Detecção de Ataques ECN Não-Responsivos em Arquiteturas L4S por Meio de Aprendizado Supervisionado

Lucas de Sousa Correia[1], Jaaziel Batista da Silva[1], José Silva Floriano[1], Leandro Cavalcanti de Almeida[1,2] e Paulo Ditarso Maciel Jr.[1,2]

[1]Curso Superior de Tecnologia em Redes de Computadores (CSTRC)

[2]Programa de Pós-Graduação em Tecnologia da Informação (PPGTI)

Unidade Acadêmica de Informação e Comunicação
Instituto Federal da Paraíba (IFPB) – João Pessoa/PB – Brasil

{lucas.correia,jaaziel.silva,jantony.jose}@academico.ifpb.edu.br

{leandro.almeida,paulo.maciel}@ifpb.edu.br

Abstract. *Low Latency, Low Loss, and Scalable Throughput (L4S) has emerged as a promising architecture to reduce latency and jitter in IP networks through coupled dual-queue active queue management. However, its correct operation depends on end-host compliance with explicit congestion signals. This paper presents the design rationale, implementation strategy, and experimental evaluation of an intrusion detection approach for identifying non-responsive ECN attacks in L4S environments. The study combines a virtualized testbed built with VirtualBox, Vagrant, and Ansible with traffic generation using iperf, packet inspection, and a supervised machine-learning classifier based on decision trees. The results show that a malicious sender that marks packets with ECT(1) while ignoring congestion feedback can monopolize the bottleneck link, reducing the throughput of legitimate L4S and Classic flows by more than 99% and 97%, respectively. In addition, the proposed IDS captured a consistent behavioral signature of the attack using flow- and queue-level features, indicating the feasibility of deploying lightweight, interpretable security mechanisms in L4S routers.*

Resumo. *Baixa Latência, Baixa Perda e Taxa de Transferência Escalável (L4S) surgiu como uma arquitetura promissora para reduzir a latência e a variação de atraso (jitter) em redes IP por meio do gerenciamento ativo de filas duplas acopladas. No entanto, seu funcionamento correto depende da conformidade do host final com os sinais explícitos de congestionamento. Este artigo apresenta a justificativa do projeto, a estratégia de implementação e a avaliação experimental de uma abordagem de detecção de intrusão para identificar ataques ECN não responsivos em ambientes L4S. O estudo combina um ambiente de teste virtualizado construído com VirtualBox, Vagrant e Ansible com geração de tráfego usando o iperf, inspeção de pacotes e um classificador de aprendizado de máquina supervisionado baseado em árvores de decisão. Os resultados mostram que um remetente malicioso que marca pacotes com ECT(1) enquanto ignora o feedback de congestionamento pode monopolizar o link de gargalo, reduzindo a taxa de transferência de fluxos legítimos L4S e Clássicos em mais de 99% e 97%, respectivamente. Além disso, o IDS proposto capturou uma assinatura comportamental consistente do ataque usando recursos de nível de fluxo*

e fila, indicando a viabilidade de implantar mecanismos de segurança leves e interpretáveis em roteadores L4S.

1. Introdução

Aplicações interativas, como jogos em nuvem, realidade virtual, teleoperação e serviços multimídia em tempo real, impõem requisitos cada vez mais rigorosos de latência, *jitter* e estabilidade fim a fim. Nesse contexto, a arquitetura L4S foi proposta para conciliar baixa latência, baixa perda e alta escalabilidade de *throughput*, mitigando os efeitos do *bufferbloat* por meio de um esquema de *Active Queue Management* (AQM) com filas duplas acopladas [Briscoe et al. 2023a, Schepper et al. 2023].

A lógica de operação do L4S é baseada na coexistência entre dois domínios de tráfego. A fila *Low Latency* (L) atende fluxos escaláveis que utilizam *Explicit Congestion Notification* (ECN) e respondem rapidamente a marcações de congestionamento, enquanto a fila *Classic* (C) atende fluxos tradicionais, como TCP Reno e CUBIC. O acoplamento entre as filas procura garantir justiça de banda entre fluxos com diferentes semânticas de controle de congestionamento. Em um cenário ideal, esse arranjo permite reduzir a latência sem comprometer a utilização do enlace [Briscoe et al. 2023b, Schepper et al. 2023].

