IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 13 May 2026, accepted 8 June 2026. Date of publication 00 xxxx 0000, date of current version 00 xxxx 0000.

Digital Object Identifier 10.1109/ACCESS.2026.3705697

CubeSat Cybersecurity: An Overview

JOÃO PAOLO CAVALCANTE MARTINS OLIVEIRA[1], FRANCISCO JOSÉ TARGINO VIDAL[1], AND FÁTIMA MATTIELLO-FRANCISCO[2]

[1]School of Science and Technology, Federal University of Rio Grande do Norte, Natal 59078-900, Brazil
[2]National Institute for Space Research (INPE), São José dos Campos 12227-010, Brazil

Corresponding author: João Paolo Cavalcante Martins Oliveira (paolo.oliveira.948@ufrn.edu.br)

ABSTRACT CubeSats are no longer used only as educational platforms. They now support Earth observation, communications, scientific payloads, and missions with defense relevance. Their low cost and short development cycles expand access to space, but they also expose missions to security weaknesses that are often harder to control than in larger spacecraft programs. CubeSat projects rely more heavily on commercial off-the-shelf components, compact radios, mixed ground-software stacks, small operations teams, and strict size, weight, power, and cost limits. Drawing on recent papers, policy guidance, and space-security standards, this survey examines how those conditions shape cyber risk across onboard software, communication links, supply chains, ground systems, and operational procedures. It reviews defenses discussed in the literature, including secure-by-design practices, command authentication, encryption, segmentation, verification and validation, and mission-oriented test strategies. The paper argues that CubeSat security should not be treated as a reduced version of conventional spacecraft security. Instead, mission assurance depends on disciplined lifecycle engineering, protected command paths, and realistic testing practices that fit the technical and organizational constraints of small-spacecraft missions.

INDEX TERMS CubeSat, cybersecurity, mission assurance, satellite security, small spacecraft.

I. INTRODUCTION

CubeSats have become one of the main entry points to space for universities, startups, civil agencies, and defense organizations because their standardized form factor and lower launch cost reduce barriers to mission development [1], [2]. What began primarily as an educational platform now supports Earth observation, technology demonstration, communications, and operational constellations [1], [2]. The security consequence is straightforward: a compromised CubeSat mission may no longer be an isolated student failure, but a source of corrupted mission data, denied service, degraded communications, or loss of control over an orbital asset. Here, compromise follows the NIST definition of unauthorized disclosure, modification, substitution, or use of sensitive data, or unauthorized modification of a security-related system, device, or process to gain unauthorized access [3], [4], [5], [6], [7].

The associate editor coordinating the review of this manuscript and approving it for publication was Zijian Zhang.

Cybersecurity risks in space systems are not new, but CubeSats sharpen them in distinctive ways. Programs are often schedule-driven, budgets are tight, and the engineering culture may reward reaching orbit more than sustaining resilience after commissioning. Small teams also reuse public software stacks, commodity radios, and mixed ground infrastructure, creating security dependencies that extend well beyond the spacecraft bus [8], [9], [10]. This is precisely the environment in which secure-by-design principles are hardest to implement and most costly to defer [4], [11].

Recent work shows a field moving from warning to engineering. CubeSat-specific studies have produced attack trees, best-practice recommendations, and lightweight communication protections [9], [12], [13]. Broader satellite-communications surveys map attack surfaces across the space, link, and ground segments [5], [6], [7], [14]. Empirical and infrastructure-oriented papers have begun to examine satellite software artifacts, community sandboxes, and reproducible testbeds so that satellite-security research can become less anecdotal and more comparable across institutions [15], [16], [17]. The unresolved problem

is not awareness of vulnerability; it is converting awareness into repeatable engineering processes, standards adoption, and evidence that survives operational constraints.

Based on research from IEEE, AIAA, ACM, NASA, CISA, ESA, CCSDS, and related venues, this review takes a different approach: instead of cataloging attacks, it treats CubeSat cybersecurity as a question of how the field has been studied and understood, focusing on how the field has evolved, which methods dominate, where the evidence remains thin and which conclusions recur across independent studies. The central claim is that CubeSat cybersecurity now needs less generic advocacy and more disciplined treatment of mission boundaries, command authority, ground operations, and validation.

II. BACKGROUND

CubeSat cybersecurity sits at the intersection of spacecraft engineering, wireless communications, embedded systems, and operational security. The three-segment model used throughout satellite-security work–space, link, and ground or user segments–is a necessary baseline because it prevents the spacecraft from becoming the only object of analysis [4], [5], [6]. A mission can be compromised without a direct exploit of flight hardware: mission-planning workstations, cloud services, ground-station software, key-management systems, and the telecommand or telemetry radio path can all become decisive attack surfaces [6], [11].

For CubeSats, that architecture is compressed but not simplified. A typical mission uses a small onboard computer, a compact power system, one or more sensors or payloads, an attitude-control subsystem of varying sophistication, and a UHF, S-band, X-band, or similar communications chain. On the ground, the same mission may depend on university laboratories, outsourced mission operations, shared antenna networks, amateur-radio infrastructure, or cloud-hosted data pipelines [2], [8]. These choices increase flexibility, but they make the security boundary diffuse. The operational system may include laptops, scripts, APIs, networked test equipment, and third-party services that were never designed as parts of a high-assurance spacecraft control enclave [6], [11].

