Security Architecture for Decentralized Mesh Networks in Critical Sensor Infrastructure

Threat modeling, authenticated joining and resilient management

Abstract

Mesh communication can increase connectivity while enlarging the number of devices entrusted to process and forward data. This paper organizes the security controls described in MESH2026 into a lifecycle architecture for identity, device joining, secure routing, encryption, replay defense, key recovery and software updates. The analysis distinguishes external attackers from compromised authorized devices and emphasizes operation under intermittent connectivity. A threat matrix connects common attack effects to preventive and diagnostic measures. This is an architectural and standards informed assessment, not a penetration test or certification statement. The proposed test protocol targets failures that arise when otherwise correct cryptographic components interact with routing, storage and field maintenance.

Keywords: wireless mesh networks; sensor networks; security of decentralized mesh; analytical modeling; reliability; network design.

1 Introduction

Wireless mesh designs connect nodes using multiple wireless links and forwarding relays, but the existence of an alternate link does not by itself guarantee end to end service. Performance depends on radio quality, control traffic, source reporting rates, topology changes and gateway capacity. This study addresses one distinct subset of those coupled design concerns, using the models and engineering constraints developed in MESH2026-v07. Its findings are analytical statements with explicit assumptions rather than claims about a completed controlled deployment.

The objective is to define a technically traceable decision framework that converts the supplied study into testable engineering criteria. The manuscript draws solely on the source report and bibliographic entries present in that report. References retain the original source numbering so each item can be reconciled directly with the underlying bibliography. No new externally obtained sources are introduced.

2 Analytical Scope and Method

The method proceeds by identifying the system boundary, selecting a small number of measurable state variables, deriving a simplified relationship or comparison, and then distinguishing mathematical consequences from assumptions that require experimental verification. Conceptual drawings show component or information relationships; calculated charts visualize nominal conditions. They are not field measurements. This separation is especially important in wireless systems, where terrain, interference, retransmission policy and time synchronization can dominate apparent theoretical advantage.

Source basis: MESH2026-v07, sections 3.6, 3.7.3, 3.8. Reference numbers preserve the numbering of its chapter 7 bibliography. The analytical figures are original illustrations of stated assumptions, not experimental observations.

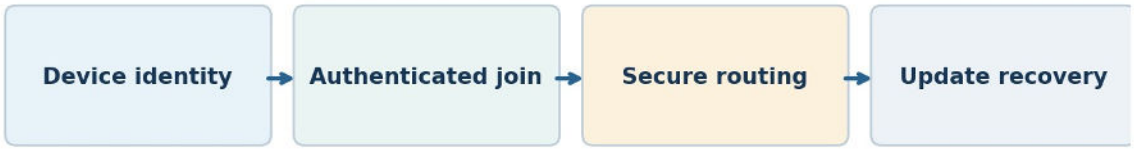


Figure 1. Functional architecture developed from the relevant MESH2026 concepts; schematic, not a tested implementation.

3 Threat surface of an autonomous mesh

Every routing node stores operational state, may forward another device's sensitive readings, and can become an attractive point of manipulation. A passive external observer threatens confidentiality. An active radio attacker may inject traffic or jam shared medium access. A compromised legitimate node can present an even harder problem: it may possess working keys and distribute believable but incorrect routing or sensor data. The attack analysis in the source bibliography includes sinkhole, wormhole and selective forwarding behaviors.

A useful threat model separates the attacker's radio proximity, possession of credentials, physical access to equipment and capability to compromise firmware. It also classifies the protected assets: telemetry values, precise location, equipment status, command authenticity, node identity and route availability. A cryptographic tunnel protects some assets but does not establish the truth of a sensor reading produced by a captured node. Security evaluation must therefore examine both protocol properties and the trust boundary around each hardware module. [31], [90].

