

IoT 네트워크에서 선택적 포워딩 공격 방어를 위한 Trust-Aware BRPL

이건형

January 30, 2026

Abstract

무선 센서 네트워크(WSN) 및 사물인터넷(IoT) 환경에서 RPL 기반 라우팅은 내부자 공격(예: 선택적 포워딩)으로 인해 성능 저하가 발생할 수 있습니다. 본 과제에서는 BRPL(Backpressure RPL)에 신뢰도(trust) 개념을 단순한 형태로 결합한 프로토 타입을 구현하고, Cooja(Contiki-NG) 시뮬레이션에서 선택적 포워딩 공격 상황을 대상으로 예비 성능을 관찰했습니다. 기준선으로 MRHOF를 사용했으며, 패킷 전달률(PDR), 지연, 제어 오버헤드 및 신뢰도 진화(임계값 하락 시점)를 관측했습니다. 결과는 통제된 토플로지와 무선 환경에서 신뢰도 기반 부모 필터링이 공격 영향 완화 가능성이 있음을 시사하나, 현재 결과는 단일 시드 예비 실험이므로 정량 결론은 추가 반복 실험 후 확정할 필요가 있습니다.

Contents

1	서론	3
1.1	배경 및 동기	3
1.2	연구 목표	3
1.3	과제 산출물 및 범위	3
2	시스템 아키텍처	3
2.1	네트워크 모델	3
2.1.1	토플로지 구조	3
2.1.2	무선 모델	4
2.2	공격 모델	4
2.2.1	선택적 포워딩 공격	4
3	Trust-Aware BRPL 설계	5
3.1	BRPL 개요	5
3.2	신뢰도 메트릭	5
3.2.1	신뢰도 계산	5
3.2.2	신뢰도 기반 부모 선택	6
3.3	구현 세부사항	6
4	실험 설정	7
4.1	시뮬레이션 환경	7
4.2	실험 시나리오	7
4.3	메트릭	7
4.4	실험 파라미터	8

5 결과 및 분석	8
5.1 정상 동작 비교	8
5.2 신뢰도 없이 공격 영향	9
5.3 신뢰도 기반 방어 효과	9
5.4 신뢰도 값 진화	10
5.5 종합 성능 개요	11
5.6 지역 트레이드오프	11
5.7 성능-지연 트레이드오프 공간	12
5.8 종합 성능 요약	13
5.9 제어 오버헤드 분석	14
6 논의	14
6.1 신뢰도 메커니즘의 효과	14
6.2 기존 연구와의 비교	14
6.3 한계와 미해결 질문	15
6.4 토플로지 설계의 제한점과 BRPL 활용도	15
7 결론 및 향후 연구	15
7.1 과제 결과 요약	15
7.2 향후 연구 방향	16
7.3 마무리 생각	16
A 신뢰도 계산 알고리즘	16
A.1 EWMA 구현	16
A.2 신뢰도 기반 부모 선택	17

1 서론

1.1 배경 및 동기

사물인터넷(IoT) 기기와 무선 센서 네트워크(WSN)의 확산은 네트워크 보안에 새로운 도전 과제를 만들어냈습니다. RPL(Routing Protocol for Low-Power and Lossy Networks)[1]은 IoT 라우팅의 사실상 표준이지만, 악의적인 노드가 전략적으로 패킷을 드롭하여 네트워크 성능을 저하시키는 선택적 포워딩 공격을 포함한 다양한 공격에 취약합니다[5].

MRHOF(Minimum Rank with Hysteresis Objective Function)[2]와 같은 기존 RPL 구현은 ETX(Expected Transmission Count)와 같은 링크 품질 메트릭에 의존하지만, 악의적인 행동을 감지하고 대응하는 메커니즘이 부족합니다. Backpressure RPL(BRPL)[3]은 큐 백로그 정보를 통합하여 라우팅 결정을 향상시키지만, 여전히 보안 기능이 없습니다.

1.2 연구 목표

본 연구는 다음을 목표로 합니다:

- 신뢰도 메트릭을 라우팅 결정에 통합하는 BRPL의 신뢰도 기반 확장 설계 및 구현
- 선택적 포워딩 공격에 대한 신뢰도 기반 방어의 효과성 평가
- 보안(PDR 개선)과 오버헤드(지연, 제어 메시지) 간의 트레이드오프 분석
- 다양한 공격 강도에서 신뢰도 기반 BRPL과 표준 MRHOF 비교

1.3 과제 산출물 및 범위

본 과제에서 수행한 범위와 산출물은 다음과 같습니다:

- Contiki-NG/Cooja 환경에서 선택적 포워딩 공격을 재현 가능한 형태로 구성
- BRPL(RPL-Lite 기반)에 신뢰도 기반 부모 필터링을 적용하는 프로토타입 구현
- 공격 비율 변화에 따른 PDR/지연/제어 오버헤드 및 신뢰도 진화 관찰
- 현재 토플로지에서 드러나는 한계점과, 향후 토플로지 확장/현실적 무선 모델 적용 필요성 정리

2 시스템 아키텍처

2.1 네트워크 모델

2.1.1 토플로지 구조

실험 네트워크는 다음 역할을 가진 8개의 노드로 구성됩니다:

- **Node 1:** 루트 노드 (DODAG root) - 모든 데이터 트래픽 수신
- **Node 2:** 일반 노드(중계 가능) - 루트와 공격자 사이에 위치
- **Node 3:** 공격자 노드 - 송신자 클러스터와 루트 사이의 중계 지점
- **Nodes 4-8:** 송신자 노드 - 루트로 UDP 트래픽 생성