The literature repeatedly points to three structural drivers of risk: resource scarcity, organizational scarcity, and lifecycle compression. Resource scarcity follows from strict size, weight, and power constraints, which limit the margin available for cryptographic acceleration, redundant computing, tamper resistance, and fault-tolerant software [8], [9]. Organizational scarcity appears when teams lack dedicated security engineers, red teams, or 24/7 mission-security operations [9], [11]. Lifecycle compression appears when missions are built quickly from inherited components and accept design shortcuts that would be less tolerable in larger spacecraft programs [4], [8].

Guidance written for general space systems therefore needs reinterpretation rather than direct transplantation. Space Policy Directive-5 (SPD-5) states that cybersecurity principles for terrestrial systems also apply to space systems, while emphasizing space-specific needs such as positive control, protected command and telemetry paths, and cybersecurity integration before launch because most spacecraft cannot be physically accessed once deployed [4]. For CubeSats, this lock-in is severe. Once a 3U or 6U platform is on orbit, retrofitting trust anchors, redesigning boot chains, or replacing radios is usually impossible. Security design errors have a longer operational half-life than they would in many terrestrial embedded systems.

The selected literature reinforces this point from several angles. Falco et al. develop attack trees for CubeSat systems and show that small-spacecraft missions inherit a broad set of cyber-physical attack paths, not a single radio-layer problem [12]. Visner and Kordella argue for small-satellite cyber best practices adapted from other critical sectors [9]. Caudill's proposal for shared security infrastructure is more pointed: many small-spacecraft developers cannot realistically build bespoke protections under severe cost constraints [8]. More recent work on reproducible satellite-security testbeds exposes the same gap from an evaluation perspective: without standard experimental baselines, defenses are difficult to compare and their operational cost is difficult to measure [17], [18].

A. HOW THE LITERATURE HAS EVOLVED

Chronology matters because the unit of analysis has changed. Early CubeSat-specific work concentrated on lightweight communications protection and operational practicality. Challa et al. proposed CubeSec and GndSec in 2012 as a lightweight security approach for CubeSat communications, reflecting an early assumption that the primary challenge was securing a constrained command-and-telemetry path rather than the entire mission ecosystem [13]. In parallel, SATCOM resilience work such as Lehto et al. treated protection more broadly but remained oriented toward communications architecture and mission assurance rather than end-to-end cyber operations [19].

Around 2020–2021, best-practice papers and ASCEND contributions reframed CubeSat security as an engineering-governance problem involving resource limitations, supply-chain reliance, and the absence of mature shared infrastructure [8], [9], [20]. Attack-tree and sandbox work made that shift more concrete by introducing explicit attack decomposition and community test platforms [12], [16]. The field was no longer concerned only with security for radios; it was beginning to ask how missions fail.

From 2022 onward, the field became more recognizably research-driven. Systematic and historical syntheses argue that the discipline remains fragmented, with scattered terminology, thin empirical validation, and limited cross-pollination between spacecraft engineering and security research [21]. Willbold et al. provided rare experimental evidence in 2023 by showing that satellite software security lags behind state-of-the-art software security practice [15]. More recent review papers attempt wider synthesis across

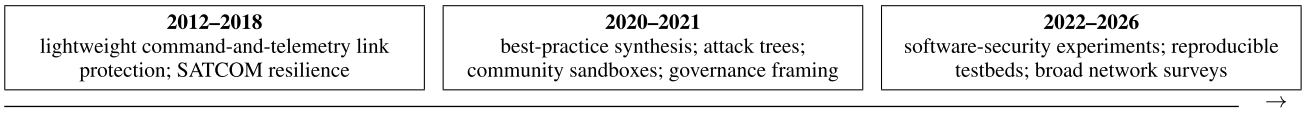

FIGURE 1. How the reviewed literature has shifted emphasis, from securing the radio link toward mission-level analysis and reproducible evaluation. The arrow indicates changing focus, not uniform progress.

satellite networks, IoT-driven satellite systems, and space information networks [5], [6], [7], [14], [22]. The literature is now large enough for comparison, but not mature enough to claim methodological convergence. Figure 1 sketches this shift in emphasis over time; it marks where attention moved, not a claim that each phase reached operational maturity.

B. WHAT THE LITERATURE STILL MISSES

Even with this expansion, the evidence remains skewed. Survey and policy papers outnumber flight-grounded evaluations; conceptual taxonomies outnumber comparative experiments; and many papers discuss satellites generically even when CubeSat constraints are materially different [7], [21]. Perception studies suggest that practice has evolved unevenly. Jones' survey of space professionals indicates that organizational understanding of satellite cybersecurity improved between 2012 and 2022, but not at a pace that implies consistent operational maturity across the sector [23]. The state of the art should therefore be described cautiously: broader and deeper than a decade ago, but still consolidating.

Several recent surveys cover overlapping ground, and it is worth being explicit about where this overview sits among them. Table 1 summarizes that comparison. Most existing reviews treat satellites broadly or center on the communications network; the contribution here is to read the same literature through CubeSat constraints, where resource, organizational, and lifecycle limits change which controls are realistic.

III. THREAT MODEL

An adequate threat model for CubeSats must cover more than unauthorized radio transmission. The protected assets include the ability to maintain positive control of the spacecraft, the integrity and confidentiality of telemetry and payload data, the availability of mission services, the correctness of orbital and timing information, and the trustworthiness of the ground systems that plan and issue commands [4], [6], [11]. Depending on the mission, sensitive assets may also include proprietary algorithms, mission timelines, remote-sensing products, or dual-use data of strategic value.

