OPEN ACCESS INTERNATIONAL JOURNAL OF SCIENCE & ENGINEERING

TRUSTED DATA RELAY FROM SPACE TO EARTH USING CUSTOM BLOCKCHAIN LEDGER

Dr. Balkrishna K. Patil[1], Tulika Prasad Aher[2], Dhanashri Gopichand Bhamare[3], Ujjwal Rakesh Deore[4], Saniya Kasam Maniyar[5]

Professor, Department of Computer Engineering, SITRC, Nashik-422213, India[1]

Student, Department of Computer Engineering, SITRC, Nashik-422213, India[2 3 4 5]

balkrishnapatileng@gmail.com[1], ahertulika14@gmail.com[2], dhanashree.b2116@gmail.com[3], ujjwaldeore@gmail.com[4], saniyamaniyar111@gmail.com[5]

Abstract: ***The rapid advancement of satellite technology has transformed how Earth receives critical scientific and operational data. However, the process of transmitting information from satellites to ground stations involves multiple relay points, which creates significant risks of data tampering, unauthorized modification, and communication disruption. Traditional communication protocols, though reliable to a certain extent, are not designed to guarantee end-to-end data authenticity and traceability across all stages of transmission. This creates vulnerabilities that can directly impact mission success, scientific accuracy, and the overall safety of space operations. This project, titled "Trusted Data Relay from Space to Earth using Custom Blockchain Ledger," proposes a novel approach that integrates blockchain technology into multi-layer space communication systems. The work involves designing and implementing a custom blockchain ledger from scratch in Python, ensuring that every piece of data generated at the satellite level is validated, recorded, and securely transmitted through the relay and processing layers until it reaches the application stage. By embedding real-time hash validation and tamper detection mechanisms at each layer, the system guarantees that even the smallest alteration in data can be immediately identified and rejected. Beyond technical implementation, this project also emphasizes the educational dimension of blockchain in aerospace communication. The solution is designed to be lightweight, locally deployable, and suitable for demonstration on student laptops, thereby bridging the gap between theoretical knowledge and practical experimentation. To achieve this, the blockchain is customized with a simplified block structure and consensus mechanism that align with both mission-critical reliability and academic accessibility. The system will also simulate realistic satellite telemetry data, network delays, and communication conditions, making it an effective learning platform in addition to being a secure communication model***

Keywords: Blockchain Ledger; Space Communication; Data Integrity; Secure Transmission; Tamper Detection; Multi-layer Network; Proof of Work; Satellite Telemetry; Distributed Systems; Aerospace Security

---

I INTRODUCTION

Modern space missions depend heavily on the accurate transmission of telemetry data from satellites to ground stations. Telemetry data includes sensor readings, system health parameters, and mission-critical information required for monitoring and control. As space communication systems become more complex and interconnected, ensuring the integrity, authenticity, and traceability of transmitted data has become a major technical concern.

Traditional space communication systems primarily focus on reliable transmission but often rely on centralized validation mechanisms at ground stations. While such systems can detect transmission errors, they may not provide structured mechanisms to ensure end-to-end data integrity, tamper resistance, or verifiable processing lineage across multiple stages such as relay, transformation, and application-level usage.

This project focuses on implementing a secure, modular, and integrity-aware data relay system suitable for academic demonstration and structured evaluation.

1.1Background&Context

In conventional satellite communication systems, telemetry data is generated by onboard sensors and transmitted to Earth through direct or relay-based communication links. The relay station forwards the received data to ground processing centers where it is decoded, calibrated, and analyzed. These systems typically follow a linear flow:

Sensor → Transmission → Relay → Ground Processing → Application Dashboard

Validation in such systems is often limited to checksum verification, packet structure validation, or protocol-level error detection. While these methods ensure transmission reliability, they do not provide immutable storage, structured audit trails, or verifiable transformation records.