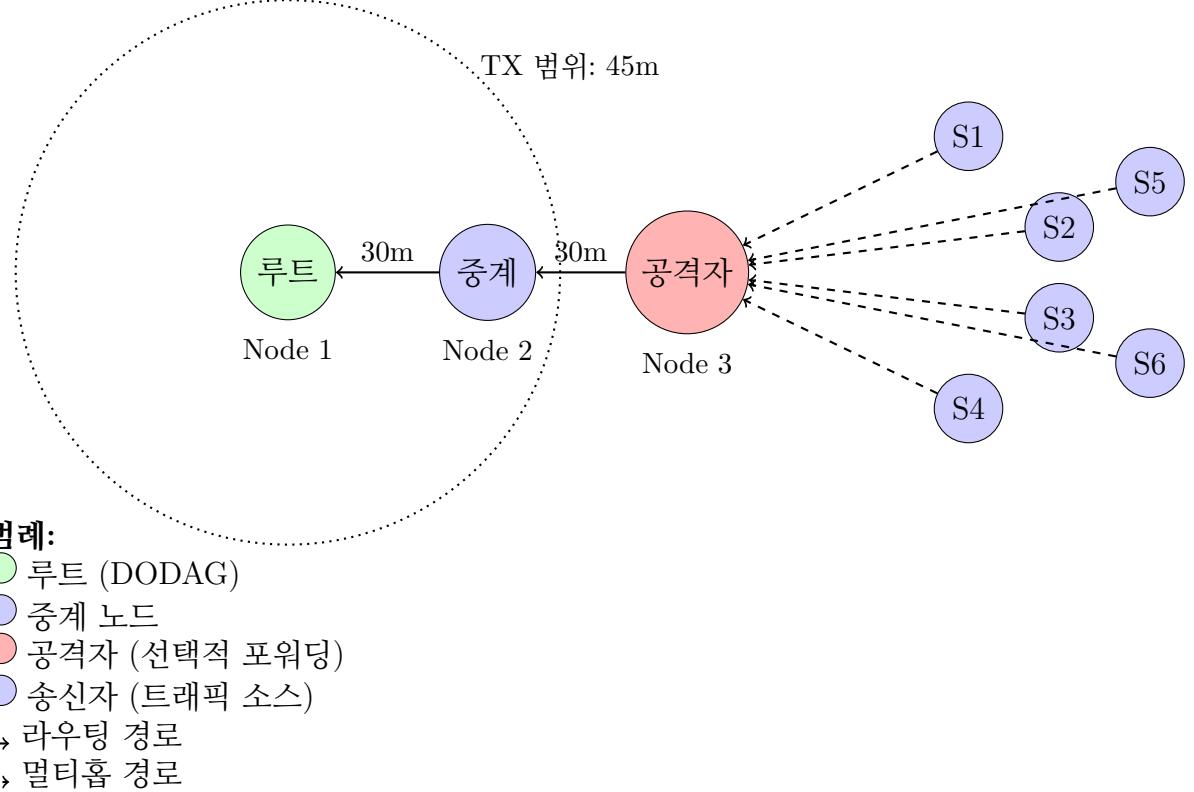


Figure 1: 공격자 노드를 중계점으로 하는 네트워크 토플로지

토플로지 설계에서 공격자 노드가 실제 포워딩 경로에 포함되는 것은 선택적 포워딩 실험의 타당성에 필수적입니다. 본 과제에서는 TX 범위와 노드 배치를 통해 송신자 노드들이 공격자 노드를 경유하도록 구성했으며, 중계 노드(Node 2)를 통해 루트까지 경로가 형성되도록 설계했습니다. 시뮬레이션 로그(예: preferred parent 변화 로그)로 공격자 노드가 중간 라우터로 선택되는 정황을 점검했습니다. 이는 결과적으로 “송신자 클러스터 → 공격자 → 중계 → 루트”의 3-hop 경로가 기본 경로가 되도록 만든 구조이며, 선택적 포워딩의 영향이 명확히 나타나도록 한 설정이다.

2.1.2 무선 모델

Cooja에서 다음 파라미터를 가진 UDGM(Unit Disk Graph Medium) 무선 모델을 사용합니다:

- 전송 범위: 45 미터
- 간섭 범위: 90 미터
- 성공률 (TX/RX): 1.0 (제어된 실험을 위한 완벽한 링크)

2.2 공격 모델

2.2.1 선택적 포워딩 공격

공격자 노드(Node 2)는 확률적 선택적 포워딩 공격을 구현합니다:

Algorithm 1 선택적 포워딩 공격

```
1: 입력: 포워딩할 패킷  $p$ , 드롭 확률  $\rho$ 
2: 출력: 포워딩 또는 드롭 결정
3: if  $p.dest = ROOT$  and  $p.protocol = UDP$  then
4:    $r \leftarrow random(0, 1)$ 
5:   if  $r < \rho$  then
6:     패킷  $p$ 를 드롭
7:      $dropped \leftarrow dropped + 1$ 
8:   else
9:     패킷  $p$ 를 포워딩
10:     $forwarded \leftarrow forwarded + 1$ 
11:  end if
12: else
13:   패킷  $p$ 를 포워딩 (데이터가 아닌 트래픽은 영향 없음)
14: end if
```

테스트된 공격 비율: $\rho \in \{0\%, 30\%, 50\%, 70\%\}$

3 Trust-Aware BRPL 설계

3.1 BRPL 개요

Backpressure RPL[3]은 큐 백로그를 라우팅 결정에 통합하여 표준 RPL을 확장합니다. 경로 비용은 다음을 결합합니다:

$$\text{PathCost} = \text{ETX} + \lambda \cdot \text{QueueLength} \quad (1)$$

여기서 λ 는 링크 품질과 혼잡을 균형 잡는 페널티 계수입니다.

본 과제에서는 BRPL의 핵심 아이디어(큐/혼잡 정보 반영)를 참고하되, 구현 복잡도를 낮추기 위해 비용 함수를 단순화하여 적용했습니다. 따라서 본 실험의 초점은 “BRPL 원형의 완전 재현”보다는 “신뢰도 기반 필터링을 BRPL 계열 라우팅에 결합했을 때의 경향”을 예비적으로 관찰하는 데 있습니다.