Entretanto, essa arquitetura incorpora uma hipótese forte de cooperação dos sistemas finais. Um emissor malicioso pode explorar o mecanismo de classificação da fila L ao marcar seus pacotes como habilitados para ECN sem, contudo, implementar a lógica de reação aos sinais de congestionamento. Como consequência, o fluxo invasor obtém prioridade de escalonamento, mantém elevada taxa de envio mesmo sob intensa marcação de congestionamento e induz degradação severa nos fluxos legítimos. Esse comportamento caracteriza o ataque ECN não-responsivo [Schepper et al. 2023, Briscoe et al. 2023a].

Diante desse problema, este artigo propõe um estudo experimental sobre a vulnerabilidade da arquitetura L4S e sobre a viabilidade de detectar esse tipo de ataque com aprendizado supervisionado. Este trabalho consolida a estrutura arquitetural do L4S, a metodologia experimental, a configuração do ambiente, a engenharia de atributos e os resultados de *throughput*, articulando-os a uma análise crítica das evidências obtidas.

As contribuições principais deste trabalho são as seguintes: *(i)* sistematizar o vetor de ataque ECN não-responsivo em redes L4S; *(ii)* descrever um *testbed* controlado e reprodutível para avaliação do problema; *(iii)* apresentar resultados quantitativos que evidenciam a inversão de justiça provocada pelo atacante; e *(iv)* estruturar uma abordagem de Sistema de Detecção de Intrusão (IDS) baseada em árvore de decisão, com foco em interpretabilidade e baixa sobrecarga computacional.

Adiante neste artigo, a Seção 2 apresenta os trabalhos relacionados. A Seção 3 discute a fundamentação teórica e o modelo de ameaça adotado. A Seção 4 descreve o ambiente experimental e o protocolo de avaliação. A Seção 5 detalha a arquitetura e o treinamento do IDS supervisionado. A Seção 6 apresenta os resultados obtidos, enquanto a Seção 7 analisa a capacidade de detecção do modelo. Por fim, as Seções 8 e 9 sintetizam, respectivamente, as limitações e desdobramentos futuros, bem como as conclusões deste estudo.

2. Trabalhos Relacionados

Esta pesquisa baseia-se no conjunto normativo do IETF para L4S: a arquitetura do serviço [Briscoe et al. 2023a], a semântica ECN para tráfego escalável [Briscoe et al. 2023b] e o mecanismo *DualQ Coupled AQM* [Schepper et al. 2023]. Esses documentos definem o modelo de operação de duas filas acopladas e deixam explícita a dependência de cooperação dos emissores para que a justiça e a baixa latência sejam preservadas.

No contexto mais amplo de controle de congestionamento e gerenciamento de filas, a literatura clássica de ECN [Ramakrishnan et al. 2001] e as recomendações de AQM da IETF [Baker and White 2015] estabelecem a importância de sinais explícitos de congestionamento e da disciplina de filas para estabilidade do enlace. Trabalhos como CoDel [Nichols and Jacobson 2018] reforçam o foco em redução de atraso de enfileiramento, mas não endereçam diretamente o caso em que o emissor manipula a classificação de tráfego e ignora o feedback de congestionamento.

Em relação à adoção e comportamento do ecossistema L4S, estudos recentes discutem os incentivos e limitações práticas de implantação parcial do TCP Prague em ambientes heterogêneos [Sarpkaya et al. 2024]. Entretanto, a maior parte dessas análises concentra-se no desempenho e na coexistência entre fluxos cooperativos, sem examinar em profundidade o vetor de abuso no qual um emissor utiliza o ECN de modo oportunista para obter vantagem de banda.

Com relação à avaliação experimental, Oljira et al. [Oljira et al. 2020] validam propriedades de compartilhamento e latência da arquitetura L4S sob cenários controlados, evidenciando ganhos de atraso com desafios de coexistência que dependem de correta configuração do AQM e dos emissores. Mais recentemente, propostas orientadas a redes programáveis e 5G ampliam o debate para implantação prática. Messaoudi et al. [Messaoudi et al. 2024] exploram controle de congestionamento L4S com suporte de SDN em cenários Beyond 5G, enquanto Monteiro et al. [Monteiro et al. 2024] apresentam um estudo de caso em rede privada 5G industrial com vídeo em tempo real.