Adversaries span a wide range. At the low end are opportunistic actors exploiting exposed web services, default credentials, or misconfigured VPNs in mission operations. More capable actors may include motivated researchers, criminal groups targeting ground infrastructure, or state-backed teams able to combine cyber intrusion, RF interference, spoofing, and supply-chain manipulation [5], [6], [24]. CubeSats reduce the cost of entry to space, but they also reduce the cost of entry for attackers by normalizing commodity software, public documentation, and modular hardware ecosystems [8].

Attack surfaces can be organized by segment:

- **Space segment:** onboard computers, flight software, payload interfaces, software-defined radios, update mechanisms, key stores, and attitude or propulsion control paths.
- **Link segment:** telecommand uplinks, telemetry downlinks, ranging or synchronization services, relay links, and any protocol layers used for authentication, replay protection, or confidentiality.
- **Ground and user segment:** mission-control software, operator endpoints, cloud data services, CI/CD pipelines, test equipment, key-management systems, and third-party network providers.

Figure 2 places these segments alongside the cross-cutting dependencies that the literature keeps returning to: supply-chain provenance, physical custody during distributed integration, and the trust relationships among the organizations that build and operate the mission.

CubeSat attack trees are valuable because they reframe these surfaces as mission paths instead of device inventories. Falco, Viswanathan, and Santangelo show how compromises can propagate through development, launch preparation, communications, and operations rather than appearing only as isolated technical flaws [12]. This is consistent with broader space-security analyses that stress the coupling between cyber and physical effects: a digital attack on command authorization, ephemeris data, or collision-warning inputs can alter attitude, expend propellant, deny payload service, or degrade constellation behavior [6], [24].

Two threat-model characteristics are particularly CubeSat-specific. Mission assurance often depends on a narrow margin: a larger satellite may absorb partial degradation through redundancy or operational workarounds, while a CubeSat with a single computer, a single radio chain, and limited energy storage may not [2], [19]. Trust is also frequently federated across organizations. University-built payloads, commercial buses, shared ground stations, launch integrators, and cloud analytics vendors can all sit inside the effective mission boundary. The relevant question is therefore not simply whether the spacecraft is secure, but whether the mission's trust relationships have been explicitly modeled and bounded [18], [25].

IV. VULNERABILITIES

A. COMMAND AND TELEMETRY LINKS

The most widely recognized CubeSat vulnerability is insufficient protection of telecommand and telemetry links. SPD-5 treats unauthorized access to critical space vehicle functions,

TABLE 1. How this overview relates to recent satellite-security reviews.

Review	Primary scope	CubeSat emphasis	Emphasis of this overview by comparison
Kavallieratos and Katsikas [21]	Space-systems security literature	Limited	Reads the field through small-mission constraints rather than cataloguing it at large
Salim *et al.* [6]	Space, link, and ground segments	Partial	Keeps the segment model but foregrounds lifecycle, custody, and small-team economics
Verma [5]	Satellite communication threats and controls	Partial	Extends beyond the link to onboard software, supply chain, and validation culture
Stojnic *et al.* [14]	IoT-driven satellite networks	Partial	Centers the CubeSat mission rather than the networked constellation
Sharmin *et al.* [22]	Space information networks	Limited	Separates what the evidence supports from what remains aspirational
Wang *et al.* [7]	Satellite-network cybersecurity	Partial	Adds an explicit account of adoption barriers and evidence maturity for small missions

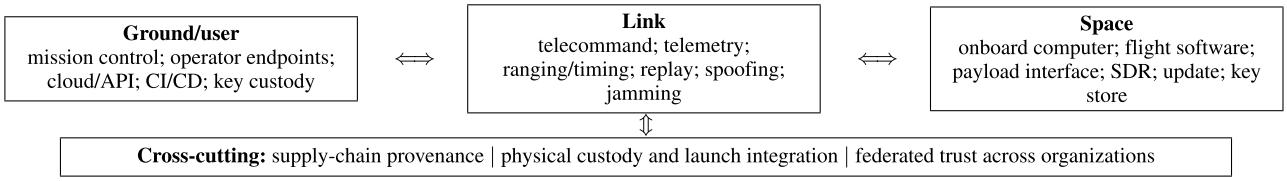

FIGURE 2. CubeSat mission attack surface across the space, link, and ground or user segments. Supply-chain provenance, physical custody, and cross-organization trust cut across all three rather than belonging to any single segment.

jamming, spoofing, and weak protection of command channels as foundational risks [4]. CCSDS has standardized data-link-layer protections through the Space Data Link Security Protocol, but adoption is uneven, especially in low-cost missions where bespoke or simplified radio stacks remain common [26]. Where authentication is absent or weak, the impact is immediate: an attacker may inject commands, replay valid traffic, manipulate timing, or undermine operator confidence in the control channel [5], [6], [14]. The persistence of this concern from early lightweight-protection proposals to recent network-level reviews suggests that command-link insecurity is not a transitional weakness; it remains a structural failure mode [7], [13].

CubeSats are exposed because communications subsystems are often selected for affordability, integration speed, and power budget rather than for security features. Caudill argues that many small-satellite teams face a structural mismatch between the sophistication of the threat and the resources available to implement robust protections in-house [8]. The consequence is predictable: cryptography, authentication, and key rotation are deferred, partially implemented, or left to vendor defaults.