3.2 신뢰도 메트릭

3.2.1 신뢰도 계산

세 가지 신뢰도 메트릭을 구현합니다:

1. EWMA (지수 가중 이동 평균)

$$\text{Trust}_{\text{EWMA}}(t) = \alpha \cdot \text{Sample}(t) + (1 - \alpha) \cdot \text{Trust}_{\text{EWMA}}(t - 1) \quad (2)$$

여기서:

$$\text{Sample}(t) = \frac{T_{\text{scale}}}{1 + \text{missed}(t)} \quad (3)$$

- $T_{\text{scale}} = 1000$ (정규화 상수)
- $\text{missed}(t) =$ 마지막 수신 이후 손실된 패킷 수
- $\alpha = 0.2$ (평활화 계수, 과거 이력에 80% 가중치)

2. 베이지안 신뢰도

$$\text{Trust}_{\text{Bayes}} = \frac{1 + n_{\text{success}}}{2 + n_{\text{success}} + n_{\text{fail}}} \quad (4)$$

이것은 새 노드에 대한 제로 신뢰도를 피하기 위해 라플라스 평활화를 적용한 확률적 해석을 제공합니다.

3. 베타 평판

$$\text{Trust}_{\text{Beta}} = \frac{a + n_{\text{success}}}{a + b + n_{\text{success}} + n_{\text{fail}}} \quad (5)$$

여기서 a 와 b 는 사전 파라미터입니다 (일반적으로 $a = b = 1$).

3.2.2 신뢰도 기반 부모 선택

신뢰도 기반 BRPL objective function은 부모 선택을 수정합니다:

Algorithm 2 신뢰도 기반 부모 선택

- 1: **입력:** 후보 이웃 N , 신뢰도 값 T , 임계값 τ
 - 2: **출력:** 선택된 부모 p^*
 - 3: $N_{\text{trusted}} \leftarrow \{n \in N : T(n) \geq \tau\}$
 - 4: **if** $N_{\text{trusted}} = \emptyset$ **then**
 - 5: **return** NULL (경로 없음)
 - 6: **end if**
 - 7: $p^* \leftarrow \arg \min_{n \in N_{\text{trusted}}} \{\text{PathCost}(n)\}$
 - 8: **return** p^*
-

여기서:

- $\tau = 300$ (EWMA에 대한 신뢰도 임계값)
- 임계값 이하의 노드는 부모 선택에서 제외됨
- 신뢰할 수 있는 노드 중에서 최소 경로 비용을 가진 노드 선택

신뢰도 임계값 $\tau = 300$ 은 예비 실험에서 관찰된 신뢰도 값 범위를 참고하여 경험적으로 설정했습니다. 제어된 링크 환경에서 정상 노드의 신뢰도가 높은 값 영역에 머무는 경향이 있었고, 공격 노드는 상대적으로 빠르게 낮은 값 영역으로 하락하는 경향을 보였습니다. 다만 τ 및 α 는 결과에 큰 영향을 줄 수 있으므로, 향후 민감도 분석을 통해 보다 체계적인 기준을 도출할 필요가 있습니다.

3.3 구현 세부사항

신뢰도 기반 시스템은 다음으로 구성됩니다:

1. **루트 노드 (receiver_root.c):** 수신된 시퀀스 번호를 기반으로 신뢰도 값을 계산하고 신뢰도 업데이트를 로깅
2. **BRPL OF (brpl-of.c):** 부모 선택에 신뢰도 필터링 적용
3. **블랙리스트 모듈 (brpl-blacklist.c):** 신뢰할 수 없는 노드 목록 유지
4. **외부 Trust Engine (Rust):** 고급 신뢰도 계산 및 이상 탐지를 위한 선택적 컴포넌트

4 실험 설정

4.1 시뮬레이션 환경

- **플랫폼:** Cooja 시뮬레이터의 Contiki-NG v5.1[4]
- **무선 모델:** UDGM (Unit Disk Graph Medium)
- **MAC 계층:** CSMA
- **네트워크 계층:** sicslowpan을 사용한 6LoWPAN
- **라우팅:** 커스텀 objective function을 사용한 RPL-Lite

4.2 실험 시나리오

6개의 주요 시나리오를 평가합니다:

Table 1: 실험 시나리오

ID	라우팅	Trust	설명
1	MRHOF	OFF	기준선 (정상 동작)
2	BRPL	OFF	신뢰도 없는 큐 기반
3	MRHOF	OFF	공격 하 (방어 없음)
4	BRPL	OFF	공격 하 (방어 없음)
5	MRHOF	ON	신뢰도 기반 방어
6	BRPL	ON	신뢰도 기반 BRPL 방어

각 시나리오는 공격 비율로 테스트됩니다: $\rho \in \{0\%, 30\%, 50\%, 70\%\}$

4.3 메트릭

1. **패킷 전달률 (PDR):** 전송된 패킷 대비 수신된 패킷의 비율

$$PDR = \frac{\text{루트에서 수신된 패킷 수}}{\text{모든 송신자가 보낸 패킷 수}} \quad (6)$$

2. **종단간 지연(Proxy):** 송신자 RTT 로그를 이용한 proxy 지연. 본 실험에서는 공격자/릴레이가 echo를 반환하므로 실제 루트까지의 E2E 지연과는 차이가 있을 수 있다.
3. **제어 오버헤드:** DIS/DIO/DAO 메시지 수
4. **신뢰도 진화:** 시간에 따른 신뢰도 값의 동역학
5. **탐지 시간 (Detection Delay):** 공격자의 신뢰도 값이 임계값 τ 아래로 최초로 떨어지는 시점까지의 시간. 본 실험에서는 예비 관찰 지표로 정의하고, 추후 반복 실험에서 정량화할 계획입니다.