Esses trabalhos reforçam a maturidade do L4S do ponto de vista de desempenho e integração arquitetural, mas mantêm foco predominante em eficiência de transporte, orquestração e qualidade de experiência. Em geral, a hipótese de emissor cooperativo continua implícita, com pouca ênfase na modelagem de comportamento malicioso deliberado e em mecanismos de detecção de abuso baseados em observabilidade de fila e fluxo.

Assim, há uma lacuna específica na interseção entre desempenho de L4S e segurança operacional: faltam estudos experimentais que caracterizem, com métricas de fluxo e fila, o ataque ECN não-responsivo e sua detecção em tempo de execução. Este trabalho avança nesse ponto ao combinar *testbed* reprodutível, quantificação da inversão de justiça e validação de um IDS supervisionado interpretável no roteador de gargalo.

3. Fundamentação e Modelo de Ameaça

Esta seção apresenta os fundamentos da análise, abordando o funcionamento da arquitetura L4S com DualQ acoplado, a dependência da cooperação dos emissores e a caracterização do ataque ECN não-responsivo, além de discutir sua dinâmica e seus impactos sobre justiça de banda e disponibilidade do enlace.

3.1. Arquitetura L4S e Mecanismo DualQ

A arquitetura L4S foi concebida para oferecer suporte a fluxos de baixa latência por meio de um esquema *Dual Queue Coupled AQM* [Briscoe et al. 2023a, Schepper et al. 2023]. Nesse arranjo, a fila L é propositalmente rasa e priorizada, sendo destinada a fluxos que utilizam mecanismos escaláveis de controle de congestionamento, como o TCP Prague. Já a fila C é voltada ao tráfego tradicional, que tende a conviver com filas mais profundas e a reagir a perda de pacotes ou a sinalização ECN convencional.

O elemento central do modelo é o acoplamento probabilístico entre as filas. A carga observada na fila clássica influencia a política de marcação na fila L, de modo a evitar que fluxos escaláveis obtenham vantagem injusta apenas por transitarem na fila prioritária. Assim, o sistema pressupõe uma malha fechada de controle em que os emissores interpretam corretamente os sinais CE (*Congestion Experienced*) e reduzem sua taxa de transmissão sempre que necessário [Briscoe et al. 2023b, Schepper et al. 2023].

3.2. Ataque ECN Não-responsivo

No ECN clássico, conforme definido pela RFC 3168 [Ramakrishnan et al. 2001], os códigos ECT(0) (ex.: pacote não usa ECN) e ECT(1) (ex.: pacote suporta ECN) são tratados de forma equivalente pelos roteadores, indicando apenas que o pacote é compatível com ECN e pode ser marcado em vez de descartado em situações de congestionamento. Posteriormente, a RFC 8311 [Black 2018] passou a permitir o uso de semânticas alternativas para o campo ECT(1), destacando-se o contexto do L4S, no qual esse código passou a identificar um tipo específico de tráfego associado a tratamento de baixa latência, deixando de representar apenas compatibilidade genérica com ECN.

O ataque investigado neste trabalho rompe justamente a premissa de cooperação dos emissores. O emissor malicioso realiza três ações coordenadas: *(i)* marca seus pacotes com ECT(1) para ser classificado na fila L; *(ii)* ignora deliberadamente o feedback de congestionamento; e *(iii)* mantém a taxa de envio elevada mesmo sob marcação intensiva de CE. Com isso, a fila L deixa de cumprir seu papel de baixa latência e passa a operar sob saturação persistente, contrariando o comportamento esperado pela semântica ECN em L4S [Ramakrishnan et al. 2001, Briscoe et al. 2023b].

O efeito sistêmico do ataque vai além da degradação do próprio domínio L4S. Como o acoplamento tenta preservar justiça entre as filas, o AQM aumenta a pressão de marcação e descarte também sobre a fila clássica. Assim, fluxos cooperativos são punidos exatamente por seguirem o protocolo, enquanto o atacante é recompensado com maior participação de banda. Esse comportamento configura uma inversão de justiça e transforma o mecanismo de controle em um amplificador do ataque [Schepper et al. 2023].