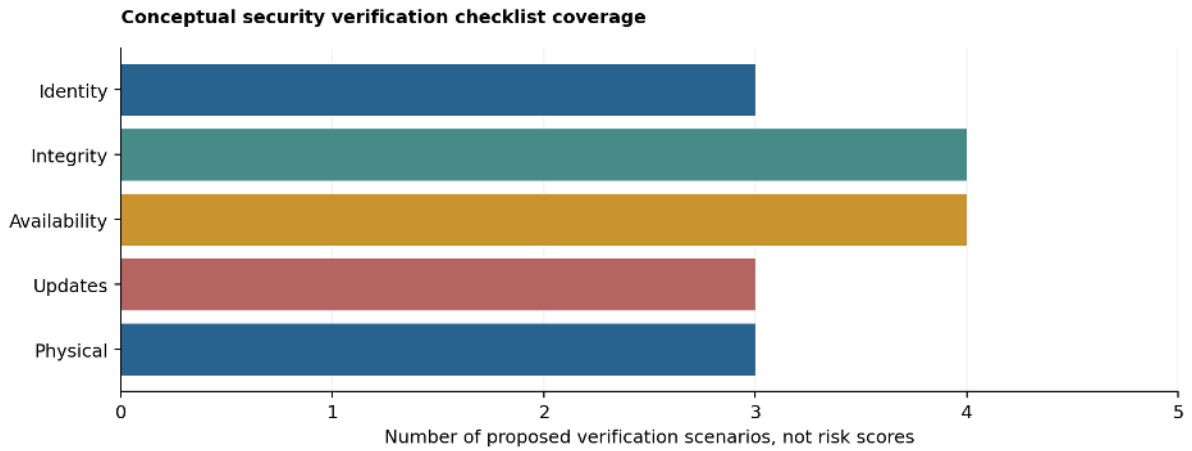


Figure 2. Calculated or qualitative illustration of the design mechanism. Axes and assumptions are described in the main text.

4 Secure onboarding and identity continuity

The joining process should uniquely identify the device and authorize it for a defined network and role before enabling application privileges. A modular sensor package raises additional concerns because an attached GPS or microphone may expose new data or commands. Module presence should not automatically grant access to stored keys or network-wide routing privileges. Changes to role and membership must be auditable and reversible.

In disconnected environments onboarding may have to work without a permanently available central server. This motivates carefully constrained preprovisioned credentials or local authorization rules, but it also raises revocation latency. Network operators must know what happens when a device is lost, a battery is replaced or a node is reinstalled after a long offline interval. The threat model should specify when cached authorization expires and which actions are permitted without fresh trust information. [91], [98].

Table 1. Scope of the principal design comparison.

Threat	Potential effect	Verification control
Spoofed identity	Unauthorized join or command	Authentication and role checks
Replay	Repeated command / false history	Persistent nonce or counter handling
Sinkhole / selective drop	Data loss through malicious relay	Path diagnostics and anomaly reporting
Stolen node	Disclosure of credentials	Key revocation and physical hardening
Interrupted update	Unavailable or unsafe firmware	Signed update and recovery test

5 Message integrity, replay and routes

Confidentiality is achieved only by an appropriate cipher and key arrangement; integrity and origin authentication have separate failure modes. Nonces, sequence numbers and anti-replay windows require persistent or safely recoverable state. A brownout that resets a counter can invalidate otherwise sound cryptography. Routing announcements and route updates deserve equivalent attention because tampering can divert packets, suppress service or increase battery expenditure.

Link level protection and application end to end protection do not have the same trust boundary. Forwarding relays may need to inspect routing headers but should not necessarily read sensor payloads. The design must decide which fields require hop protection and which remain authenticated end to end. Security overhead also affects packet size, airtime, stored state and key management energy. [72], [92], [93].

6 Key rotation and resilient updates

Cryptographic materials must be provisioned, renewed and retired through a defined lifecycle. A planned rotation should avoid partitioning a duty cycled mesh when some nodes miss their downlink window. Dual key transition periods can improve availability but expand the period in which more credentials remain valid. Lost nodes and compromised keys require a separate response, since rejoining with a known secret is not adequate recovery.

Firmware updates should verify origin and integrity, tolerate interrupted transfer, and provide a rollback or known safe startup path. Repeated downloads over low capacity links can overwhelm channel and battery budgets. A phased rollout through gateways and representative relays can reveal regressions before a network-wide loss of communication. Protection mechanisms are therefore part of operational availability, not a cosmetic security layer. [94], [95], [96], [97].

