. Satellite Connectivity and Managed Networks. https://www.hughes.com/.

[33] Hughes. 2008. HX-System Overview. https://usermanual.wiki/Hughes/HxSystemoverview.867933836/view.

[34] HUGHES. 2015. Antenna Pointing Guide. https://txdish.com/wp-content/uploads/2017/09/Antenna-pointing-guide.pdf.

[35] iDirect. [n. d.]. TRANSMISSION SECURITY. https://www.idirect.net/wp-content/uploads/2020/03/WhitePaper-GovDef-TRANSEC.pdf.

[36] INTELSAT. [n. d.]. Galaxy 37/Horizons-4 at 127°W Fact Sheet. https://www.intelsat.com/wp-content/uploads/2023/08/G-37-H4-Factsheet.pdf.

[37] INTELSAT. 2020. Adjacent Satellite Interference in Mobile/VSAT Environments. https://www.intelsat.com/wp-content/uploads/2020/04/intelsatgeneral-satellite-interference-vsat-whitepaper.pdf.

[38] ISO/IEC. 2000. Information Technology — Generic Coding of Moving Pictures and Associated Audio Information: Systems. https://www.iso.org/standard/31537.html.

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

[40] James Pavur, Daniel Moser, Vincent Lenders, and Ivan Martinovic. 2019. Secrets in the Sky: On Privacy and Infrastructure Security in DVB-S Satellite Broadband. In *WISEC: Security and Privacy in Wireless and Mobile Networks*. 277–284.

[41] Johannes Willbold, Moritz Schloegel, Robin Bisping, Martin Strohmeier, Thorsten Holz, and Vincent Lenders. 2024. VSAsTer: Uncovering Inherent Security Issues in Current VSAT System Practices. In *Proceedings of the 17th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. Seoul Republic of Korea.

[42] Kai Wirt. 2004. Fault Attack on the DVB Common Scrambling Algorithm.

[43] Frank Kelly and David Kloper. 2008. System and Method for Scaling a Two-Way Satellite System. https://patentimages.storage.googleapis.com/e5/e6/d3/49dcd30eb81211/US7463582B2.pdf.

[44] Louis F. DeKoven, Audrey Randall, Ariana Mirian, Gautam Akiwate, Ansel Blume, Lawrence K. Saul, Aaron Schulman, Geoffrey M. Voelker, and Stefan Savage. 2019. Measuring Security Practices and How They Impact Security. In *Proceedings of the Internet Measurement Conference*. Amsterdam Netherlands, 36–49.

[45] LyngSat. [n. d.]. LyngSat. https://www.lyngsat.com/.

[46] Alexander May. 2010. Using LLL-Reduction for Solving RSA and Factorization Problems. Springer, 315–348. doi:10.1007/978-3-642-02295-1

[47] Minghao Lin, Minghao Cheng, Dongsheng Luo, and Yueqi Chen. 2023. CLExtract: Recovering Highly Corrupted DVB/GSE Satellite Stream with Contrastive Learning. http://arxiv.org/abs/2310.08210. arXiv:2310.08210 [eess]

[48] Multicom. 2023. 1m DTH Ku Band Satellite Dish. https://www.multicominc.com/product/multicom-mul-1m-ku-1m-dth-ku-band-satellite-dish/.

[49] NSA. 2022. NSA Issues Recommendations to Protect VSAT Communications. https://www.nsa.gov/Press-Room/News-Highlights/Article/Article/2910409/nsa-issues-recommendations-to-protect-vsat-communications/.

[50] OneSat. [n. d.]. Flat Panel Electronically Steerable Ku Antenna. http://www.inverto.tv/en/over_satellite/520/flat-panel-electronically-steerable-ku-antenna.

[51] Doreet Oren. 2021. Cellular Backhauling over Satellite - Not for you? Think again. https://www.gilat.com/wp-content/uploads/2021/04/Gilat-article-Eastern-Digital-Media-2021-03-Cellular-Backhauling-over-Satellite.pdf

[52] Palo Alto Networks Inc. [n. d.]. Mobile Network Infrastructure Getting Started. https://docs.paloaltonetworks.com/content/dam/techdocs/en_US/pdf/service-providers/10-1/mobile-network-infrastructure-getting-started/mobile-network-infrastructure-getting-started.pdf.

[53] Panasonic Avionics. [n. d.]. Panasonic Avionics. https://www.panasonic.aero.