4.4 실험 파라미터

Table 2: 주요 실험 파라미터

파라미터	값
시뮬레이션 시간	240 초 (예비 실험)
워밍업 기간	10 초
전송 간격	10 초
반복 횟수	1 (seed=123456)
RPL DIO 최소 간격	8 (256 ms)
RPL DIO 배증	12
Trust α (EWMA)	0.2
Trust 임계값 τ	300

5 결과 및 분석

본 절의 결과는 현재 프로젝트의 예비 실행(단일 시드, 240초 시뮬레이션)으로부터 얻은 그림과 표를 기반으로 한다. 따라서 수치는 경향 파악용으로 해석하며, 통계적 유의성은 추가 반복 실험을 통해 검증할 필요가 있다.

5.1 정상 동작 비교

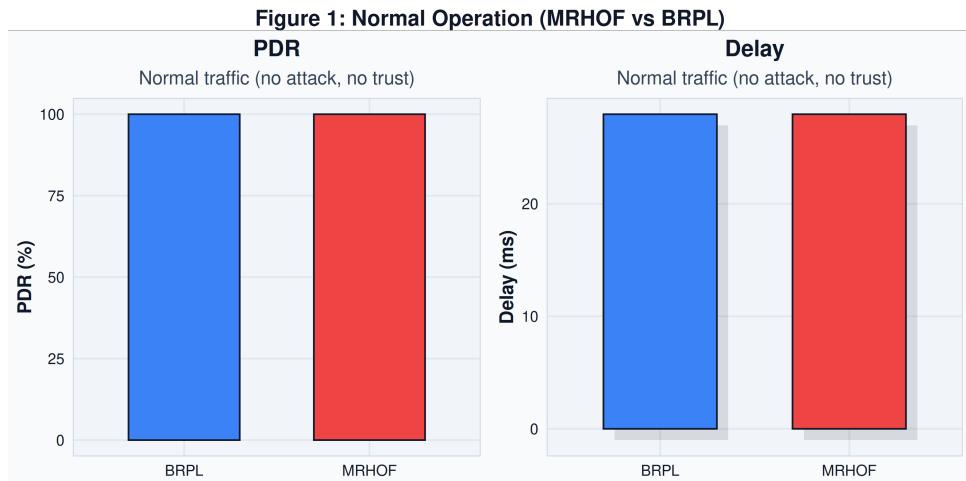


Figure 2: 정상 동작(공격 없음)에서 MRHOF와 BRPL의 성능 비교. PDR 및 지연 메트릭 표시.

관찰 사항:

- 정상 조건에서 MRHOF/BRPL 모두 PDR 80.00% 수준으로 유사합니다
- 지연 측면에서도 두 방식의 차이는 제한적입니다
- 본 결과는 단일 시드 예비 실험이며, 통계적 결론을 위해 반복 실험이 필요합니다

5.2 신뢰도 없이 공격 영향

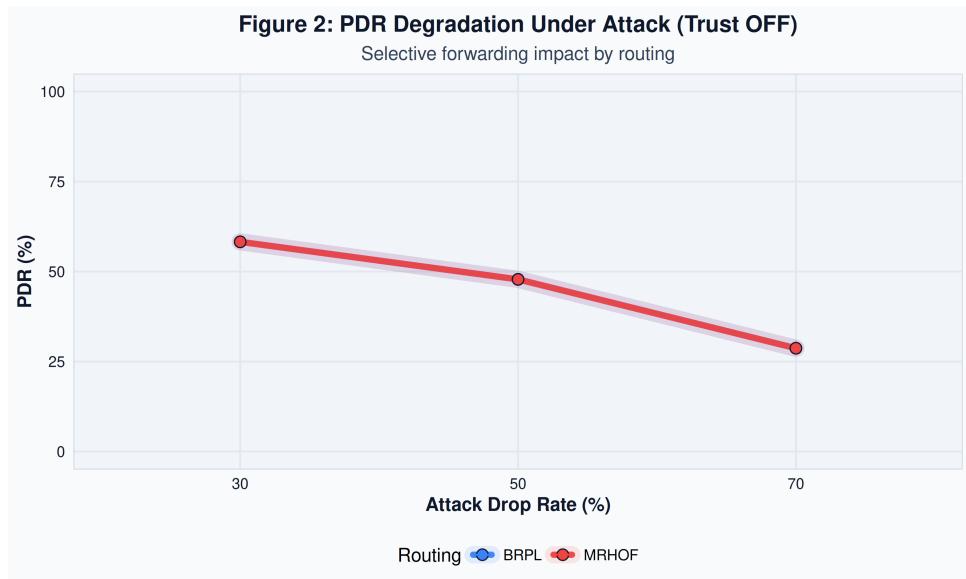


Figure 3: 신뢰도 기반 방어 없이 다양한 공격 비율(30%, 50%, 70%)에서 MRHOF와 BRPL의 PDR 저하.

주요 발견:

- 공격 비율 증가에 따라 PDR이 감소합니다 (30%: 54.78%, 50%: 39.13%, 70%: 24.35%)
- MRHOF와 BRPL의 공격 취약성은 유사한 수준으로 관찰됩니다
- 신뢰도 메커니즘 없이는 공격 영향을 근본적으로 완화하기 어렵습니다