B. GROUND SEGMENT WEAKNESSES

Recent surveys consistently identify the ground segment as a likely route to mission compromise [5], [6], [22]. Ground stations, operator laptops, web dashboards, file-transfer services, and cloud analytics pipelines run on general-purpose IT infrastructure and inherit mainstream cyber risks. CISA's recommendations to space system operators emphasize malware infection, insider threats, insufficient segmentation, weak patching discipline, and inadequate asset inventory as recurrent concerns [11]. CubeSat programs amplify these issues through small staff, shared academic networks, and operational scripts or tools originally built for convenience rather than assurance.

A purely ''spacecraft-centric'' security posture fails at this boundary. A mission can have carefully designed flight software and still be vulnerable if command generation occurs on inadequately secured workstations or if signing keys live in poorly controlled environments. Salim et al. and Verma both stress that space, link, and ground vulnerabilities must be assessed together because the ground side often provides the least expensive path to a high-impact result [5], [6].

C. ONBOARD SOFTWARE AND HARDWARE CONSTRAINTS

CubeSats commonly rely on commercial off-the-shelf processors, boards, operating systems, and middleware. That reuse speeds development but widens the attack surface to include inherited software flaws, insecure update mechanisms, weak isolation between payload and bus functions, and undocumented vendor assumptions [2], [6]. Limited processing headroom can also make cryptographic protocols, high-fidelity logging, and runtime anomaly detection compete directly with mission performance and power consumption [8], [9]. Willbold et al. strengthen this concern experimentally by showing that satellite software often lags conventional secure software engineering practice, which turns onboard software quality into a first-order cyber issue rather than a secondary implementation detail [15].

Falco's work on satellite-to-satellite cyber attack also highlights a broader concern: once autonomy, inter-satellite coordination, or increasingly programmable radios

are introduced, compromise paths need not remain confined to the original trusted ground operator [20]. Even where such attacks remain mostly conceptual in today's CubeSat deployments, they reveal how software flexibility can become an attack multiplier when the trust model is under-specified.

D. SUPPLY CHAIN AND INTEGRATION RISK

Supply-chain risk is central in both SPD-5 and CISA guidance [4], [11]. CubeSat programs often integrate components from multiple vendors and may inherit undocumented firmware, limited provenance data, or counterfeit-risk exposure. The concern is not limited to malicious implants. Even benign inconsistencies in component behavior, library versions, or undocumented interfaces can undermine the assumptions on which authentication, fault handling, and recovery procedures depend. Because CubeSat missions are often integrated rapidly, opportunities for deep supplier assessment and independent verification are limited [8], [9].

Physical custody belongs in the same discussion. A CubeSat typically passes through several hands before launch: university laboratories, environmental-test facilities, integrators, and deployer providers. Each handoff is an opportunity to access debug interfaces, replace firmware, or read out stored keys, and processors handling cryptographic material can leak information through power or timing behavior when an attacker has physical access during testing. These are surface exposures rather than demonstrated CubeSat exploits, and the practical response is custody discipline: record who handled trusted hardware, lock or authenticate debug paths, and verify firmware and keys at each transfer.

E. VALIDATION GAPS

Weak evidence is itself a vulnerability. Much of the field still relies on conceptual reasoning, tabletop attacks, or lab demonstrations that are difficult to reproduce across platforms [7], [17], [21]. Without comparable testbeds and public benchmarks, defenders cannot easily determine whether a proposed control is effective under realistic latency, power, and RF conditions. Sandbox and honeypot efforts create reusable observational infrastructure, but they remain exceptions rather than common research infrastructure [16], [27]. The validation gap encourages paper compliance and design optimism instead of operational assurance.

V. DEFENSES

A. SECURE-BY-DESIGN MISSION ENGINEERING

The strongest defensive move is architectural: security decisions must occur before integration has made them expensive or impossible. SPD-5 states that cybersecurity should be integrated before launch and sustained across the full mission lifecycle [4]. For CubeSats, this means threat modeling during architecture definition, identifying critical functions and positive-control paths, and treating command authorization, key management, and update security as mission requirements rather than optional enhancements. Visner and Kordella make a similar argument in their small-satellite best-practices paper: effective guidance should be adapted from mature critical sectors and embedded into development and operations, not bolted on after integration [9].

B. PROTECTING COMMAND PATHS

Command protection deserves disproportionate attention because telecommand compromise often dominates mission risk. At a minimum, missions should provide strong authentication, replay resistance, protected key storage, and a defined recovery path for loss of communications or partial key compromise [4], [26]. CCSDS 355.0-B-2 offers a standardized way to provide data authentication and confidentiality at the data-link layer for telemetry and telecommand protocols [26]. For CubeSats, such standards matter beyond interoperability: they reduce the temptation to design ad hoc security wrappers that have not been widely reviewed.

Standards adoption alone is insufficient. Security mechanisms must fit realistic size, weight, power, and timing budgets. That may require mission-specific tradeoffs such as authenticating all command traffic while encrypting only selected data flows, or provisioning offline emergency recovery procedures that preserve positive control without creating backdoor access [9], [11]. The design question is where limited mission resources buy the largest reduction in irreversible failure modes.

C. GROUND-SEGMENT HARDENING AND ZERO TRUST

Ground security is mission security. CISA's recommendations emphasize segmentation, patching, physical and logical access control, media restrictions, monitoring, and training [11]. Its zero-trust paper extends that logic by arguing that trust decisions in space systems should be continuously evaluated across identities, devices, workloads, and communications paths rather than assumed from network location alone [25]. This framing fits CubeSat missions that increasingly depend on cloud services, shared ground networks, and geographically distributed operators.