[54] Timothy Pratt and Jeremy Allnutt. 2020. *Satellite Communications*.

[55] PROMAX. 2017. PLS - Physical Layer Scrambling. https://www.promaxelectronics.com/ing/news/524/pls-physical-layer-scrambling/.

[56] Ralf-Philipp Weinmann and Kai Wirt. 2005. Analysis of the DVB Common Scrambling Algorithm. (2005).

[57] SATBEAMS. [n. d.]. Satbeams - World Of Satellites at Your Fingertips. http://www.satbeams.com/.

[58] Satellite Industry Association. [n. d.]. Broadband Connectivity. https://sia.org/satellites-services/broadband-connectivity/.

[59] Satellite Industry Association. 2024. COMMERCIAL SATELLITE INDUSTRY CONTINUES HISTORIC GROWTH WHILE DOMINATING GLOBAL SPACE BUSINESS – SIA RELEASES 27TH ANNUAL STATE OF THE SATELLITE INDUSTRY REPORT. https://sia.org/commercial-satellite-industry-continues-historic-growth-dominating-global-space-business-27th-annual-state-of-the-satellite-industry-report/.

[60] SatNow. [n. d.]. GEO Satellite Terminals. https://www.satnow.com/search/satellite-terminals/filters?page=1&country=global&sorbit=;GEO;.

[61] Matthew Scholl and Theresa Suloway. 2023. *Introduction to Cybersecurity for Commercial Satellite Operations*. Technical Report. NIST. https://csrc.nist.gov/pubs/ir/8270/final.

[62] Shalanda D Young. 2022. Moving the U.S. Government Toward Zero Trust Cybersecurity Principles.

[63] Shoghi. [n. d.]. Satellite Monitoring System and Satellite Interception. https://www.shoghicom.com/products/intelligence-surveillance-reconnaissance/satellite-monitoring.

[64] Joshua Smailes, Sebastian Kohler, Simon Birnbach, Martin Strohmeier, and Ivan Martinovic. 2023. Watch This Space: Securing Satellite Communication through Resilient Transmitter Fingerprinting. *ACM Conference on Computer and Communications Security (CCS)*.

[65] ST Engineering. 2019. Newtec Dialog.

[66] Stab Italia S.r.l. 2005. *Installationsanleitung Stab HH DiSEqC/USALS Motor: Anleitung, Datenblatt, Aufbau, Manuell (EN-DE-IT-FR-ES)*. Stab Rotor Sat Division. Installation manual for the Stab HH100/HH120 rotor motor.

[67] TBS. [n. d.]. TBS5927 Professional DVB-S2 TV Tuner USB. http://www.tbsiptv.com/download/tbs5927/tbs5927_professtional_dvb-S2_TV_Tuner_USB_data_sheet.pdf.

[68] Union of Concerned Scientists. 2023. UCS Satellite Database. https://www.ucsusa.org/resources/satellite-database.

[69] Inc. Viasat. 2022. KA-SAT Network Cyber Attack Overview. https://news.viasat.com/blog/corporate/ka-sat-network-cyber-attack-overview.

[70] Wei Li and Dawu Gu. 2007. Security Analysis of DVB Common Scrambling Algorithm. In *The First International Symposium on Data, Privacy, and E-Commerce (ISDPE 2007)*. 271–273.

[71] Wenyi Morty Zhang, Annie Dai, Keegan Ryan, Dave Levin, Nadia Heninger, and Aaron Schulman. [n. d.]. Satellite Communication Research Project. https://satcom.sysnet.ucsd.edu.

Table 4: Ground-truth anchor transponder service identifiers and their associated satellite parameters.