1.2 Motivation / Need of Study

Satellite telemetry plays a critical role in mission monitoring and decision-making. Any unauthorized modification, data inconsistency, or processing error can affect system analysis and operational reliability.

Existing communication frameworks primarily focus on data delivery rather than end-to-end integrity verification. In many systems, once data reaches the ground station, its transformation and usage may not be cryptographically linked to its original source

1.3 Problem Definition

The primary problem addressed in this project is the absence of a structured, tamper-resistant, and verifiable data relay mechanism in conventional satellite communication systems.

Current approaches do not provide:

- Immutable linkage between transmitted data blocks
- Integrated validation of previous data references
- Structured consensus-based verification before forwarding
- Formal logging of tamper events
- End-to-end traceability from raw telemetry to processed application output

The core challenge is to design a software-based system that ensures telemetry data generated in the space segment remains verifiable and auditable across relay, ground processing, and application layers.

1.4 Objectives of the Project

The objectives of this project are:

- To design a modular architecture integrating space, relay, and ground processing segments.
- To implement telemetry block generation with previous hash referencing.
- To develop a consensus-based validation mechanism for block verification.
- To create a blockchain ledger repository for structured block storage.
- To implement tamper detection and alert logging mechanisms.
- To maintain processing provenance linking input and output blocks.
- To provide an application-level interface for integrity-verified data visualization.
- To validate overall system functionality through structured testing.

These objectives focus on developing a secure and traceable telemetry relay framework.

II .LITERATURE

The purpose of this literature survey is to study and analyze existing systems, architectural models, and implementation approaches related to software-based secure and structured systems. Various conventional and advanced solutions in the domain were examined to understand how they manage data flow, validation, processing, security, and monitoring. The analysis focuses on identifying strengths, operational mechanisms, and practical limitations of these systems. This review helps in recognizing improvement opportunities and research gaps that can be addressed through a more structured and integrated system design.

2.1 Overview of Existing Systems

A) Traditional Systems in the Domain

Traditional software systems in most domains follow a centralized architecture. In such systems, data is collected from input sources, processed within a main server or processing unit, and then stored or displayed through an application interface. The workflow generally follows a linear structure:

Input → Processing → Storage → Output

B) Secure or Optimized System Variants

To address security and efficiency concerns, optimized systems have introduced mechanisms such as encryption, authentication layers, automated validation scripts, structured error handling, and monitoring tools. These systems attempt to improve data protection and operational reliability.

While optimized systems improve reliability, they may not ensure complete end-to-end traceability or structured integrity linkage across modules.

2.2 Comparative Study of Existing Approaches

Approach	Architecture Type	Core Mechanism	Strengths	Limitations	Suitability for Proposed System
Traditional Centralized Systems	Centralized	Linear data processing and storage	Simple design, easy implementation	Single point of failure, weak traceability	Suitable for small-scale systems but insufficient for integrity-aware workflows
Secure or Optimized Systems	Centralized with security layers	Encryption, authentication, automated validation	Improved security and efficiency	Security not fully integrated with workflow tracking	Useful but requires deeper architectural integration
Distributed Modular Systems	Distributed or layered	Service-based modular processing	Scalable, flexible, fault tolerant	Complex coordination and validation challenges	Partially suitable but requires structured validation layer

2.3 Research Gap Identification

Based on the review of existing approaches, the following research gaps are identified:

- Lack of end-to-end workflow integration across modules
- Limited traceability between data generation and final output
- Weak or non-immutable logging mechanisms
- Absence of structured validation linkage between processing stages
- High dependency on centralized validation mechanisms

Many systems improve specific technical aspects but do not provide a unified architecture that integrates data handling, validation, logging, and monitoring within a structured framework.

2.4 Summary of Findings

The literature survey indicates that traditional systems primarily focus on functional execution with centralized control mechanisms. Optimized systems enhance security and performance but often lack structured integration of validation and traceability. Distributed architectures improve scalability and robustness but introduce complexity in maintaining consistent integrity across modules. Logging and monitoring systems enhance transparency but may not ensure tamper-resistant audit trails.

