Trust-Aware Backpressure RPL for Resilient Routing in LLNs under Routing Attacks

이건형

Abstract

Low-Power and Lossy Networks (LLNs) rely on RPL and its backpressure-based variant BRPL to achieve stable routing under lossy links and congestion. However, these protocols do not explicitly account for insider routing attacks that manipulate data-plane forwarding or control-plane advertisements, which can lead to abrupt performance collapse beyond a topology-dependent attack intensity. This paper presents *Trust-Aware BRPL*, a lightweight integration of trust-derived penalties into BRPL's routing decision. We model data-plane reliability via a Beta-based estimator with EWMA smoothing and capture control-plane anomalies using rank-inconsistency and stability signals. Using Contiki-NG/Cooja, we conduct reproducible parameter sweeps over attack intensity, topology classes, and network scale, and compare BRPL vs. Trust-Aware BRPL in terms of packet delivery ratio (PDR), end-to-end delay, control overhead, and parent churn. The results (to be inserted) are organized to quantify how trust parameters shift the collapse threshold and to characterize the resulting resilience–overhead trade-off.

Keywords: RPL, BRPL, Backpressure, Trust, LLN, IoT Security, Selective Forwarding, Sinkhole, Contiki-NG, Cooja

1 Introduction

LLNs are widely used in IoT sensing and control where devices operate under strict power and memory constraints and communicate over lossy wireless links. RPL provides a standardized IPv6 routing framework for LLNs based on DODAG formation, rank, and parent selection [4]. BRPL extends RPL with backpressure-inspired decisions to improve throughput and latency in the presence of congestion by exploiting path diversity [8].

Despite these advances, routing in LLNs remains vulnerable to insider attacks. Two representative threats are (i) *selective forwarding* (grayhole/blackhole), where a compromised forwarder probabilistically drops packets, and (ii) *sinkhole* behavior, where a node attracts traffic through misleading control-plane advertisements (e.g., rank manipulation) and subsequently disrupts delivery [5, 13, 6, 7]. Empirically, such attacks often induce a *phase-transition-like* behavior: performance degradation is small for low attack intensity but collapses sharply beyond a threshold that depends on topology and routing structure.

1.1 Goal and Approach

This work aims to delay or mitigate this collapse by incorporating lightweight trust signals into BRPL's decision process. Instead of introducing heavy cryptographic mechanisms or standalone intrusion detection, we integrate trust as a *penalty* in the routing metric: suspected nodes become less attractive as forwarders/parents while preserving BRPL's congestion-aware benefits.

1.2 Contributions

- **Trust-Aware BRPL design:** a lightweight trust-penalized BRPL metric that down-weights untrustworthy neighbors while retaining backpressure-based adaptivity.

- **Dual-plane trust modeling:** data-plane trust via Beta estimation with EWMA smoothing and control-plane trust via rank-inconsistency and stability signals derived from RPL semantics.
- **Reproducible evaluation framework:** Contiki-NG/Cooja implementation with scripted sweeps across topology classes, scales (S/M/L), and attack intensities for selective forwarding, sinkhole, and combined attacks.
- **Collapse-threshold analysis:** result organization that quantifies how trust parameters shift the collapse point and how resilience trades off against overhead and churn.

1.3 Paper Organization

Section 2 reviews background and related work. Section 3 defines the system and threat model. Section 4 presents Trust-Aware BRPL. Section 5 describes implementation. Section 6 details experimental setup. Section 7 and Section 8 report and discuss results. Section 9 concludes.

2 Background and Related Work

2.1 RPL and BRPL

RPL constructs a DODAG rooted at a sink and uses rank to enforce loop avoidance and convergence [4]. BRPL augments RPL with backpressure principles, combining queue differentials with routing costs to improve throughput and delay under congestion, leveraging multiple paths when available [8].

2.2 Routing Attacks in LLNs

Selective forwarding and sinkhole/rank attacks are well-studied threats that can severely degrade delivery and stability in LLNs [5, 13, 6, 7]. Sinkhole behavior often manifests through misleading control-plane information that attracts traffic; once positioned on critical paths, an attacker can amplify disruption.

2.3 Trust-Based Routing

Trust-based routing introduces behavioral reliability into routing decisions, commonly via direct observation and/or reputation systems [2, 3, 9]. In LLNs, imperfect wireless observations and limited overhearing motivate lightweight estimators and smoothing rather than full watchdog schemes.