Satellite Name	Long.	Freq. (MHz)	Pol.	Identifier(s)
Telstar 14R/19 Vantage	63.0W	12184.197	H	NI-VRN2-063W
Telstar 14R/19 Vantage	63.0W	12184.197	H	NI-VRN2-063WJ
Eutelsat 65 West A & Star One C1	65.0W	10718.770	H	Eutelsat Advance E65WA AMERICAS
Eutelsat 65 West A & Star One C1	65.0W	10801.762	H	Eutelsat Advance E65WA AMERICAS
Eutelsat 65 West A & Star One C1	65.0W	10926.253	H	Eutelsat Advance E65WA AMERICAS 8
SES 10	67.0W	10969.750	V	US-SMT-1
SES 10	67.0W	11489.709	H	BEAM_SES10MC_W01_0469/0469J
Star One D1	84.0W	12009.749	V	NI-VRN1-081W
DirecTV 11/14 & Galaxy 16 & Spaceway 2	99.2W	11840.405	H	ACPD.IM.MTN.FOX.1
DirecTV 11/14 & Galaxy 16 & Spaceway 2	99.2W	11899.726	H	BEAM_G16_W01_0067
SES 1	101.0W	11828.755	H	NI-VRN4-101W
Eutelsat 117 West A/B	116.9W	11980.256	V	ST4G-LCS-2
Eutelsat 117 West A/B	116.9W	12170.697	V	NI-BRW4-117W, NI-BRW4-117WJ
Galaxy 13/Horizons 1	127.0W	11899.711	V	G13-NAK-10K, G13-NAK-16K, G13-NAK-16KJ
Galaxy 13/Horizons 1	127.0W	12119.709	H	G13-NAK-21K, G13-NAK-21KJ
Galaxy 13/Horizons 1	127.0W	11959.721	H	YAC10-SN03
Ciel 2 & SES 15	129.0W	11571.704	V	BEAM_SES15WB_W01_0029
Ciel 2 & SES 15	129.0W	11317.197	V	BEAM_SES15-S27_S01_0030
Ciel 2 & SES 15	129.0W	11989.230	V	NI-BRW-129W, NI-BRW-129WJ
Ciel 2 & SES 15	129.0W	12068.225	V	NI-BRW-129W, NI-BRW-129WJ

A Supplementary Dish Alignment Strategies

A.1 Reference-Based Calibration Techniques

To overcome dish azimuth alignment limitations, we explored several experience-based methods using transponder signals as alignment anchors. By scanning for known satellites and analyzing signal strength, users can identify systematic pointing errors. For example, if the motor is configured to point at longitude X° but instead receives signal from a satellite located at Y°, this indicates a misalignment of $|X - Y|^\circ$. The user can then physically rotate the dish mount by this offset and re-scan iteratively until the alignment is corrected.

A.2 Signal-Based Alignment References

We identified several practical reference signals that offer reliable feedback during alignment:

- **Public/Unencrypted TV Services:** Many GEO satellites broadcast free-to-air (FTA) television channels that provide stable reference signals. Resources like `LyngSat` [45] list transponder frequencies, symbol rates, and polarizations widely used for alignment in the satellite TV community.
- **Transponder Identifiers:** During blind scans, we found that certain transponders broadcast satellite-specific identifiers via DVB-SI tables, specifically the Network Information Table (NIT) and Service Description Table (SDT) [23]. While the Satellite Delivery Descriptor is rarely populated in practice, these fields offer strong confirmation of satellite identity. We identified at least 12 transponders across 8 longitudes (from 65.0°W to 129.0°W; see Table 4) that can serve as ground-truth anchors—spanning both central and edge positions within the GEO arc.
- **Transponder Service List:** Over a year-long scan, we compiled a comprehensive list of active Ku-band transponders visible from our location. This dataset enables satellite identification by matching observed parameters to known metadata. We observed frequent churn, including deactivations and migrations, reinforcing the need for periodic updates and long-term monitoring.

A.2.1 Signal Strength and Bandwidth Distribution of Transponders.

We extracted two key physical-layer parameters of all the transponders: signal-to-noise ratio (SNR) and bandwidth. Spanning longitudes from 57.2°W to 177.2°W, our results provide the first system-wide snapshot of real-world Ku-band transponder characteristics. The observed heterogeneity across satellites highlights the diversity of physical-layer configurations.

Signal-to-noise ratio (SNR) quantifies the clarity and strength of received signals. Fig. 9 presents the SNR distribution for all detected transponders, grouped by satellite and ranked by median SNR. Most satellites exhibit tightly clustered SNR values, indicating standardized transponder configurations. In contrast, several satellites (e.g., 127.0°W, 91.0°W, and 114.9°W) display wide SNR variability (interquartile range > 6 dB; full range > 10 dB), likely reflecting heterogeneous hardware setups or beam-specific characteristics, underscoring the complexity of real-world satellite deployments. Notably, 98.6% of all transponders had SNRs exceeding 6 dB, demonstrating the robustness and sensitivity of our scanning system.