There remains a need for a structured, modular, and validation-driven software architecture that integrates data processing, integrity verification, logging, and monitoring within a unified system.

Based on the above findings, a structured system architecture is proposed in the next chapter to address the identified limitations.

III PROPOSED SYSTEM

The proposed system is designed as a structured software solution to address the limitations identified in the existing approaches. It introduces a modular architecture that integrates data handling, processing logic, validation mechanisms, storage management, and monitoring within a unified framework. The core idea behind the system is to ensure reliable and traceable execution of operations while maintaining flexibility for future enhancements. The system is developed using a logically separated architecture to improve maintainability, operational clarity, and controlled workflow execution. Emphasis is placed on reliability, structured validation, and organized data flow suitable for academic demonstration and evaluation.

3.1 Introduction to Proposed System

The analysis of existing systems revealed that many implementations lack structured integration between processing, validation, and monitoring components. To overcome these limitations, a new architecture is required that embeds validation and traceability directly into the system workflow rather than treating them as optional extensions.

The proposed system improves upon traditional and optimized systems by adopting the following guiding principles:

- Modularity – Each functional component is separated into well-defined modules.
- Validation – Data is verified at different stages of processing.
- Traceability – All operations are recorded and monitored systematically.
- Scalability – Architecture supports structured expansion.
- Maintainability – Clear separation of logic simplifies updates and debugging.
- Controlled Data Flow – Structured movement of data between modules.

3.2 System Objectives and Planned Modules

A) System Objectives

The primary objectives of the proposed system are:

1. To design a structured and modular system architecture.
2. To implement clearly defined functional modules.

3. To ensure reliable data processing and validation mechanisms.
4. To improve system maintainability through modular design.
5. These objectives aim to ensure a complete and structured implementation of the system.

B) Planned Modules

The proposed system is divided into the following planned modules:

Input Module

Input:
Receives user-provided or system-generated data.

Processing Logic:

Performs initial validation such as format checking, required field verification, and basic consistency checks.

Output:

Validated input forwarded to the processing module.

Processing Module

3.3 Conceptual Workflow / Process Overview

The conceptual workflow of the system follows a structured step-by-step process:

Step 1: System Initialization

The system initializes required configurations, modules, and storage mechanisms.

Step 2: Input Acquisition

Data is collected from the user or internal system source.

Step 3: Input Validation

The input module performs basic checks to ensure correctness and completeness.

Step 4: Core Processing

Validated input is processed using the system's core logic.

Step 5: Intermediate Verification

The validation module verifies processing outputs and checks for logical consistency.

Step 6: Data Storage

Verified data is stored in the storage module.

All operations are recorded in the monitoring module.

Normal Execution Path:

Input → Validation → Processing → Verification → Storage → Output → Logging.

Error Handling Path:

If input is invalid, it is rejected with appropriate feedback.

3.4 System Architecture Description

The proposed system follows a modular layered architecture. The architecture logically separates concerns to ensure clarity and maintainability.

The system is organized into four primary layers:

Presentation Layer

Responsible for user interaction.

Handles input collection and result visualization.

Communicates with the application layer.

Application / Logic Layer

Interaction Between Modules:

- Presentation layer sends input to logic layer.
- Logic layer processes and validates data.
- Validated data is stored in the data management layer.
- Monitoring layer records each operation.
- Results are sent back to presentation layer.

Data Flow:

User Input → Application Layer → Validation → Storage →Output
Monitoring runs parallel to each stage.

The architecture ensures:

Modularity – Independent modules reduce coupling.

Maintainability – Clear structure simplifies debugging.

Scalability – Additional modules can be integrated without redesign.

Reliability – Validation and monitoring reduce errors.