A CubeSat zero-trust posture would include strong identity and role separation for mission operators, dedicated signing workflows for command generation, device attestation where feasible, strict separation between development and operational networks, and auditable procedures for emergency access. These controls are operationally mundane, but they address the part of the mission where current attacks are most likely to succeed [6], [11].

D. TESTING, RED TEAMING, AND REPRODUCIBLE EVALUATION

Recent testbed work responds to the field's weak validation culture. Peled et al. describe an open-source and reproducible testbed intended to make satellite-security experiments portable and comparable across institutions [17]. Santangelo's LinkStar sandbox and HoneySat's network-based

honeypot approach point to the same methodological direction: defenses should be exercised in reusable, observable environments that let researchers compare attack paths and mitigations rather than report isolated demonstrations [16], [27]. ESA has also invested in a dedicated cybersecurity laboratory and in-orbit experimentation capabilities, including facilities for end-to-end security assessment and the CyberCUBE initiative for cybersecurity experiments on a 3U satellite platform [18], [28]. These efforts matter because many proposed defenses have never been tested under realistic RF impairments, resource constraints, and operational procedures.

CubeSat test and evaluation should therefore include adversarial exercises: authenticated-command fuzzing, replay attempts, degraded-link scenarios, compromised-ground-node drills, and recovery exercises for lost keys or corrupted state. Security that cannot be rehearsed is unlikely to be dependable in orbit.

Table 2 draws the vulnerabilities and defenses together by mapping each recurring threat area to the controls the literature recommends, the mission segment most affected, and the kind of evidence currently behind the recommendation. The pattern worth noting is that the strongest near-term controls protect command authority and ground workflows, while the evidence behind them is still mostly policy guidance and laboratory work rather than flight results.

VI. STANDARDS AND POLICY LANDSCAPE

The standards landscape is fragmented but increasingly usable. At the policy level, SPD-5 remains the clearest high-level statement of what secure space-system engineering should protect: positive control, critical functions, command links, ground systems, cybersecurity hygiene, and supply-chain integrity [4]. Its value for CubeSats lies in forcing small missions to articulate minimum survivability and recovery requirements, not just nominal mission objectives.

At the technical standard level, CCSDS is the central reference point for interoperable space-data security. CCSDS 355.0-B-2 defines the Space Data Link Security Protocol and supports authentication and confidentiality protections for standard telemetry and telecommand links [26]. This does not solve all mission-security problems, but it gives programs a common foundation for protecting one of the most critical interfaces in any spacecraft system.

Operational guidance has expanded in parallel. CISA's 2024 recommendations adapt mainstream cyber controls to the space domain and are especially relevant to CubeSat operators because they address the mundane but decisive issues of segmentation, patch management, monitoring, insider risk, and asset management [11]. The companion zero-trust paper reframes space missions as distributed systems whose security depends on continuous trust evaluation across ground, link, and mission components [25].

ESA's recent activity suggests that European practice is moving toward more operationalized space cybersecurity as well. ESA now maintains a dedicated cybersecurity laboratory for testing and validation and has begun sponsoring in-orbit cyber experimentation through CyberCUBE and its Security for Space Systems ecosystem [18], [28], [29]. These are not standards in the formal CCSDS sense, but they help fill a critical gap between policy aspiration and implementable engineering evidence.

The main weakness is adoption rather than availability. CubeSat missions can now draw on policy principles, link-security standards, and operational guidance, yet many still lack the staffing, schedule margin, or procurement leverage to implement them consistently. Standardization helps, but it does not remove the economic asymmetry that Caudill identified: for small programs, shared infrastructure and reusable security services may be more realistic than expecting each mission to independently build a mature cyber program [8]. Review papers published in 2025 and 2026 repeatedly converge on this point: standards exist, but implementation maturity remains the bottleneck [7], [14], [22].

Table 3 lays out the main instruments a CubeSat team can draw on, what each one offers, and why adoption tends to stall. The barriers are less about the content of the guidance than about the conditions small programs work under: limited staff, compressed schedules, radios chosen for cost rather than security features, and little leverage over suppliers. Naming these barriers explicitly is more useful to a small team than repeating that the guidance exists.

VII. LITERATURE SYNTHESIS AND DISCUSSION

CubeSat cybersecurity is no longer a niche concern. Broader mission adoption, increasing software programmability, and tighter integration with terrestrial digital infrastructure have made cyber risk inseparable from mission assurance [2], [4], [6], [7]. The field has moved beyond asking whether satellites can be attacked and toward identifying where engineering effort has the highest defensive return.

The literature remains unbalanced. Conceptual threat papers, policy guidance, and broad surveys are now plentiful, but flight-validated evidence remains scarce [7], [14], [17], [21]. This matters because many security controls that look attractive on paper impose latency, power, operational-complexity, or failure-recovery costs that are significant on small spacecraft. CubeSat research needs fewer abstract control lists and more evidence about what actually works on realistic platforms.

The dominant risk narrative should shift from ''protect the satellite'' to ''secure the mission system''. The strongest recent surveys repeatedly show that the ground and link segments are often the more attractive and accessible targets [5], [6]. That conclusion is consistent with CISA's focus on operator workflows and infrastructure hygiene [11]. For CubeSat teams, some of the highest-value security work may involve operator identity, command release processes, key custody, and cloud-service boundaries rather than exotic onboard defenses.