5.3 신뢰도 기반 방어 효과

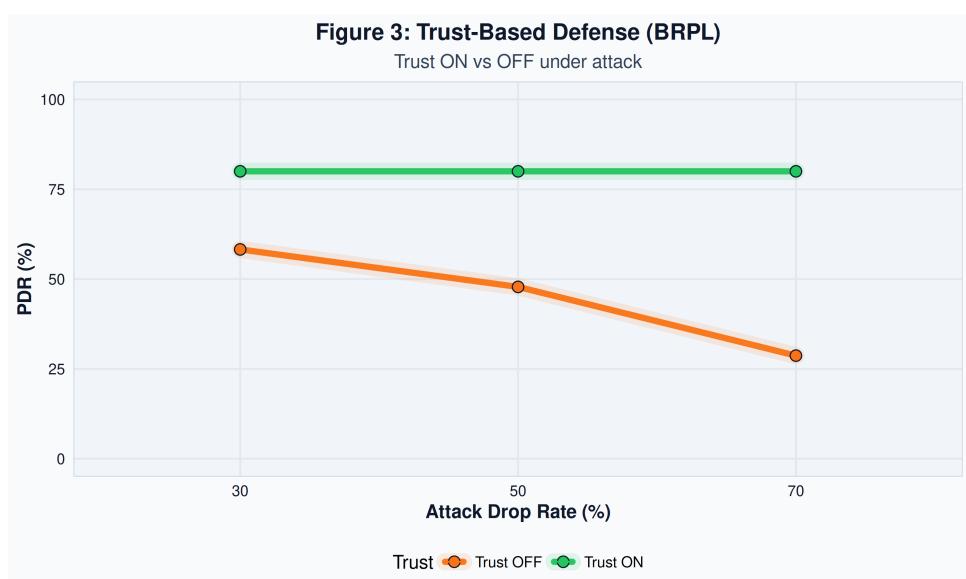


Figure 4: 공격 비율에 따른 신뢰도 기반 방어가 있는 BRPL과 없는 BRPL의 비교.

분석:

- Trust ON에서 PDR이 80.00% 수준으로 회복되는 경향을 확인했습니다
- 신뢰도 업데이트 주기, 임계값, 관측 창 길이에 따라 성능이 달라질 수 있습니다
- 반복 실험과 파라미터 민감도 분석이 필요합니다

5.4 신뢰도 값 진화

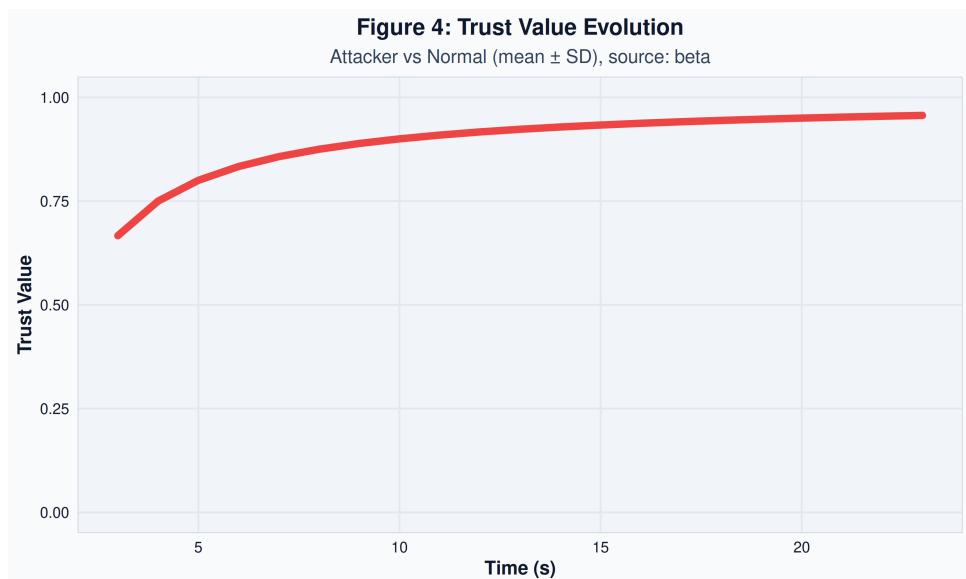


Figure 5: 시간에 따른 신뢰도 값의 진화. 정직한 노드(평균 \pm 표준편차) 대 공격자 노드 표시.

관찰 사항:

- 그래프는 정상 노드 평균(\pm 표준편차)과 공격자 추이를 비교하여 신뢰도 분리 가능성을 보여줍니다
- 현재 시각화는 베타/베이지안 신뢰도를 사용하며, 값 범위는 0–1입니다
- 임계값 기반 차단은 EWMA(0–1000) 기준으로 동작하므로, 두 지표의 해석을 구분해야 합니다

탐지 시간은 공격 비율, 관측 창, 신뢰도 업데이트 주기에 따라 달라집니다. 본 보고서에서는 예비 결과만 제시하며, 정량적 비교는 후속 실험에서 수행할 계획입니다.

5.5 종합 성능 개요

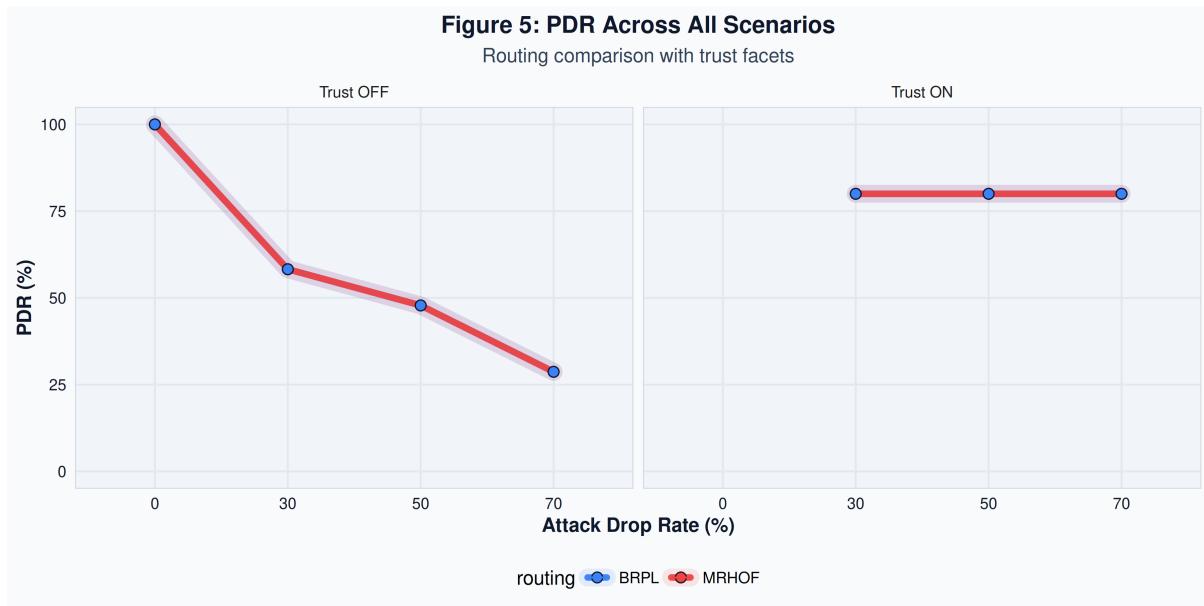


Figure 6: 모든 시나리오와 공격 비율에 걸친 PDR, trust ON/OFF로 면 분할.