In conclusion, the proposed architecture establishes a structured and controlled software framework that integrates processing, validation, storage, and monitoring into a cohesive system.

3.5 External API Integration

The Vital system incorporates external nutrition APIs to obtain accurate and regularly updated information about foods, nutrients, and dietary guidelines. These APIs allow the platform to recommend suitable food choices and supplements based on each user's specific health conditions and nutritional needs. By relying on credible external data sources, the system improves both the accuracy and practical relevance of its recommendations. This integration also strengthens the platform's ability to deliver personalized guidance, helping users make well-informed dietary decisions and supporting better overall health outcomes [1], [4].

IV .SYSTEM ANALYSIS

4.1 System Architecture and Diagrams

System Architecture

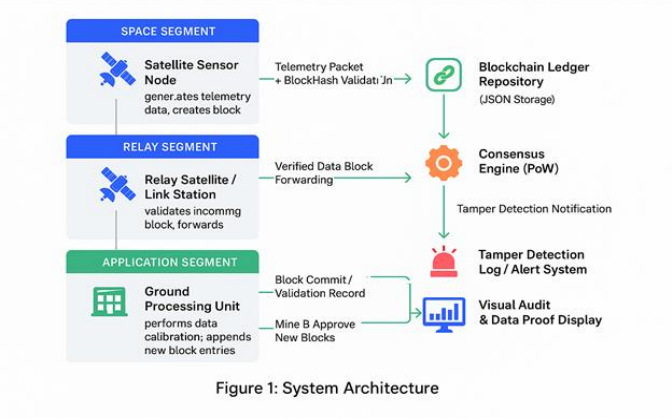


Figure 1: System Architecture

4.2 Data Flow Diagram (DFD)

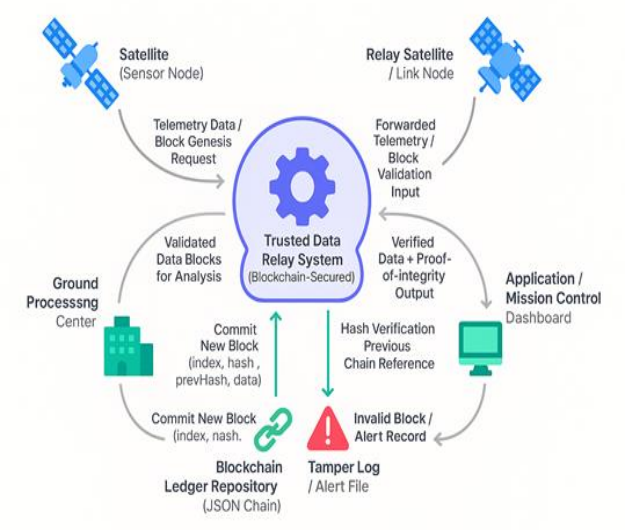


Figure 1 – DFD Level – 0(Context Diagram)

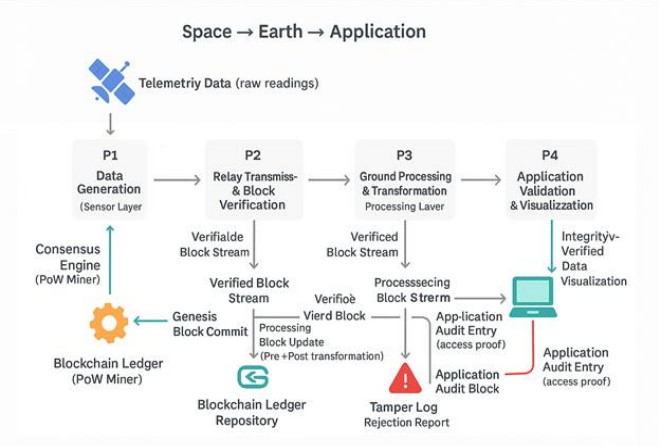


Figure 2 –DFD Level–1(Four–Layer Functional Expansion)