TABLE 2. Recurring threat areas, Controls discussed in the literature, and the evidence behind them.

Threat area	Segment	Representative controls in the reviewed work	Evidence basis
Unauthenticated or replayed commands	Link	Command authentication, replay resistance, protected key storage, defined recovery path [4], [13], [26]	Standard and early prototype
Ground-segment compromise	Ground/user	Segmentation, identity and role separation, dedicated command-signing workflow, zero-trust evaluation [11], [25]	Operational guidance
Onboard software and update flaws	Space	Secure-by-design engineering, isolation between payload and bus, controlled update mechanisms [4], [9], [15]	Guidance and one software-security experiment
Supply-chain and integration risk	Cross-cutting	Provenance records, component assessment, integration-time verification [4], [8], [11]	Policy and conceptual
Weak validation and unproven defenses	Cross-cutting	Reproducible testbeds, sandboxes, honeypots, adversarial exercises [16], [17], [27]	Emerging reusable infrastructure

TABLE 3. Available guidance for cubesat security and why adoption lags.

Instrument	What it provides	Why adoption lags in small missions
SPD-5 [4]	High-level principles: positive control, protected command and telemetry, supply-chain integrity, lifecycle cybersecurity	States objectives, not implementations; leaves each mission to translate principles into a specific radio, processor, and operations design
CCSDS 355.0-B-2 [26]	Standardized data-link authentication and confidentiality for telemetry and telecommand	Low-cost radios may not expose the required functions; key management and interoperability testing add integration effort
CISA recommendations and zero trust [11], [25]	Operational controls: segmentation, patching, monitoring, identity, continuous trust evaluation	Small teams lack staff and tooling for continuous monitoring and strict separation of duties
ESA laboratory and in-orbit programs [18], [28], [29]	Test infrastructure and in-orbit cyber experimentation	Access is uneven across programs and regions; results are not yet packaged as reusable mission guidance
Shared security infrastructure [8]	A way to amortize specialist effort across many small missions	Few mature shared services exist, and sharing introduces concentration risk that must be managed

Different papers answer different questions, often with different units of analysis, from packet flows to mission governance. Methodologically, the field can be divided into at least four strands: lightweight protocol and communications protection [13]; mission-oriented conceptual analysis and best-practice synthesis [9], [12], [20]; empirical software or systems-security experimentation [15], [17]; and broad network or ecosystem surveys [5], [6], [7], [14], [22]. This diversity of aims explains some of the apparent inconsistency across studies: they are often answering different questions rather than disagreeing.

The selected literature reflects the same maturation path. Earlier work on best practices and conceptual attack modes established the problem framing [9], [20]. Attack-tree analysis and sandbox efforts introduced more structured mission decomposition and community experimentation [12], [16]. Empirical software analysis and reproducible testbeds suggest that the field is starting to build the evidence base needed for disciplined evaluation [15], [17]. The trajectory is promising, but still early.

Part of reading this literature well is keeping track of what each kind of source can and cannot settle. A policy document can establish that a control is expected; it cannot show that the control works on a 3U platform under realistic power and RF limits. A reproducible testbed can compare attacks and defenses; it speaks to flight behavior only when its hardware and operations resemble the mission. Table 4 sets out these evidence types and their limits so that recommendations in this overview are read at the strength their sources actually support. For most controls, the honest summary is that the case rests on guidance and laboratory work, with flight-validated evidence still scarce.

VIII. OPEN PROBLEMS

A. LIFECYCLE KEY MANAGEMENT

CubeSat missions still lack well-documented, widely reusable patterns for generating, loading, rotating, escrowing, and recovering cryptographic keys across development, launch, commissioning, operations, anomaly response, and disposal. This gap matters because command-link protection is only as strong as the surrounding key process [4], [26].

B. SECURITY FOR AUTONOMY AND CONSTELLATIONS

As CubeSats become more autonomous and more commonly deployed in constellations, cyber risk shifts from single-node compromise toward coordinated or cascading effects. The satellite-to-satellite attack literature is still relatively speculative, but it correctly anticipates that cross-satellite trust, autonomous decision making, and inter-satellite communications will become central security questions [20], [24].

C. EVIDENCE AND BENCHMARKING

The shortage of reproducible benchmarks remains acute. Open testbeds such as the one proposed by Peled et al.

TABLE 4. Types of evidence in the cubesat security literature and their limits.

Evidence type	Representative work	What it supports	What it does not establish
Policy and standard	SPD-5, CCSDS, CISA guidance [4], [11], [26]	Control objectives and protocol requirements	Feasibility or effectiveness on a specific CubeSat
Conceptual and attack-tree	Attack trees and best-practice synthesis [9], [12], [20]	Structured attack paths and candidate designs	Measured resistance or mission overhead
Software or systems experiment	Satellite-software security analysis [15]	Concrete vulnerability classes in real artifacts	End-to-end mission resilience
Reusable testbed and sandbox	Portable testbeds, sandboxes, honeypots [16], [17], [27]	Comparable, repeatable experiments	Flight validation unless hardware and operations match
Broad survey	Network and ecosystem reviews [5]–[7], [21]	Breadth and cross-study patterns	Depth or primary measurement of any single control

are promising, and ESA's in-orbit experimentation initiatives are a useful complement, but the community still lacks common datasets, attack traces, and cost models that would let researchers compare defenses rigorously [17], [28]. Sandbox and honeypot efforts show how this gap might be closed, but they have not yet become common infrastructure for the field [16], [27]. Without that infrastructure, claims of resilience are difficult to validate.