Evaluation parameters

A reproducible trial should log per-node packet identifiers, generation and reception timestamps, route changes, radio retransmissions, queue age and battery current. All measurement and reporting intervals, payload formats, antenna geometry and firmware revisions should be recorded. Reliability must be reported with variation over trials, not as a single best-run percentage. A successful link in one season or installation position is not equivalent to a verified network-level service guarantee.

7 Verification and limits

The acceptance plan should simulate replayed messages, invalid join attempts, stale key versions, packet modification, partial firmware updates, loss of time synchronization and compromise of a relay role. Log detection quality, legitimate delivery impact, control plane overhead and ability to recover after power loss. Each test must identify the exact protocol stack because RPL, BLE Mesh and LoRa MESH do not share one universal security mechanism.

This paper does not claim conformance with NATO, NIST or any other regime solely by listing relevant specifications. Such conformity demands requirements mapping and test evidence for a specific hardware/software revision. The analytic matrix is a prerequisite for a defensible security qualification program. [84].

8 Reproducible Experiment Design

Adversarial lifecycle verification

Create a controlled test network with temporary credentials and simulated compromised roles. Attempt unauthorized joining, replay a previously accepted control packet, introduce a route announcement from a non-authorized node, and interrupt a signed update. Each test should have an unambiguous expected authorization and recovery outcome. Do not use production credentials or live operational networks for adversarial qualification.

All trials should state the hypothesis before measurements begin. In particular, baseline and modified networks must generate a comparable number of application messages under identical payload semantics. Time synchronization, radio configuration, antenna placement and test duration should be documented. Apply a predefined definition of a valid delivery, including how duplicate frames, acknowledgements, partial packets and expired records are treated. Report distributions and uncertainty rather than interpreting isolated examples as population-level network behavior.

The primary measurement vector is: Rejection rate, unauthorized command execution, integrity alarms, recovery state, availability and key lifecycle audit. The experiment should archive raw records and configuration files so that independent reviewers can reconstruct packet success, route depth and energy exposure. Failed configurations and unavailable nodes must remain in the results table rather than being discarded as outliers. Where an event is delayed by buffering, store both the origin timestamp and the gateway arrival timestamp to avoid falsely classifying a stale packet as timely delivery.

9 Model Sensitivity and Acceptance Criteria

A design parameter has operational meaning only relative to an application threshold. For packet delivery, an acceptable loss ratio may differ between a periodic weather reading and an emergency alarm. For latency, a median value can be misleading when occasional extreme delay is harmful. Therefore the acceptance plan should name the threshold and its aggregation level: individual node, gateway sector or whole deployment. At minimum, retain a reliability target, a maximum age for useful data, and a minimum expected autonomy for each energy role.

Sensitivity analysis should vary one major assumption at a time and then test coupled effects. Candidate variables include per-hop reception probability, reporting frequency, number of downstream children, active listening duration, gateway count and available battery capacity. An improvement observed under one parameter combination may disappear when the network is

congested or when temperature reduces battery availability. The purpose of a sensitivity plot is to identify which physical measurement would most change the final design decision, not to predict field performance without evidence.

Successful encryption alone is insufficient; compromised authorized relays and partial power failures may threaten integrity or availability.

A final acceptance matrix should include four outcomes: the test condition, the expected service limit, the observed numerical result and the engineering decision. The source technical study supports the selection of variables and design mechanisms, but a numerical pass/fail result requires an actual implemented protocol and controlled measurements. The distinction prevents a model assumption from being silently promoted to verified specification.

Evaluation scenario matrix

Scenario	Variable held constant	Question to resolve
Nominal baseline	Same radio / payload / gateway position	Does the reference service meet its target?
Increased offered traffic	Same node count and physical layout	Does age or packet loss rise before a stated limit?
Relay or link interruption	Same sensor sample generation	Can useful records still reach the gateway?
Reduced energy reserve	Same routing and application service	Which role becomes the limiting component?
Configuration change	Same declared application goal	Does adaptation improve the system-level outcome?

The nominal baseline provides a necessary denominator for every claimed improvement. A second, more demanding workload should keep the number and location of devices unchanged, which helps isolate the effect of the altered report rate or route policy. The failure trial should define in advance whether losing the first packet after an outage counts as unacceptable service or whether a bounded recovery period is allowed. A device can legitimately buffer packets for later transfer, but the resulting data age must be visible in the assessment.