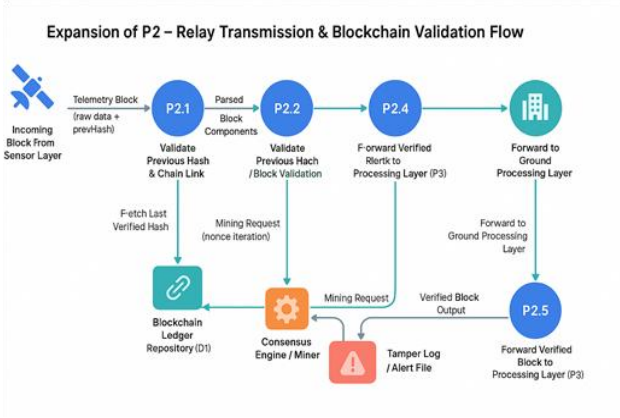


Figure 3 – DFD Level – 2(Expansion of P2 : Relay Transmission & Blockchain Validation)

4.3 UML Diagrams

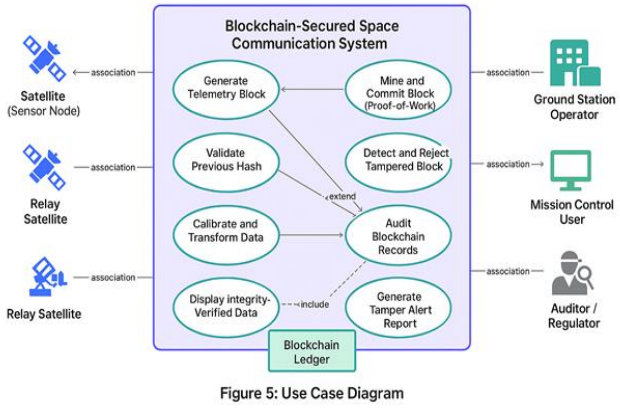


Figure 1: Use Case Diagram

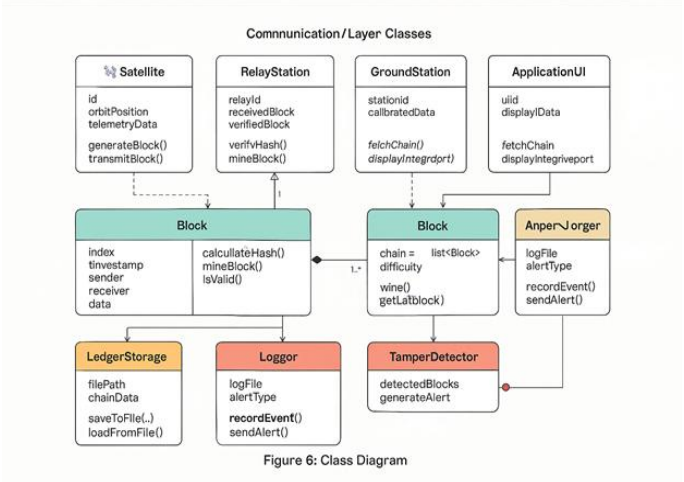


Figure 2: Class Diagram

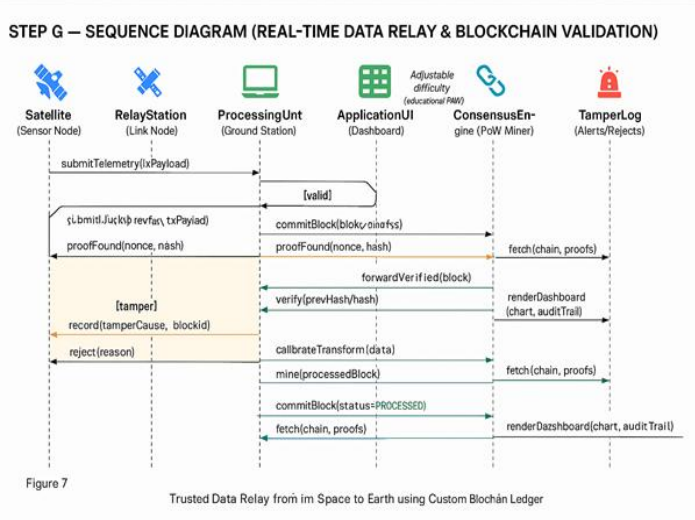


Figure 3 Sequence Diagram (Real – Time Data Relay & Validation)