3 System and Threat Model

3.1 System Model

We use Contiki-NG and Cooja to simulate 6LoWPAN/IPv6 LLN stacks with RPL/BRPL routing. Traffic follows a many-to-one periodic pattern from sensor nodes to a root. Nodes maintain neighbor tables and select preferred parents according to the underlying routing logic [4, 8].

3.2 Threat Model

We consider a single insider attacker (extensions to multiple attackers are future work). The attacker participates as a normal node but executes one of the following:

- **Selective forwarding:** probabilistic packet dropping on forwarded traffic, with drop probability swept as attack intensity.
- **Sinkhole (rank manipulation):** control-plane manipulation to attract traffic (e.g., advertising an artificially low rank), followed by normal forwarding unless combined with dropping.
- **Combined attack:** sinkhole attraction plus selective forwarding on captured flows.

We assume the root is non-malicious. Cryptographic authentication and secure bootstrapping are outside the scope of this paper.

4 Trust-Aware BRPL Design

This section defines trust signals and how they modulate BRPL decisions. We denote by i the observing node and j a neighbor candidate (potential parent/forwarder). Trust values are normalized to $[0, 1]$ where larger is better.

4.1 Data-Plane Trust for Selective Forwarding

We model forwarding reliability as the probability that a neighbor forwards packets successfully. Let s_j and f_j denote observed successful and failed forwarding events attributed to j (e.g., via log-based inference). Using a Beta prior (α_0, β_0) for Bernoulli forwarding outcomes [2, 3], the posterior mean is

$$\hat{T}_{\text{gray}}(j) = \frac{\alpha_0 + s_j}{\alpha_0 + \beta_0 + s_j + f_j}. \tag{1}$$

To reduce short-term noise, we apply EWMA smoothing:

$$T_{\text{gray}}(j; t) = \rho\, T_{\text{gray}}(j; t-1) + (1 - \rho)\, \hat{T}_{\text{gray}}(j), \tag{2}$$

where $\rho \in [0, 1)$ controls smoothing.

4.2 Control-Plane Trust for Sinkhole Behavior

Sinkhole attacks primarily affect control-plane advertisements. RPL rank semantics impose monotonicity constraints on feasible parent relations [4]. Let R_i be node i's current rank and R_j the advertised rank from neighbor j. Define

$$\Delta_{ij} = R_j + \text{MIN\_HOPRANKINC} - R_i. \tag{3}$$

If j advertises an implausibly low rank relative to i, Δ_{ij} becomes negative. We define a deviation score with tolerance $\tau \geq 0$:

$$s_{ij} = \max(0,\, -\Delta_{ij} - \tau), \tag{4}$$

and map it to an exponential trust decay [12, 1]:

$$T_{\text{adv}}(j) = \exp(-\lambda_{\text{adv}}\, s_{ij}), \tag{5}$$

where $\lambda_{\text{adv}} > 0$ sets sensitivity.

Sinkhole behavior can also induce instability in rank evolution and parent selection. Over a window W, define rank increase

$$\Delta R_i = R_i(t) - R_i(t - W), \tag{6}$$

and penalize abnormal increases beyond $\kappa \geq 0$:

$$u_i = \max(0,\, \Delta R_i - \kappa), \qquad T_{\text{stab}}(t) = \exp(-\lambda_{\text{stab}}\, u_i). \tag{7}$$

We combine control-plane signals multiplicatively:

$$T_{\text{sink}}(j) = (T_{\text{adv}}(j))^{w_1}\, (T_{\text{stab}}(t))^{w_2}\,, \tag{8}$$

with weights $w_1, w_2 \geq 0$.

4.3 Total Trust Aggregation

We aggregate data-plane and control-plane trust via a weighted geometric mean:

$$T_{\text{total}}(j) = (T_{\text{gray}}(j))^{\alpha} \, (T_{\text{sink}}(j))^{1-\alpha}, \tag{9}$$

where $\alpha \in [0, 1]$ controls the emphasis on selective-forwarding vs. sinkhole signals.