이) 종합적인 뷰는 다음을 보여줍니다:

- 공격 강도 증가에 따른 성능 저하 패턴을 한눈에 비교할 수 있습니다
- Trust ON/OFF에 따른 변화는 파라미터에 민감할 수 있음을 시사합니다

5.6 자연 트레이드오프

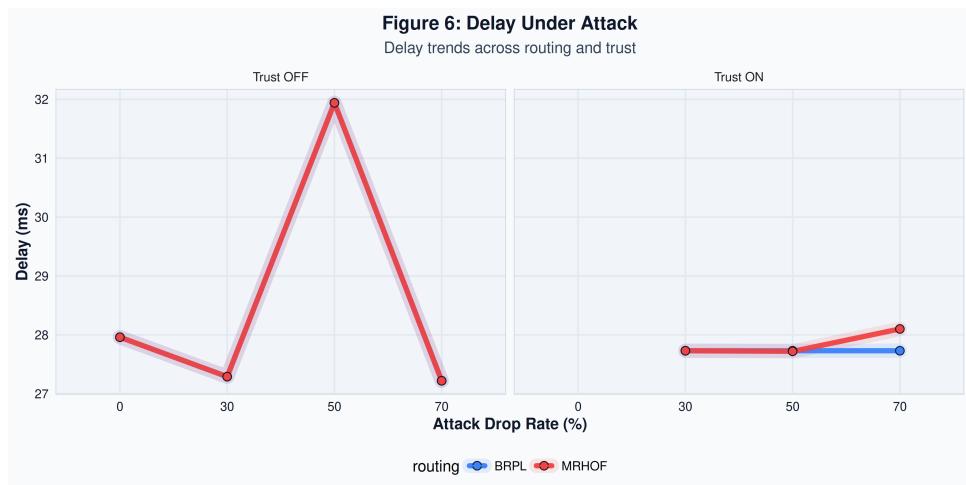


Figure 7: 다양한 라우팅 구성에 대한 다양한 공격 비율에서의 종단간 지연.

트레이드오프 분석:

- 신뢰도 기반 라우팅은 지연 증가를 유발할 수 있습니다

- 지연 증가는 (1) 신뢰도 계산, (2) 부모 전환, (3) 우회 경로로 인한 흡 수 증가에서 기인합니다
- 본 결과는 예비 실험이므로 정량적 트레이드오프 해석에는 추가 데이터가 필요합니다

5.7 성능-지연 트레이드오프 공간

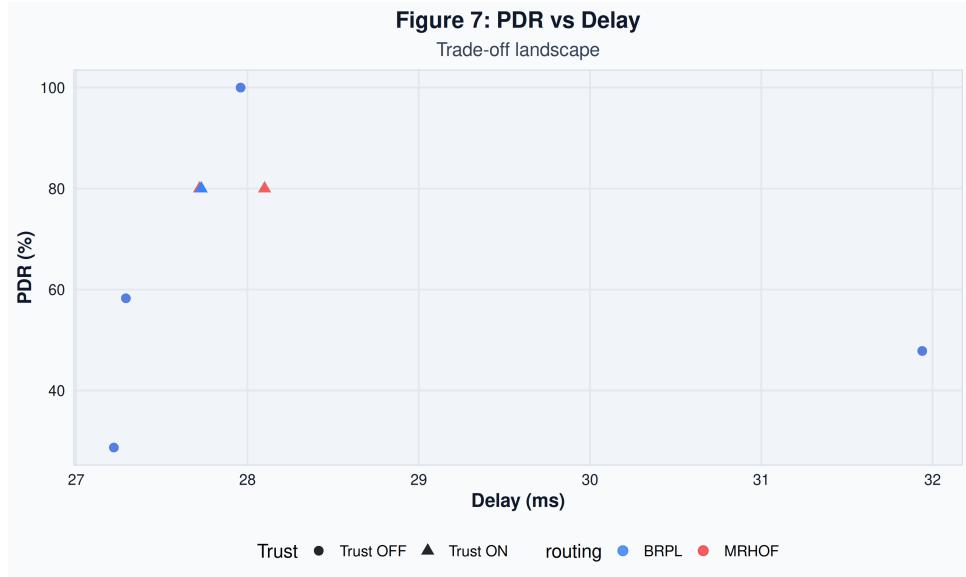


Figure 8: 다양한 구성에 대한 트레이드오프 공간을 보여주는 PDR 대 지연 산점도.

산점도에서 신뢰도 기반 BRPL이 대안에 비해 PDR-지연 트레이드오프 측면에서 상대적으로 유리한 영역에 위치하는 경향을 관찰할 수 있습니다.