3.3. Aprendizado Supervisionado para Classificação dos Ataques

O aprendizado supervisionado é um paradigma de modelagem estatística no qual um classificador aprende, a partir de exemplos rotulados, uma função que associa vetores de atributos a classes de interesse [Bishop 2006, Amor et al. 2004]. No contexto deste trabalho, cada amostra representa o comportamento observável de um fluxo em janelas temporais de monitoramento, enquanto o rótulo indica se o fluxo é *benigno* ou *malicioso*. Esse enquadramento transforma o problema de segurança em uma tarefa de classificação binária,

na qual o modelo deve generalizar padrões de responsividade (ou não) ao congestionamento para dados não vistos durante o treinamento.

Em termos operacionais, a qualidade da classificação depende de três elementos principais: *(i)* representatividade dos atributos, *(ii)* consistência da rotulagem e *(iii)* separabilidade entre classes. Para o cenário de ataque ECN não-responsivo, atributos como razão de marcações CE, frequência de sinalização CWR (*Congestion Window Reduced*), taxa de envio sustentada, tamanho médio de janela TCP e intervalos entre pacotes fornecem uma descrição comportamental diretamente alinhada à semântica do L4S. Enquanto fluxos cooperativos tendem a reduzir emissão sob marcação de congestionamento, fluxos maliciosos mantêm vazão elevada e exibem baixa evidência de reação, formando uma assinatura estatística explorável por modelos supervisionados.

Entre os algoritmos possíveis, árvores de decisão são particularmente adequadas quando se busca equilíbrio entre acurácia, custo computacional e interpretabilidade. O modelo particiona o espaço de atributos por meio de testes condicionais sucessivos (por exemplo, limiares sobre `ratio_ce` ou `ratio_cwr`), produzindo regras explícitas do tipo *se-então*. Essa estrutura é vantajosa para ambientes de rede porque permite inferência rápida em tempo de execução, facilita auditoria por operadores e torna transparente o racional da decisão, aspecto crítico em sistemas de detecção de intrusão.

Além disso, a interpretabilidade da árvore facilita sua integração a mecanismos de resposta, pois cada classificação pode ser associada a políticas específicas, como limitação de taxa, reclassificação de fila ou inspeção adicional. Assim, o uso de árvores de decisão no aprendizado supervisionado não apenas viabiliza a detecção de anomalias, mas também fornece uma base prática para integrar observabilidade, decisão e mitigação em arquiteturas L4S.

4. Metodologia Experimental

Esta seção descreve a metodologia adotada para investigar o ataque ECN não-responsivo em ambiente L4S, detalhando a infraestrutura de experimentação, a definição dos cenários e o protocolo de coleta de dados. Também são apresentados os procedimentos de geração de tráfego, extração e rotulagem de atributos e organização do conjunto de dados.

4.1. Ambiente de Experimentação

O ambiente experimental foi construído como um *testbed* virtualizado, controlado e reproduzível. Para isso, foram utilizadas ferramentas de infraestrutura como código e orquestração de máquinas virtuais. O hipervisor VirtualBox foi empregado como base de virtualização, enquanto Vagrant e Ansible foram utilizados para provisionamento automático dos nós e definição da topologia ilustrada na Figura 1.

A topologia é composta por três emissores conectados a um roteador central que representa o gargalo da rede e executa a disciplina de filas DualPI2 [Schepper et al. 2023], correspondente a uma implementação de referência do DualQ. Os papéis dos nós presentes na topologia são descritos a seguir.

- **Vítima L4S:** *host* legítimo compatível com tráfego escalável, representando um emissor que reage adequadamente às marcações ECN.
- **Vítima Clássica:** *host* legítimo com controle de congestionamento tradicional, representando tráfego CUBIC ou Reno.

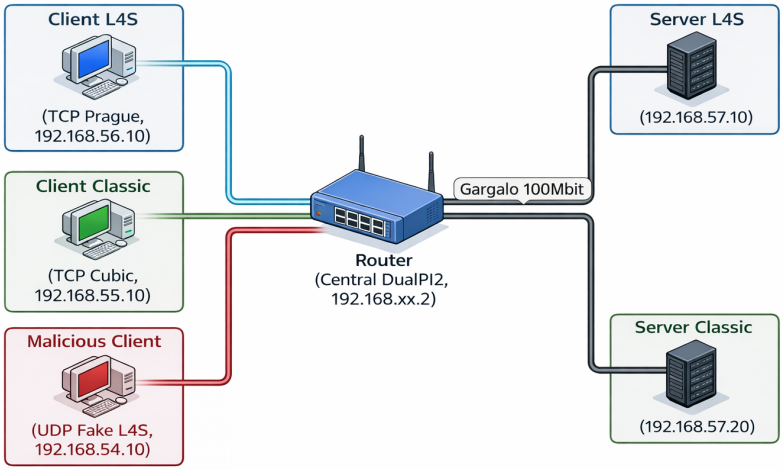


Figura 1. Topologia do ambiente experimental virtualizado, composta por dois emissores legítimos (L4S e Clássico), um emissor malicioso não-responsivo e um roteador central com DualPI2 no enlace de gargalo.