4.4 Trust-Penalized BRPL Metric

Let BP_{ij} denote the baseline BRPL weight/utility for neighbor j from node i (as defined by BRPL's backpressure and cost combination) [8]. We define a trust penalty factor

$$\phi(T) = \frac{T^{\gamma}}{1 + \lambda \, (1 - T)^{\gamma}}, \tag{10}$$

where $\lambda \geq 0$ controls avoidance aggressiveness and $\gamma \geq 1$ controls risk sensitivity. The trust-aware metric is

$$BP_{ij}^{(\text{trust})} = BP_{ij} \cdot \phi(T_{\text{total}}(j)) \, . \tag{11}$$

This form preserves ordering when trust is uniform, while strongly down-weighting neighbors with low trust as λ or γ increases. The use of tunable penalty parameters follows common drift-plus-penalty / risk-sensitive control intuition in stochastic network optimization [10].

4.5 Algorithm Outline

At each decision epoch, node i:

1. Updates trust values (data-plane and/or control-plane) and clamps to $[0, 1]$.
2. For each neighbor candidate j, computes $BP_{ij}^{(\text{trust})}$.
3. Selects the preferred parent / forwarding next hop that maximizes the trust-penalized objective, with optional hysteresis to reduce churn.

4.6 Overhead

Trust maintenance stores $O(\text{deg})$ values per node, where deg is neighbor degree. The metric adds $O(\text{deg})$ arithmetic per decision. No additional control packets are required by design; any extra overhead arises from changed routing dynamics (e.g., parent switching).

5 Implementation in Contiki-NG

We implement Trust-Aware BRPL in Contiki-NG by integrating the trust penalty into the BRPL metric computation and parent selection path. The attacker node supports: (i) probabilistic forwarding drops for selective forwarding, (ii) rank advertisement manipulation for sinkhole behavior, and (iii) combined operation.

All trust values used for routing decisions are computed and applied *online* during the simulation runtime, while the trust engine additionally performs *offline* log-based analysis solely for measurement, visualization, and reproducibility purposes; offline results are never fed back into routing decisions.

To support measurement and reproducibility, we maintain a scripted pipeline that runs batch experiments over seeds, topologies, and attack parameters, and extracts metrics from Cooja logs. Implementation-specific file names and scripts are omitted from the main narrative and can be provided as an artifact alongside the code repository.

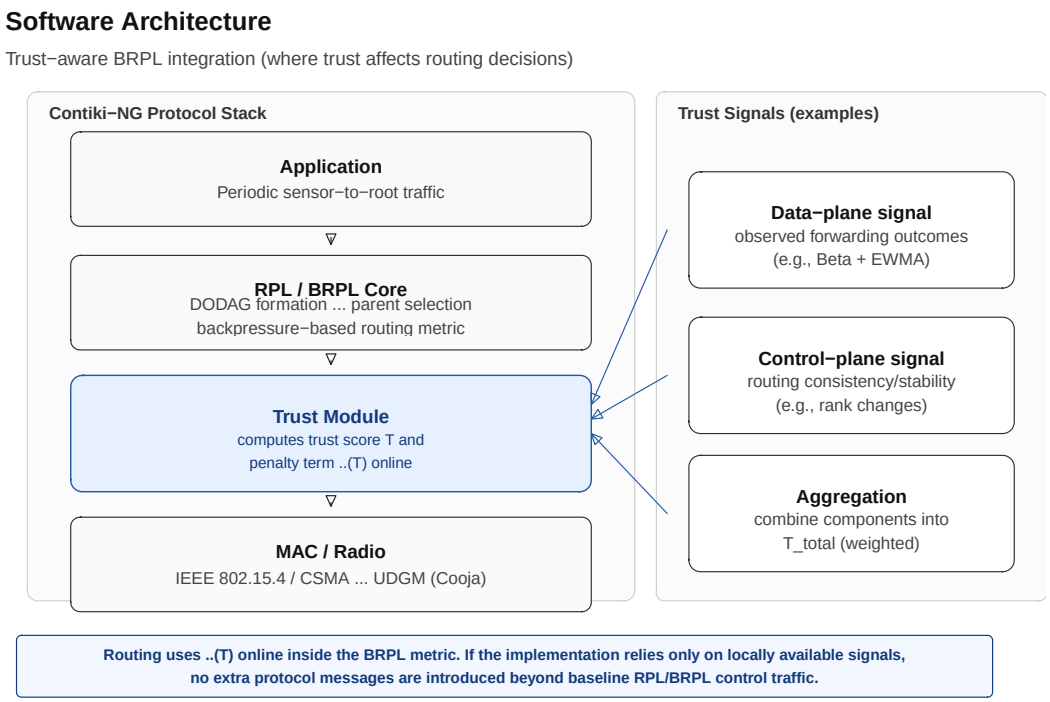

Figure 1: Software architecture of Trust-Aware BRPL and its integration into the Contiki-NG protocol stack.