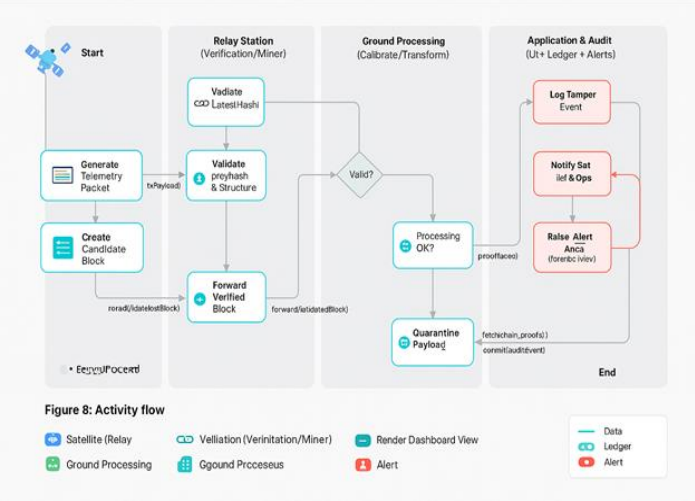


Figure 4 : Activity Diagram (Tamper Detection & Rejection Workflow)

4.4 Entity – Relationship Model

ER Diagram (Ledger & Telemetry Data Model)