- **Atacante:** *host* malicioso que utiliza ECT(1) para ingressar na fila L, porém ignora os sinais de congestionamento e mantém taxa de envio agressiva.
- **Servidor:** *host* de destino responsável por receber o tráfego.
- **Roteador:** responsável por concentrar as medições no enlace de gargalo

Cada bateria de experimentos teve duração de 60 segundos, com geração de tráfego TCP por meio da ferramenta *iPerf*. Os logs foram coletados em intervalos de 1 segundo, o que permitiu observar tanto o comportamento transitório quanto o estado estacionário dos fluxos.

4.2. Cenários Avaliados

Dois cenários principais foram considerados. No cenário de *baseline*, os três emissores se comportam de forma cooperativa: um fluxo L4S legítimo e dois fluxos clássicos compartilham o enlace. Esse cenário estabelece a linha de base de coexistência e permite verificar se o *DualQ Coupled AQM* distribui a largura de banda de forma coerente [Schepper et al. 2023, Sarpkaya et al. 2024].

No cenário de ataque, o terceiro emissor altera seu comportamento para explorar o domínio L4S. Os pacotes passam a ser marcados com ECT(1), mas o emissor deixa de reagir ao *feedback* CE. O contraste entre os dois cenários permite mensurar o impacto do ataque tanto sobre os fluxos da fila L quanto sobre os fluxos da fila C.

4.3. Coleta de Dados e Geração do Conjunto Rotulado

Como não foi encontrado *dataset* público para ataques ECN não-responsivos especificamente no contexto do L4S, os dados foram gerados no próprio ambiente experimental. O processo de construção do conjunto rotulado compreendeu:

1. A execução de tráfego benigno, com coleta de logs associados ao cenário cooperativo;
2. A execução de tráfego sob ataque, com injeção do emissor não-responsivo;
3. A extração de atributos em janelas temporais de 1 segundo;
4. A rotulação dos fluxos como `BENIGNO` ou `MALICIOSO`.

Além dos logs de *iPerf*, o ambiente também considerou informações derivadas de captura de pacotes e de estatísticas de fila, de modo a enriquecer a caracterização comportamental dos fluxos observados.

5. Sistema de Detecção Baseado em Aprendizado Supervisionado

Esta seção descreve a construção do IDS supervisionado proposto, destacando a escolha do modelo, o processo de engenharia de atributos e o protocolo de treinamento e avaliação adotado para distinguir tráfego legítimo de comportamento malicioso no contexto L4S.

5.1. Escolha do modelo

A estratégia de detecção adotada foi baseada em aprendizado supervisionado, com uso de árvore de decisão. A escolha desse algoritmo se justifica por duas propriedades particularmente relevantes ao contexto do problema. A primeira é a interpretabilidade. O modelo produz regras legíveis, o que facilita auditoria, explicação e posterior integração com mecanismos de mitigação. A segunda é a baixa latência de inferência. Por se tratar de um classificador leve, sua execução no roteador se torna mais plausível sob restrições operacionais de tempo real.

5.2. Engenharia de atributos

A seleção dos atributos foi guiada pela hipótese de que fluxos ECN não-responsivos apresentam uma assinatura estatística distinta: elevada marcação CE, ausência de reação explícita por meio de CWR, manutenção de alto *throughput* e baixa variabilidade temporal. Com base nisso, foram considerados os seguintes atributos:

- `flow_throughput_bps`: taxa de transferência do fluxo em bits por segundo;
- `ratio_ect1`: proporção de pacotes marcados com ECT(1);
- `ratio_ce`: proporção de pacotes marcados com CE;
- `flag_cwr`: presença da sinalização *Congestion Window Reduced*;
- `ratio_cwr`: proporção de pacotes com CWR;
- `tcp_win_mean`: média da janela TCP anunciada;
- `iat_mean`: média do intervalo entre chegadas dos pacotes;
- `pkt_len_mean`: tamanho médio dos pacotes.