Bandwidth reflects the maximum data-carrying capacity of each transponder. A transponder's bandwidth determines how much spectrum is allocated for its signal transmission. Higher bandwidth allows for more symbols to be transmitted per unit time, which directly increases raw throughput. Additionally, modern modulation

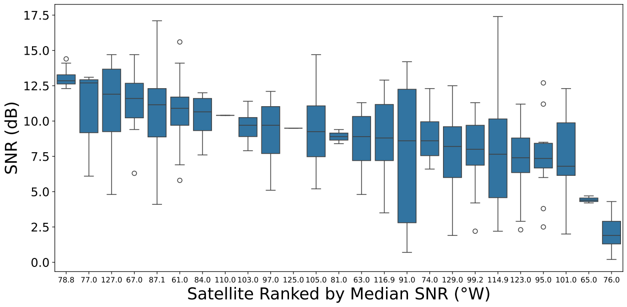


Figure 9: Distribution of Transponder SNR.

and coding schemes (e.g., 8 Phase Shift Keying (8PSK) or 16 Phase Shift Keying (16APSK)) enable higher spectral efficiency—measured in bits per second per Hz—making wideband transponders essential for delivering high-capacity services such as trunked backhaul, IP-based content delivery, and satellite internet access.

Figure 10 shows the distribution of detected transponder bandwidths. While sub-5 MHz channels are the most common (131 transponders, 34.8%), transponders span a wide range of bandwidth classes. Notably, 73 transponders (19.4%) have bandwidths ≥ 30 MHz, sufficient to support data rates exceeding 100 Mbps under 8PSK modulation.

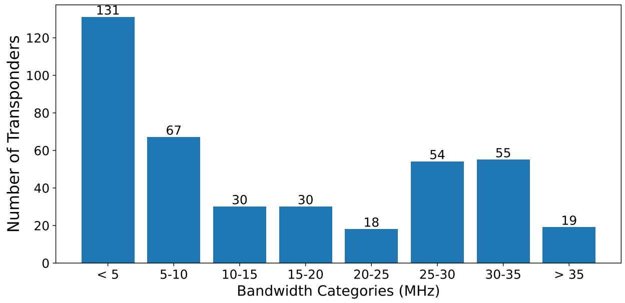


Figure 10: Distribution of Transponder Bandwidth.

B Key Recovery

As described in Section 6.5, in the course of our study we observed partial PEM-encoded RSA private keys. To fully understand the severity of this partial exposure, we wanted to determine whether or not enough bytes were leaked to lead to complete compromise of the private key. Although existing cryptanalytic approaches are insufficient to accomplish this, we develop new methods to recover private keys from this amount of private exposure.

PEM encoding is a commonly used format for representing cryptographic keys. For a RSA private key represented using ASN.1, integers in the private key are first encoded as big-endian byte arrays, the full DER-encoded byte array is Base64-encoded as printable text, and a “`BEGIN RSA PRIVATE KEY`” header and corresponding footer are added. Our observations included the suffixes of two PEM-encoded RSA private keys.

In particular, we observed the final 472 and 474 bytes of these encoded keys. Based on the standard order that private key values appear in, Base64-decoding and DER-decoding the partial PEM keys reveal the following values, corresponding to a 2048-bit RSA modulus n with prime factors p and q, public exponent e, and private exponent d:

- 512 least significant bits of $d_p = d \pmod{p-1}$
- all 1024 bits of $d_q = d \pmod{q-1}$
- all 1024 bits of $q_{\text{inv}} = q^{-1} \pmod{p}$.

Notably, this does not include the public modulus $n = pq$. Although the public exponent e is also missing, it is extremely common in practice that $e = 65537$, so we assume this is the correct value for the remainder of our recovery algorithm.

Heninger and Shacham [31] document how knowledge of (n, e, d_q) or (n, q_{inv}) reveal all private key values, but because the attacker in our example does not know n, those attacks do not apply. We are unaware of any other prior work that demonstrates key recovery from $(\text{LSB}(d_p), d_q, q_{\text{inv}})$, so we develop our own.

First, the modular equation $ed_q \equiv 1 \pmod{q-1}$ implies the existence of k_q such that $1 \le k_q \le e$ and

$$ed_q - 1 = k_q(q-1).$$

Because there are only 2^{16} possible values for k_q, we try each value until we find a candidate where $q = (ed_q - 1)/k_q + 1$ is both integer and prime. Typically this is sufficient to uniquely recover q.