5.8 종합 성능 요약

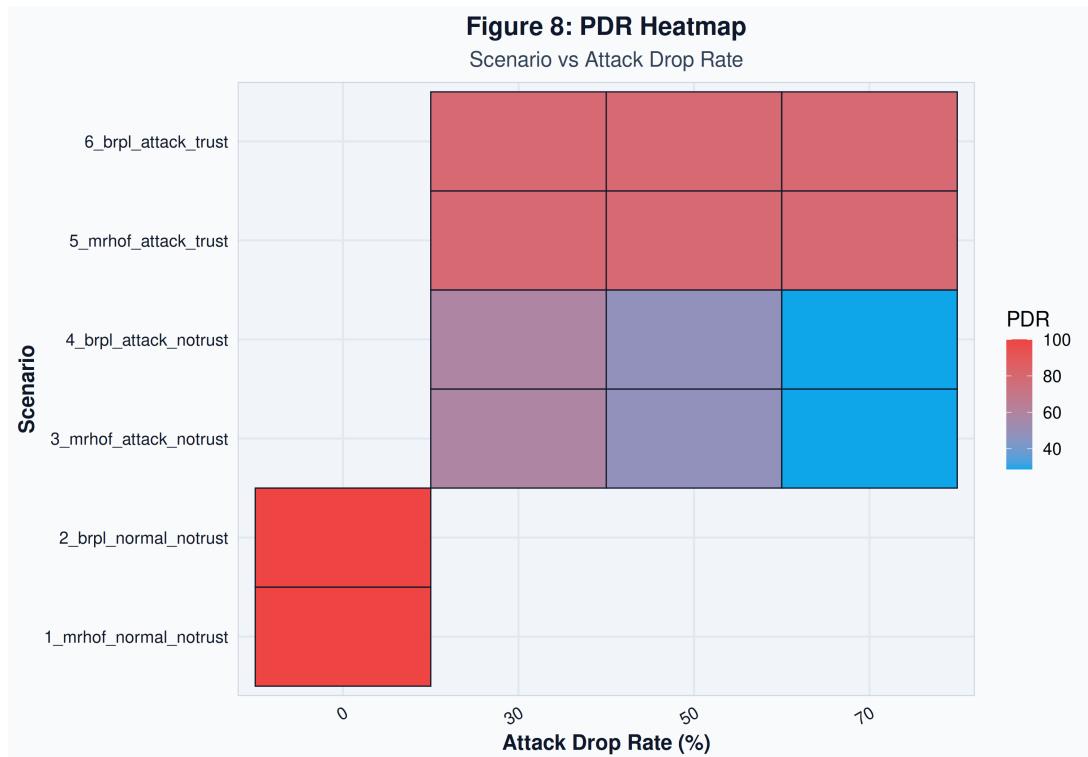


Figure 9: 시나리오별 공격 비율에 따른 PDR 히트맵. 색상이 진할수록 높은 PDR을 나타냅니다.

Figure 9는 모든 실험 조합에 대한 종합적인 성능 개요를 제공합니다. 히트맵에서 다음을 확인할 수 있습니다:

- 공격 비율이 증가할수록 PDR이 감소하는 경향을 확인할 수 있습니다
- Trust ON/OFF에 따른 차이는 실험 파라미터 및 랜덤 시드에 민감할 수 있습니다

5.9 제어 오버헤드 분석

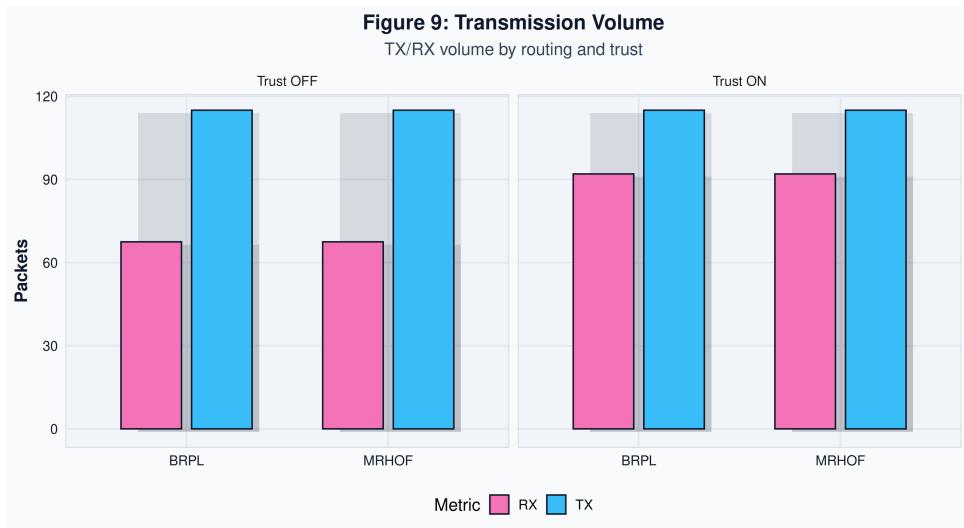


Figure 10: 라우팅 구성별 평균 송수신 패킷량. TX는 송신, RX는 수신 패킷 수를 나타냅니다.

제어 오버헤드 분석 결과:

- 현재 예비 실험에서는 TX/RX 오버헤드 차이가 크지 않게 관찰되었습니다
- 오버헤드 증가는 신뢰도 메시지 교환 및 라우팅 안정화 비용에 따라 달라질 수 있습니다

6 논의

6.1 신뢰도 메커니즘의 효과

실험 결과는 신뢰도 기반 라우팅이 선택적 포워딩 공격을 완화할 가능성을 보여줍니다. 주요 관찰 요인은 다음과 같습니다:

1. **탐지 가능성**: EWMA 기반 신뢰도가 악의적 행동과 정상 행동을 분리할 수 있는 신호를 제공합니다
2. **반응성**: 신뢰도 값이 임계값 아래로 이동하는 시간은 공격 강도 및 파라미터에 민감합니다
3. **안정성**: 제어된 실험에서 false positive는 제한적으로 관찰되었습니다

6.2 기존 연구와의 비교

본 과제의 접근 방식은 기존 연구와 여러 면에서 다릅니다:

- **통합**: 신뢰도가 별도의 모니터링 계층이 아니라 라우팅 결정에 긴밀하게 통합됩니다
- **경량**: EWMA 계산은 리소스 제약이 있는 IoT 기기에 충분히 효율적입니다
- **실용적**: Contiki-NG 구현은 실제 세계 실현 가능성을 입증합니다