Energy experiments should compare the same battery assumptions and device roles. A measured reduction in the charge consumed by a leaf node may be offset by increased listening or forwarding costs in its parent. The appropriate system measure is therefore not simply the largest lifetime estimate in the network; it is the minimum acceptable service continuity across critical coverage points. Temperature, battery cutoff and intermittent backhaul availability may change which node becomes critical even when network geometry is unchanged.

Finally, sensitivity should be reported as an envelope rather than an unexplained single-number promise. State which parameters were deliberately changed, identify the origin of every numerical assumption, and preserve the complete measured distribution if physical tests are later performed.

When the source provides only theoretical values, label the result a calculation or hypothesis. This audit trail makes the framework useful for deciding the next experiment and reduces the risk that an optimistic reference scenario will be mistaken for a certified operating limit.

10 Discussion

The analysis points to a recurring challenge: radio reach, packet reliability, bounded latency and energy endurance are not independently optimizable in a shared forwarding network. Improving one dimension can worsen another. Greater redundancy adds possible routes but can increase scanning or contention. More frequent reporting improves data freshness but increases relay burden. Stronger integrity protection changes frame overhead and configuration responsibilities. These effects require a scenario specific performance contract before hardware or protocol claims can be generalized.

The most defensible next step is a measured comparison with a declared baseline and pre-registered acceptance thresholds. Test design should preserve the conditions under which the mathematical assumptions are approximately valid, document deviations, and present limits as prominently as advantages. This article supplies an analytical framework suitable for subsequent empirical investigation but does not claim experimental superiority or compliance certification.

11 Conclusion

The MESH2026 material supports a structured analytical treatment of security architecture for decentralized mesh networks in critical sensor infrastructure. The resulting framework separates topology, traffic, radio and energy constraints, supplies scenario dependent decision criteria, and defines a path to experimental verification. The final engineering choice should be guided by the application’s maximum data age, required delivery probability, actual relay workload and maintainability under failure. Numerical examples are conditional calculations, and validation against physical deployments remains necessary.

References

[31] W. K. Tan et al., “A Security Analysis of the 802.11s Wireless Mesh Network Routing Protocol,” Sensors, 13(9), 11553–11585, 2013. DOI 10.3390/s130911553.

[72] A. Lacava et al., “Securing Bluetooth Low Energy networking,” Computer Networks, 211, 108953, 2022. DOI 10.1016/j.comnet.2022.108953.

[84] E. Birrane and K. McKeever, Bundle Protocol Security, RFC 9172, 2022. DOI 10.17487/RFC9172.

[90] T. Tsao et al., A Security Threat Analysis for RPL, RFC 7416, 2015. DOI 10.17487/RFC7416.

[91] M. Vučinić et al., Constrained Join Protocol for 6TiSCH, RFC 9031, 2021. DOI 10.17487/RFC9031.

[92] G. Selander et al., Object Security for Constrained RESTful Environments, RFC 8613, 2019. DOI 10.17487/RFC8613.

[93] E. Rescorla et al., Datagram Transport Layer Security Protocol Version 1.3, RFC 9147, 2022. DOI 10.17487/RFC9147.

[94] E. Barker, Recommendation for Key Management Part 1, NIST SP 800-57 Rev. 5, 2020. DOI 10.6028/NIST.SP.800-57pt1r5.

[95] B. Moran et al., A Firmware Update Architecture for Internet of Things, RFC 9019, 2021. DOI 10.17487/RFC9019.

[96] B. Moran et al., A Manifest Information Model for Firmware Updates in IoT, RFC 9124, 2022. DOI 10.17487/RFC9124.

[97] A. Regenscheid, Platform Firmware Resiliency Guidelines, NIST SP 800-193, 2018. DOI 10.6028/NIST.SP.800-193.

[98] M. Fagan et al., IoT Device Cybersecurity Capability Core Baseline, NISTIR 8259A, 2020. DOI 10.6028/NIST.IR.8259A.

[M] MESH2026-v07. Technical study, sections 3.6, 3.7.3, 3.8. Unpublished technical material.