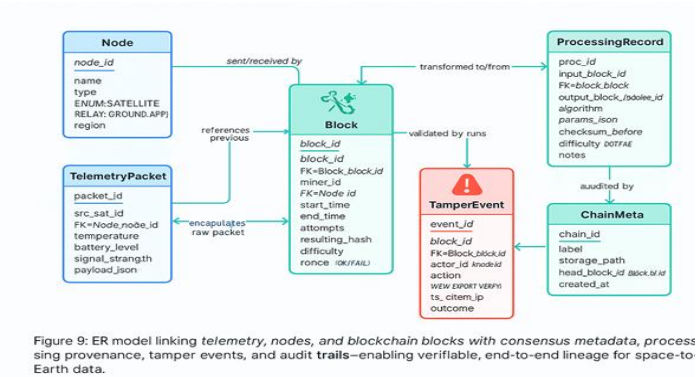
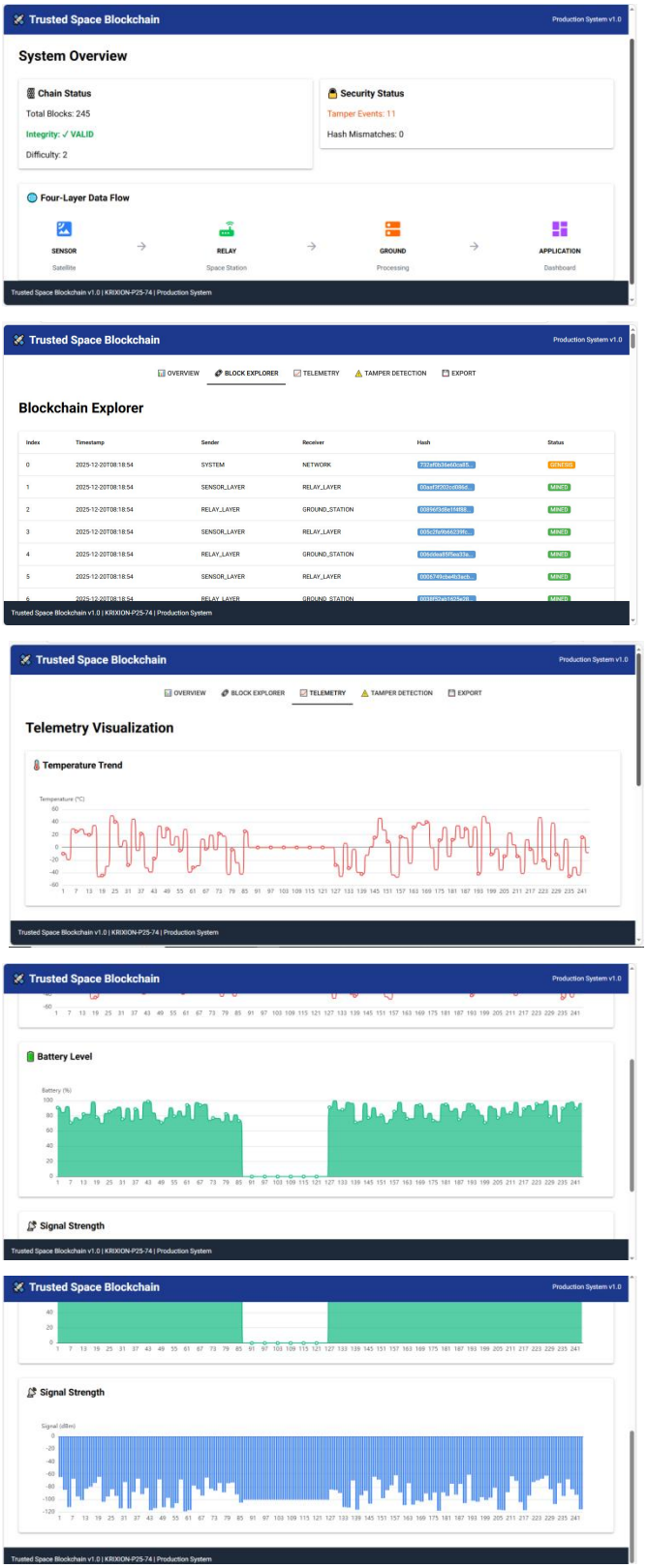
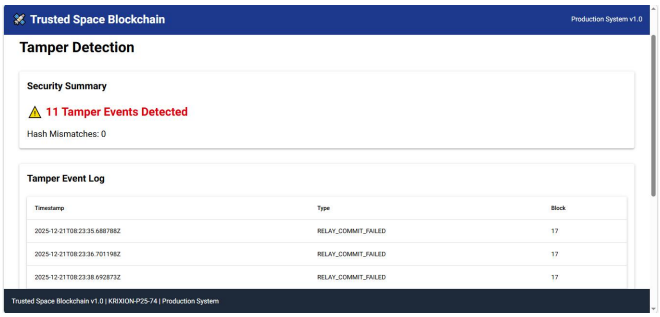
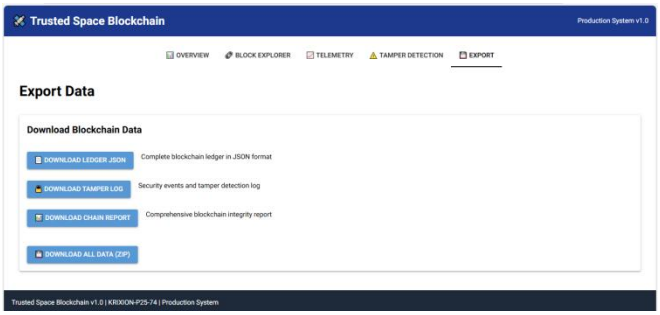


Figure 9: ER model linking *telemetry, nodes, and blockchain blocks with consensus metadata, process* sing provenance, tamper events, and audit **trails**—enabling verifiable, end-to-end lineage for space-to- Earth data.