D. SECURITY ECONOMICS FOR SMALL MISSIONS

CubeSat cybersecurity is constrained as much by economics as by technology. Small teams often cannot sustain dedicated security operations, independent verification, or bespoke secure hardware. Caudill's proposal for secure infrastructure as a service remains relevant because it addresses the organizational scale mismatch that many CubeSat missions face [8]. A major open question is which security capabilities can be centralized or shared without introducing new concentration risks.

E. FUTURE CRYPTOGRAPHY AND LONG MISSION LIFETIMES

Emerging secure-communications initiatives, including ESA's work on space-based quantum key distribution, show that the secure-communications landscape is evolving [30]. However, most CubeSat missions need practical near-term answers: lightweight authentication, robust replay protection, secure updates, and migration paths for cryptographic agility. Post-quantum migration for resource-constrained spacecraft is still largely unresolved in openly published CubeSat engineering guidance.

IX. CONCLUSION

CubeSat cybersecurity should be understood as a mission-assurance problem shaped by small-spacecraft constraints, not as a simplified version of legacy satellite security. The literature now provides a clearer picture of the threat surface: vulnerable command paths, weakly defended ground systems, inherited supply-chain exposure, and limited validation culture are the recurring risk drivers [5], [6], [7], [9], [12]. At the same time, the available defenses are no longer purely aspirational. SPD-5, CISA guidance, CCSDS data-link security standards, and emerging reproducible testbeds together provide a workable foundation for more disciplined practice [4], [11], [17], [26].

The field now has enough breadth to support comparative synthesis, but not enough empirical depth to claim a settled state of the art. The next step is not another generic warning that satellites are vulnerable. It is the construction of repeatable engineering evidence, mission-ready security patterns, and economically viable deployment models for small satellites operators. CubeSats are now consequential enough that security cannot remain a research afterthought or a procurement appendix. For many missions, it is part of the core system design.

REFERENCES

[1] National Aeronautics and Space Administration. (Feb. 2026). *What Are Smallsats and Cubesats?*. [Online]. Available: https://www.nasa.gov/what-are-smallsats-and-cubesats

[2] NASA Small Spacecraft Systems Virtual Institute. (Feb. 2025). *State-of-the-Art of Small Spacecraft Technology*. [Online]. Available: https://www.nasa.gov/smallsat-institute/sst-soa

[3] National Institute of Standards and Technology. (May 2015). *A Profile for U.S. Federal Cryptographic Key Management Systems*. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-152.pdf

[4] Executive Office of the President of the United States. (Sep. 2020). *Memorandum on Space Policy Directive-5—Cybersecurity Principles for Space Systems*. [Online]. Available: https://trumpwhitehouse.archives.gov/presidential-actions/memorandum-space-policy-directive-5-cybersecurity-principles-space-systems/

[5] A. R. K. Verma, "Cybersecurity in satellite communication networks: Key threats and neutralization measures," *IEEE Open J. Commun. Soc.*, vol. 6, pp. 5667–5692, 2025.

[6] S. Salim, N. Moustafa, and M. Reisslein, "Cybersecurity of satellite communications systems: A comprehensive survey of the space, ground, and links segments," *IEEE Commun. Surveys Tuts.*, vol. 27, no. 1, pp. 372–425, Feb. 2025.

[7] B. Wang, J. Xiao, R. Dong, C. Piao, Y. Guan, B. Zhao, Y. Yang, Z. Zhao, S. Li, and X. Lyu, "A comprehensive literature review of cybersecurity in satellite networks," *Aerospace*, vol. 13, no. 3, p. 249, Mar. 2026.

[8] H. Caudill, "Big risks in small satellites: The need for secure infrastructure as a service," in *Proc. AIAA ASCEND*, Nov. 2020, pp. 1–6, 8.

[9] S. Visner and S. Kordella. (Nov. 2020). *Cyber Best Practices for Small Satellites*. [Online]. Available: https://www.mitre.org/news-insights/publication/cyber-best-practices-small-satellites

[10] S. Higginbotham. (2017). *Cubesat Launch Initiative Overview and Cubesat 101*. [Online]. Available: https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20170010294.pdf

[11] Cybersecurity and Infrastructure Security Agency. (Jun. 2024). *Recommendations to Space System Operators for Improving Cybersecurity*. [Online]. Available: https://www.cisa.gov/resources-tools/resources/recommendations-space-system-operators-improving-cybersecurity