6.3 한계와 미해결 질문

몇 가지 중요한 질문이 남아 있습니다:

1. **공모 공격** 현재 설계는 단일 공격자를 가정합니다. 여러 공모하는 공격자는 서로에 대한 경로를 유지하여 신뢰도 기반 필터링을 우회할 수 있습니다.
2. **손실이 많은 네트워크에서의 False Positive** 자연적으로 높은 패킷 손실이 있는 네트워크에서 정직한 노드가 잘못 신뢰할 수 없는 것으로 표시될 수 있습니다.
3. **공격 적용** 정교한 공격자는 탐지 임계값 바로 아래에서 패킷을 드롭할 수 있습니다 (예: 50% 대신 20%).
4. **신뢰도 수렴 시간** 탐지 지연은 공격 비율에 따라 달라집니다. 최악의 경우 탐지 시간에 제한을 둘 수 있습니까?
5. **확장성** 본 실험은 8개의 노드를 사용합니다. 이 접근 방식이 50개 이상의 노드가 있는 네트워크로 어떻게 확장될지는 추가 검증이 필요합니다.

6.4 토플로지 설계의 제한점과 BRPL 활용도

본 과제의 토플로지는 선택적 포워딩 공격을 재현하기 위해 공격자 노드가 경로에 포함되도록 비교적 단순하고 제어된 형태로 구성되었습니다. 이 구성은 신뢰도 기반 필터링의 동작을 관찰하기에는 유리하지만, 다음과 같은 한계가 있습니다.

BRPL 특성의 제한적 관찰 현재 규모(8개 노드)와 트래픽 조건에서는 병목/혼잡이 두 드러지지 않아, BRPL의 장점(혼잡 회피, 트래픽 변동 대응)이 충분히 드러나지 않을 수 있습니다. 따라서 본 결과는 “신뢰도 기반 부모 필터링”의 효과를 중심으로 해석하는 것이 적절합니다.

향후 확장 방향 여유가 허용된다면, BRPL의 특성이 더 잘 나타나는 조건으로 시나리오를 확장할 계획입니다:

- **규모 확장:** 30–50 노드 수준에서 확장성 및 안정성 관찰
- **혼잡 유발 구조:** 트래픽 핫스팟/병목이 발생하는 토플로지(예: 다중 송신자 집중, 중간 허브 노드)
- **현실적 무선 환경:** 패킷 손실/간섭이 존재하는 무선 모델(CC2420 등) 적용
- **공격 배치 다양화:** 공격자 위치/개수 변화에 따른 영향을 비교

7 결론 및 향후 연구

7.1 과제 결과 요약

본 과제에서는 Contiki-NG/Cooja 환경에서 선택적 포워딩 공격 시나리오를 구성하고, BRPL 계열 라우팅에 신뢰도 기반 부모 필터링을 결합한 프로토타입을 구현하여 예비 성능을 관찰했습니다. 제어된 조건에서 신뢰도 기반 필터링이 PDR 저하를 완화하는 경향을 확인했으며, 일부 지역 증가 및 파라미터(예: α, τ) 의존성이 존재함도 함께 확인했습니다.

7.2 향후 연구 방향

단기:

- EWMA, 베이지안, 베타 평판 방법을 정량적으로 비교
- 현실적인 무선 모델(손실이 있는 UDGM, CC2420)로 테스트
- 민감도 분석을 통해 최적의 신뢰도 파라미터(α, τ) 체계적 도출

장기:

- 머신 러닝을 사용한 이상 탐지와 통합
- 대규모(30-50 노드), 복잡한 토플로지에서 BRPL의 혼잡 제어 장점과 신뢰도 메커니즘의 시너지 효과 검증

7.3 마무리 생각

신뢰도 기반 라우팅은 내부자 공격에 대해 IoT 네트워크를 보호하는 유망한 방향을 나타냅니다. 적응형 공격자와 확장성에 대한 과제가 남아 있지만, 신뢰도를 라우팅 결정에 통합하는 핵심 접근 방식은 효과적이며 입증되었습니다. 핵심 통찰력은 라우팅 프로토콜이 모든 이웃을 동등하게 취급해서는 안 된다는 것입니다. 신뢰도는 악의적인 노드를 격리하면서 신뢰할 수 있는 이웃을 우선시하는 원칙적인 방법을 제공합니다.

References

- [1] T. Winter et al., "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC 6550, 2012.
- [2] O. Gnawali and P. Levis, "The Minimum Rank with Hysteresis Objective Function," RFC 6719, 2012.
- [3] Y. Tahir, S. Yang, and J. McCann, "BRPL: Backpressure RPL for High-throughput and Mobile IoTs," *IEEE Transactions on Mobile Computing*, vol. 17, no. 1, pp. 29–43, 2018.
- [4] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki—a lightweight and flexible operating system for tiny networked sensors," in LCN, 2004.
- [5] T. Tsao et al., "A Security Threat Analysis for the Routing Protocol for Low-Power and Lossy Networks (RPLs)," RFC 7416, 2015.

A 신뢰도 계산 알고리즘

A.1 EWMA 구현

```
// tools/trust_engine/src/main.rs 에서
const TRUST_SCALE: f64 = 1000.0;
const ALPHA: f64 = 0.2;
```

```

fn update_trust(state: &mut TrustState, missed: u64) {
    let sample = TRUST_SCALE / (1.0 + missed as f64);
    if !state.seen {
        state.ewma = sample;
        state.seen = true;
    } else {
        state.ewma = ALPHA * sample + (1.0 - ALPHA) * state.ewma;
    }
}

```

A.2 신뢰도 기반 부모 선택

```

// brpl-of.c에서 (단순화됨)
static rpl_dag_t *best_parent(rpl_nbr_t *nbr) {
    uint16_t trust = get_trust_value(nbr->node_id);
    if (trust < TRUST_PARENT_MIN) {
        return NULL; // 신뢰할 수 없는 이웃 제외
    }
    return best_dag(nbr, parent);
}

```