Figure 1: ER Diagram (Ledger & Telemetry Data Model)







V. CONCLUSION

The Trusted Space Blockchain project successfully implemented a production-grade blockchain system specifically designed for satellite-to-Earth data integrity assurance. The project objective was to create a tamper-proof data transmission mechanism utilizing a four-layer architecture. The system integrates custom SHA-256 blockchain implementation with Proof-of-Work consensus, machine learning-based anomaly detection, and comprehensive data persistence mechanisms.

The implementation encompasses core blockchain functionality through custom Block and Blockchain classes, eliminating dependency on external blockchain libraries. The system features a complete four-layer architecture comprising Sensor Layer for telemetry generation, Relay Layer for block validation and mining, Ground Layer for data calibration and processing, and Application Layer for final verification and audit preparation. The tamper detection module provides real-time integrity monitoring with JSON-based logging of security events. The machine learning component implements RandomForest-based anomaly detection with statistical feature extraction from satellite telemetry data.

The system demonstrates successful integration of multiple technologies including Python-based backend processing, JSON-based persistence, NiceGUI dashboard interface, and React-based landing page . System validation through unit tests, integration tests, and execution-based analysis confirmed successful achievement of core functional requirements. Development assumptions included constant network latency, reliable power availability, and continuous storage access which may not reflect actual operational conditions in space missions.

The system demonstrates high reliability based on zero runtime errors during testing cycles and consistent behavior across multiple execution scenarios. The modular architecture supports independent component testing and maintenance. The comprehensive error handling mechanisms and input validation contribute to system stability. However, the single-node implementation and lack of distributed consensus mechanisms limit applicability to mission-critical scenarios requiring fault tolerance.

VI.REFERENCES

1.CCSDS Blue Books portal (TM/AOS/USLP and security standards). CCSDS

2.CCSDS TM Space Data Link Protocol (specification PDF). CCSDS

3.CCSDS AOS Space Data Link Protocol (specification PDF). CCSDS

4.CCSDS Unified Space Data Link Protocol (USLP), latest revision. CCSDS

5.CCSDS Licklider Transmission Protocol (LTP) (specification PDF). CCSDS

6.CCSDS Space Data Link Security (SDLS) (specification PDF). CCSDS

7.Fischer et al., “Finalizing the CCSDS Space-Data Link Layer Security Protocol: Interoperability Testing.” NASA NTRS. NASA Technical Reports Server

8.IETF RFC 9171: Bundle Protocol Version 7. IETF Datatracker

9.IETF RFC 9172: Bundle Protocol Security (BPSec). RFC Editor

10.NASA SCaN pages on DTN overview and mission integration. NASA+1

11.ESA EDRS/SpaceDataHighway overview. European Space Agency

12.Wired feature on early EDRS high-throughput relay demonstration. WIRED

13.IETF RFC 6962: Certificate Transparency (tamper-evident logs). RFC Editor

14.Let’s Encrypt note on CT Merkle-tree properties and append-only verification. letsencrypt.org

15.Ahmad et al., “Blockchain for aerospace and defense: opportunities and open research challenges,” Computers & Industrial Engineering, 2021. ScienceDirect

16.Mital et al., “Blockchain application within a multi-sensor satellite architecture,” NASA/IGARSS paper (NTRS PDF). NASA Technical Reports Server

17.Torky et al., “A Blockchain Protocol for Authenticating Space Communications in Satellite Networks,” Aerospace, 2022. MDPI

18.Kim et al., “Secure and Transparent Space Exploration Data Management via Hybrid Blockchain,” Applied Sciences, 2025. MDPI

19.WIPO Tech Trends note: blockchain in satellite communications (use cases and citations). WIPO+1

20.NIST FIPS 180-4 and SP 800-107r1 (hash algorithm standard and usage guidance). NIST Publications+1

21.Consensus background: ACM and MDPI surveys; Hyperledger modular consensus docs.