6 Experimental Setup

6.1 Simulation Environment

We use the UDGM Distance Loss radio model in Cooja [11]. Unless otherwise stated, simulations run for 600 s with a 120 s warm-up. Sensor nodes generate periodic traffic to the root every 30 s. RPL Trickle parameters follow a standard configuration derived from RPL's design trade-offs between stability and control overhead [4].

6.2 Topologies and Scales

We evaluate multiple topology *classes* to cover different path diversity regimes:

- **Grid:** high path diversity; attacks may be partially bypassed.
- **Two-cluster with bottleneck:** a constrained cut; attacks are amplified when the attacker sits near the bottleneck.
- **Corridor/chain:** low path diversity; collapse can occur at lower intensities.
- **Ring/spokes:** intermediate diversity; highlights root-near and mid-path influence.

We scale network size as S/M/L (e.g., 16/36/64 nodes including the root) while preserving the same placement rule per class. Attacker placement follows a rule-based selection (e.g., central relay candidate or bottleneck vicinity) to avoid ad-hoc tuning.

6.3 Attack Configuration

Selective forwarding is parameterized by drop probability (e.g., 0/30/50/70%). Sinkhole intensity is parameterized by a rank manipulation delta (e.g., 1/2/4). We evaluate three modes: selective forwarding only, sinkhole only, and combined.

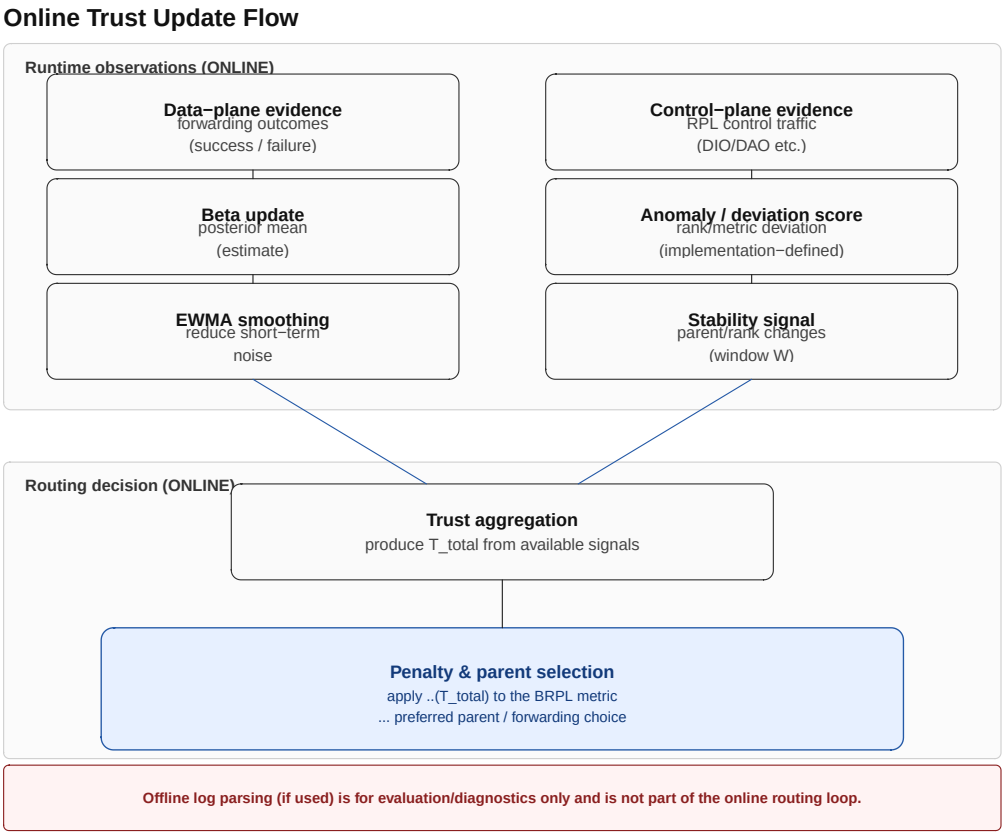

Figure 2: Online trust update and routing decision flow executed at each node during simulation runtime.