Next, we attempt to recover p. We will ultimately use Coppersmith’s method [46] to find a small solution to a linear equation modulo a divisor of a known integer, but for this to work, we need to first find a bounded multiple of p. Note that since $qq_{\text{inv}} \equiv 1 \pmod{p}$, then $qq_{\text{inv}} - 1$ is a multiple of p, but this 2048-bit value is too large for Coppersmith’s method to efficiently succeed. We apply the elliptic curve factorization method to $qq_{\text{inv}} - 1$ to find and divide out small prime factors from this integer. The running time of this method depends on the size of the smallest factor, so with high probability we are left with a number of small prime factors and a large composite divisor m of $qq_{\text{inv}} - 1$. This divisor is a bounded multiple of p. In an example run, the multiple m of p was around 1900 bits, which is small enough for Coppersmith’s method to be efficient.

Finally, we use Coppersmith’s method to recover p. We write $d_p = d_{p,\text{msb}}2^t + d_{p,\text{lsb}}$ where $d_{p,\text{msb}}$ is unknown and $d_{p,\text{lsb}}$ is known. As before, there exists k_p such that $1 \le k_p \le e$ and $ed_p - 1 = k_p(p-1)$. We rewrite this as

$$e(2^t d_{p,\text{msb}} + d_{p,\text{lsb}}) - 1 + k_p \equiv 0 \pmod{p}.$$

We try all 2^{16} possible values for k_p, and each guess gives us a linear equation modulo a divisor of m with a single 512-bit unknown $d_{p,\text{msb}}$. We use Coppersmith’s method with a dimension-30 lattice to attempt to solve this equation; if the guess of k_p is correct, then Coppersmith’s method recovers d_p and p, completing recovery of the RSA private key.

This final step is the most expensive part of our attack since it requires 2^{16} invocations of Coppersmith’s method. However, thanks to our use of factorization to reduce the size of m, the required dimension of the lattice is quite reasonable, and the overall running time is not prohibitively expensive.

Table 5: Per-Longitude Transponder Change Summary

Longitude	Total # of Beams found	# of New Beams at Feb.2025	# of Disappeared Beams at Feb.2025
129.0°W	27	5	8
127.0°W	24	6	3
125.0°W	1	1	0
123.0°W	27	5	3
116.9°W	34	4	5
114.9°W	38	11	3
110.0°W	2	0	0
105.0°W	36	6	10
103.0°W	8	0	4
101.0°W	16	3	1
99.2°W	24	7	3
97.0°W	24	2	4
95.0°W	12	0	0
91.0°W	8	2	0
87.1°W	20	3	2
84.0°W	5	1	0
81.0°W	3	2	0
78.8°W	22	0	0
77.0°W	6	0	0
76.0°W	13	0	0
74.0°W	11	2	0
67.0°W	20	4	7
65.0°W	3	0	0
63.0°W	8	3	3
61.0°W	19	4	6
Total	**411**	**71**	**62**

This demonstrates that although only parts of the PEM-encoded private keys were exposed in the satellite traffic, enough bytes leaked to enable full recovery. We do not have enough information to know exactly what role these keys have in the in-flight infrastructure, but this cryptanalysis demonstrates that these private keys should be considered to be fully compromised.

C Ku-band GEO satellite stablity analysis

To provide a clear view of transponder activity dynamics, we include a summary table of changes (Table 5) of all detected transponders across two scanning periods: August 2024 and February 2025. In this version of the full document, we are omitting the full list of satellite names and transponder metadata out of an abundance of caution and to prevent easy targeting of sensitive systems while disclosure is ongoing.

The results highlight the dynamic nature of GEO Ku-band transponder usage across different longitudes. While some satellites maintain stable configurations with no observable changes (e.g., 95.0°W, 78.8°W, and 77.0°W), others exhibit significant churn. For example, satellites at 129.0°W and 105.0°W experienced notable turnover, with 8 and 10 transponders disappearing respectively, alongside several newly activated channels. This suggests that some operators dynamically lease transponder capacity for time-bounded services, periodically adjusting activation based on contractual agreements, shifting user demand, or regional service needs.

These findings underscore the heterogeneous and commercially-driven nature of satellite communications. Transponder availability is not solely a function of hardware capability, but also of operational policy, market fluctuation, and service allocation strategy. The coexistence of both stable and dynamic configurations highlights the value of continuous, longitudinal scanning to understand the evolving landscape of satellite transponder deployment.