[12] G. Falco, A. Viswanathan, and A. Santangelo, "CubeSat security attack tree analysis," in *Proc. IEEE 8th Int. Conf. Space Mission Challenges Inf. Technol. (SMC-IT)*, Jul. 2021, pp. 68–76.
[13] O. Challa, G. Bhat, and J. Mcnair, "Cubesec and gndsec: A lightweight security solution for cubesat communications," in *Proc. 26th Annu. AIAA/USU Conf. Small Satell.*, 2012, pp. 1–2, 4, 7. [Online]. Available: https://digitalcommons.usu.edu/smallsat/2012/all2012/25/
[14] T. Stojnic, A. S. M. Kayes, W. Rahayu, and M. J. M. Chowdhury, "A comprehensive literature review of cyber threats and vulnerabilities in IoT-driven satellite networks: Research challenges and future directions," *Comput. Netw.*, vol. 272, Nov. 2025, Art. no. 111678.
[15] J. Willbold, M. Schloegel, M. Vögele, M. Gerhardt, T. Holz, and A. Abbasi, "Space odyssey: An experimental software security analysis of satellites," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2023, pp. 1–19.
[16] A. Santangelo, "The linkstar cybersecurity, "The linkstar cybersecurity," in *Proc. 35th Annu. Small Satell. Conf.*, 2021, pp. 1–2, 5–8. [Online]. Available: https://digitalcommons.usu.edu/smallsat/2021/all2021/56/
[17] R. Peled, R. Idan, T. Galor, O. Chodeda, E. Marcus, A. Shabtai, and Y. Elovici, "A reproducible and open-source testbed for satellite cybersecurity," in *Proc. 3rd ACM Conf. Reproducibility Replicability*, Jul. 2025, pp. 51–61.
[18] European Space Agency. (2026). *Cybersecurity Laboratory*. [Online]. Available: https://technology.esa.int/lab/cybersecurity-laboratory
[19] G. M. Lehto, G. Edlund, T. Smigla, and F. Afinidad, "Protection evaluation framework for tactical SATCOM architectures," in *Proc. IEEE Mil. Commun. Conf.*, Nov. 2013, pp. 1008–1013.
[20] G. Falco, "When satellites attack: Satellite-to-satellite cyber attack, defense and resilience," in *Proc. AIAA ASCEND*, Nov. 2020, pp. 2, 5, 7.
[21] G. Kavallieratos and S. Katsikas, "An exploratory analysis of the last frontier: A systematic literature review of cybersecurity in space," *Int. J. Crit. Infrastructure Protection*, vol. 43, Dec. 2023, Art. no. 100640.
[22] A. Sharmin, B. U. Mahmud, N. Nabi, M. Shaima, and M. J. H. Faruk, "Cyber attacks on space information networks: Vulnerabilities, threats, and countermeasures for satellite security," *J. Cybersecurity Privacy*, vol. 5, no. 3, p. 76, 2025.
[23] R. C. Jones, "Survey of space professionals' perception of satellite cybersecurity from 2012 to 2022: Decision-makers' thoughts on satellite cybersecurity evolving," *New Space*, vol. 12, no. 4, pp. 259–272, Dec. 2024.
[24] J. Pavur and I. Martinovic, "The cyber-ASAT: On the impact of cyber weapons in outer space," in *Proc. 11th Int. Conf. Cyber Conflict (CyCon)*, vol. 900, May 2019, pp. 1–18.
[25] Cybersecurity and Infrastructure Security Agency. (Jun. 2024). *Space System Security and Resilience Landscape: Zero Trust in the Space Environment*. [Online]. Available: https://www.cisa.gov/resources-tools/resources/space-system-security-and-resilience-landscape-zero-trust-space-environment
[26] Consultative Committee for Space Data Systems. (Aug. 2022). *Space Data Link Security Protocol*. [Online]. Available: https://ccsds.org/publications/allpubs/entry/3258/
[27] E. López-Morales, U. Planta, G. Marra, C. González, J. Hopkins, M. Garoosi, E. Obreque, C. Rubio-Medrano, and A. Abbasi, "HoneySat: A network-based satellite honeypot framework," 2025, *arXiv:2505.24008*.
[28] European Space Agency. (2025). *Call for Ideas: Cybersecurity Experiments in Orbit*. [Online]. Available: https://security4space.esa.int/2025/cfi-cybersecurity-experiments-in-orbit/
[29] (2025). *Security for Space Systems (3S) Conference*. [Online]. Available: https://security4space.esa.int/
[30] (2026). *Eagle-1*. [Online]. Available: https://www.esa.int/Applications/ConnectivityandSecureCommunications/Eagle-1

JOÃO PAOLO CAVALCANTE MARTINS OLIVEIRA received the B.S. degree in computer science from the University of Fortaleza, Brazil, in 2025. He is currently pursuing the M.S. degree in aerospace engineering with the Federal University of Rio Grande do Norte, Brazil. From 2011 to 2017, he was a Research Assistant with the Federal Institute of Ceará, Brazil. He has been working in many fields of computer science, since 2001. His research interests include computer security, machine learning, embedded systems, space technologies, and astrophysics.

FRANCISCO JOSÉ TARGINO VIDAL received the M.Sc. and D.Sc. degrees in electrical and computer engineering from the Federal University of Rio Grande do Norte (UFRN), Natal, Rio Grande do Norte, Brazil. He is currently an Associate Professor with UFRN. His research interests include signal processing for communications, reconfigurable hardware for data and signal processing, machine learning, and nanosatellite technologies.

FÁTIMA MATTIELLO-FRANCISCO received the degree in computer science from ICMC-USP and the M.Sc. degree in electronic and telecommunication engineering, fault tolerance field, from the National Institute for Space Research (INPE), Brazil. She is currently pursuing the Ph.D. degree in computing engineering with the Aeronautical Institute of Technology (ITA). She is a Senior System Engineer with INPE. She has experience in designing ground systems for satellite operation and in testing embedded software. As a Space system Engineer, she worked on ground systems development, mission concept of operation, and verification and validation of software-intensive systems. She is a Project Manager of the nanosatellite CubeSat developed by INPE in cooperation with universities. She is a Lecturer of the Space System Engineering Post-Graduation School, INPE. Her current research interests include model-based testing, model-based system engineering, dependable computing, verification, and validation techniques with an emphasis on integration testing of real-time critical systems and empirical studies.