6.4 Trust Model Parameters

Table 1 summarizes the trust-related parameters used throughout the experiments. Unless otherwise stated, the same configuration is applied across all topologies and scales to avoid per-scenario tuning.

6.5 Metrics

We report:

- **PDR:** $\mathrm{RX}_{\mathrm{root}}/\mathrm{TX}_{\mathrm{senders}}$.
- **End-to-end delay:** per-packet latency from source to root.
- **Control overhead:** control-plane transmissions (e.g., DIO/DAO) and routing-related overhead.
- **Parent churn:** parent switching rate, reflecting routing stability.
- **Exposure (optional):** fraction of delivered packets traversing the attacker and/or fraction of time the attacker is a preferred parent (used to interpret attack effectiveness).

6.6 Parameter Sweeps

We sweep: (i) attack intensity, (ii) topology class and scale, (iii) trust aggregation weight α, (iv) penalty parameters (λ, γ) for trust-aware cases. Multiple random seeds are used to report mean and variability.

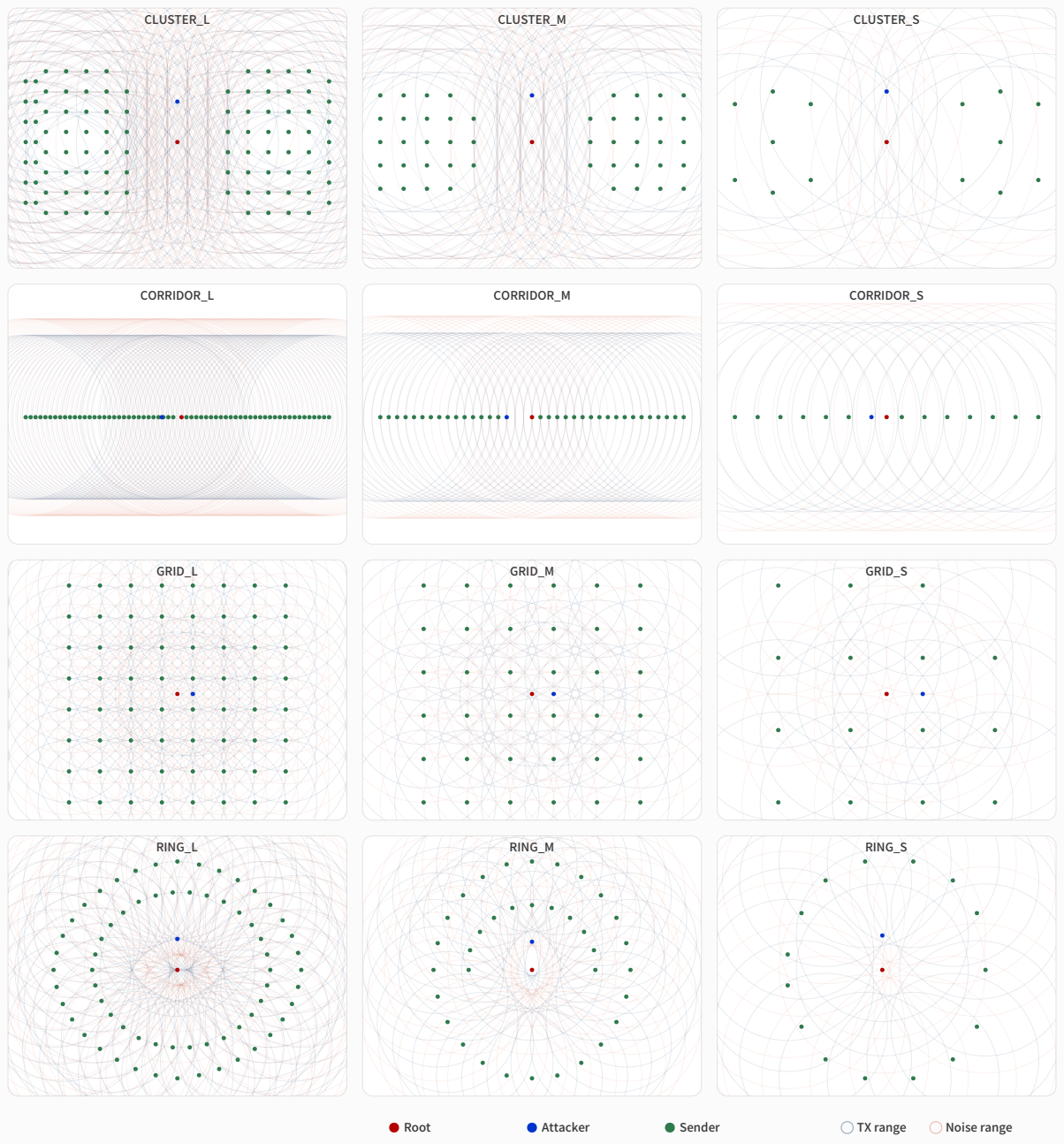

Figure 3: Topology layouts used for evaluation, covering different path diversity regimes and attacker positions.

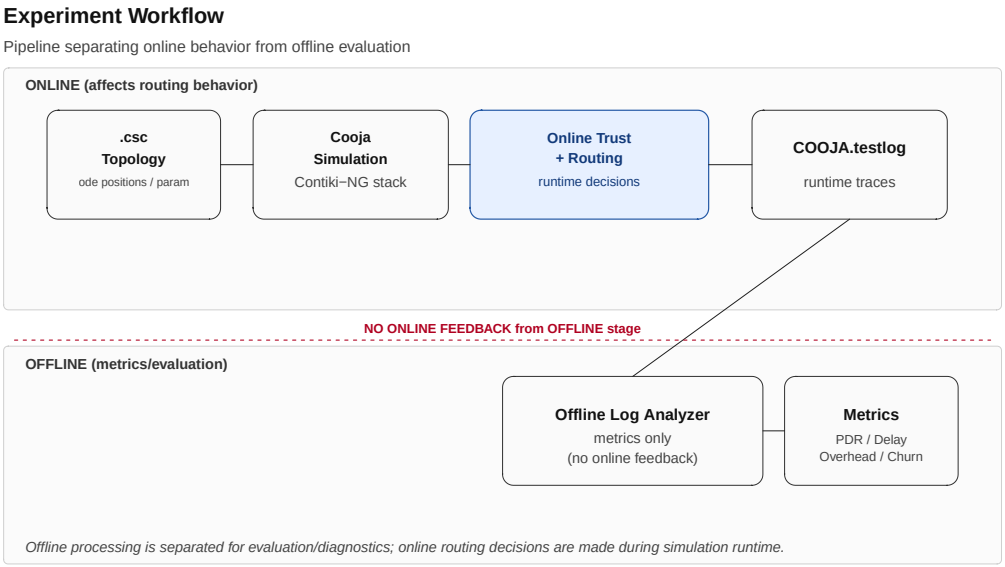

Figure 4: Experiment workflow from simulation execution to offline metric extraction.

Fig. 3 (placeholder): PDR vs. attack intensity by topology/scale

Figure 5: PDR vs. attack intensity (placeholder).

7 Results

This section will be finalized after data collection. We structure the analysis to highlight (i) collapse threshold behavior and (ii) trade-offs induced by trust penalization.

8 Discussion

We discuss how topology-dependent path diversity and attacker placement influence collapse thresholds, and how trust penalization shifts these thresholds. We also analyze (i) potential false positives under wireless loss and transient dynamics, (ii) sensitivity to $(\lambda, \gamma, \alpha)$, and (iii) stability impacts reflected in churn and overhead. Importantly, the proposed mechanism is intended to complement—not replace—cryptographic protections and secure bootstrapping, providing resilience gains even when full authentication is unavailable or impractical.

9 Conclusion and Future Work

We presented Trust-Aware BRPL, which integrates lightweight trust penalties into BRPL routing decisions using both data-plane reliability and control-plane anomaly signals. The evaluation plan is designed to quantify how trust parameters delay performance collapse under selective forwarding and sinkhole attacks and to characterize the resilience–overhead trade-off across topologies and scales. Future work includes multiple colluding attackers, adaptive tuning of trust penalty parameters, and validation on real testbeds.

References

[1] D. Chen and P. K. Varshney. Trust-based routing for wireless sensor networks. In *Proceedings of IEEE MASS*, 2010.

Table 1: Trust model parameters used in experiments.

Parameter	Description	Value
α_0	Beta prior (success)	1
β_0	Beta prior (failure)	1
ρ	EWMA smoothing factor	0.8
W	Rank stability window	5 sampling intervals
τ	Rank deviation tolerance	0
κ	Rank increase tolerance	0
λ_{adv}	Rank anomaly sensitivity	0.01
λ_{stab}	Rank instability sensitivity	0.01
w_1	Weight of rank inconsistency trust	0.5
w_2	Weight of rank stability trust	0.5
α	Grayhole vs. sinkhole trust weight	$\{1.0, 0.5\}$
λ	Trust penalty strength	$\{0, 1, 3, 10\}$
γ	Risk sensitivity exponent	$\{1, 2, 4\}$

Table 2: Core experimental parameters (default).

Parameter	Value
Field size	200 m × 200 m
Radio model	UDGM Distance Loss
TX range / Interference range	45 m / 90 m
Simulation time / Warm-up	600 s / 120 s
Traffic interval	30 s
Root ID	1
Attacker count	1 (default)

[2] Saurabh Ganeriwal and Mani Srivastava. Reputation-based framework for high integrity sensor networks. In *Proceedings of SecureComm*, 2004.

[3] Saurabh Ganeriwal, Laura Balzano, and Mani B. Srivastava. Reputation-based framework for high integrity sensor networks. *ACM Transactions on Sensor Networks*, 4(3), 2008.

[4] IETF. Rpl: Ipv6 routing protocol for low-power and lossy networks. RFC 6550, 2012. Internet Engineering Task Force.

[5] Chris Karlof and David Wagner. Secure routing in wireless sensor networks: Attacks and countermeasures. *Ad Hoc Networks*, 1(2–3):293–315, 2003.

[6] Anh Le et al. The impact of rank attack on RPL-based networks. In *Proceedings of IEEE ICC*, 2013.

[7] A. Mayzaud, R. Badonnel, and I. Chrisment. A taxonomy of attacks in RPL-based internet of things. *IEEE Communications Surveys & Tutorials*, 18(2):169–184, 2016.

[8] Scott Moeller et al. Brpl: Backpressure RPL for high-throughput and low-latency in LLNs. In *Proceedings of IEEE INFOCOM*, 2016.

[9] M. Momani and S. Challa. Survey of trust models in different network domains. *IEEE Communications Surveys & Tutorials*, 12(2):1–21, 2010.

[10] Michael J. Neely. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan & Claypool, 2010.

Figure 6: End-to-end delay vs. attack intensity (placeholder).

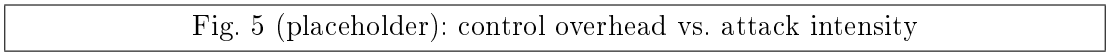

Figure 7: Control overhead vs. attack intensity (placeholder).

[11] Fredrik Osterlind, Adam Dunkels, Joakim Eriksson, Niclas Finne, and Thiemo Voigt. Cross-level sensor network simulation with COOJA. In *Proceedings of ACM SenSys*, 2006.

[12] Yan Sun, Wei Yu, Zhu Han, and K. J. Ray Liu. Trust modeling and evaluation in ad hoc networks. In *Proceedings of IEEE GLOBECOM*, 2005.

[13] Anthony D. Wood and John A. Stankovic. Denial of service in sensor networks. *IEEE Computer*, 35(10):54–62, 2002.

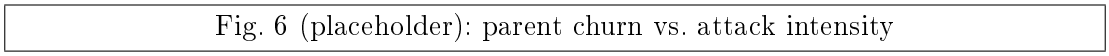

Figure 8: Parent churn vs. attack intensity (placeholder).

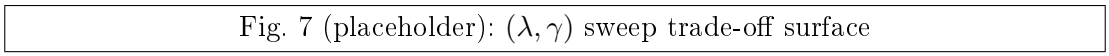

Figure 9: Trade-off under trust penalty sweeps (placeholder).