Essas variáveis foram calculadas a partir dos dados coletados no *testbed* e organizadas em estruturas tabulares para treinamento e teste do classificador.

5.3. Processo de Treinamento

O treinamento seguiu uma divisão entre dados de treinamento e teste, de forma a avaliar o desempenho do modelo em amostras não vistas previamente. Embora o foco deste trabalho não esteja na comparação exaustiva entre algoritmos, o processo adotado foi suficiente

para demonstrar a viabilidade do uso de classificadores supervisionados na identificação do ataque.

As regras aprendidas pelo modelo apontaram como fortemente indicativos de comportamento malicioso a combinação entre alta razão de pacotes CE, baixa incidência de CWR, *throughput* sustentado acima do esperado para fluxos cooperativos sob congestionamento e janelas TCP persistentemente elevadas. Em termos qualitativos, tais regras reproduzem a intuição do comportamento malicioso: o atacante aceita a priorização da fila L, mas rejeita o custo de reagir ao congestionamento.

5.4. Modelo de Detecção

O processo de treinamento do classificador foi estruturado a partir de um conjunto de dados rotulado, no qual cada fluxo de rede foi representado por oito atributos extraídos das medições experimentais e associado a uma classe binária, indicando comportamento benigno ou malicioso. Em seguida, os dados foram particionados em subconjuntos de treinamento e teste, permitindo a indução do modelo e sua avaliação inicial em dados não utilizados no ajuste dos parâmetros.

O Algoritmo 1 resume o procedimento adotado para construir o classificador supervisionado. Inicialmente, a matriz de entrada X é formada a partir das *features* extraídas dos fluxos de rede, enquanto o vetor y armazena os rótulos associados a cada instância, distinguindo tráfego benigno de tráfego malicioso. Na sequência, os dados são particionados em conjuntos de treinamento e teste, de modo a permitir uma avaliação preliminar da capacidade de generalização do modelo. Depois disso, uma Árvore de Decisão é instanciada com profundidade máxima igual a 5, escolha coerente com a necessidade de manter baixa complexidade computacional e elevada interpretabilidade. Por fim, o modelo é treinado com os dados de treinamento e sua acurácia é calculada sobre o conjunto de teste, produzindo uma medida objetiva de desempenho para a tarefa de detecção.

Algorithm 1 Treinamento do classificador de fluxos de rede

Require: Base de dados df, conjunto de atributos selecionados $features$
Ensure: Modelo treinado clf e acurácia $accuracy$

1: $X \leftarrow df[features]$
2: $y \leftarrow df[label\_is\_attack]$ ▷ 0=Benigno, 1=Malicioso
3: Dividir X e y em conjuntos de treinamento e teste, com 30% dos dados reservados para teste
4: Inicializar o classificador clf como uma Árvore de Decisão com profundidade máxima igual a 5
5: Treinar clf utilizando X_{train} e y_{train}
6: Calcular $accuracy$ a partir da predição do modelo sobre X_{test} em comparação com y_{test}
7: **return** clf, $accuracy$

6. Resultados e Discussão

Esta seção apresenta os resultados obtidos nos cenários de *baseline* e ataque, destacando os efeitos sobre *throughput*, justiça de banda e estabilidade do enlace. Em seguida, discute-se como esses achados sustentam a caracterização do ataque ECN não-responsivo no contexto L4S.

6.1. Comportamento no *Baseline*

No cenário cooperativo, ilustrado pelos resultados da Figura 2, os dois fluxos clássicos apresentaram comportamentos semelhantes, com *throughput* médio de 37.2 Mbits/s e 36.5 Mbits/s, respectivamente, o que indica boa simetria do ambiente. O fluxo L4S, por sua vez, apresentou *throughput* médio de 18.0 Mbits/s, com maior volatilidade temporal e episódios de ocupação rápida da banda quando os fluxos clássicos reduziam sua taxa.

Esse resultado não deve ser interpretado como simples desvantagem para o L4S, mas como consequência do próprio mecanismo de acoplamento e da prioridade dada à manutenção de baixa latência. Em outras palavras, o cenário *baseline* confirma que o ambiente experimental se comporta de forma coerente: há competição entre fluxos, utilização elevada do enlace e ausência de degradação na entrega dos fluxos.

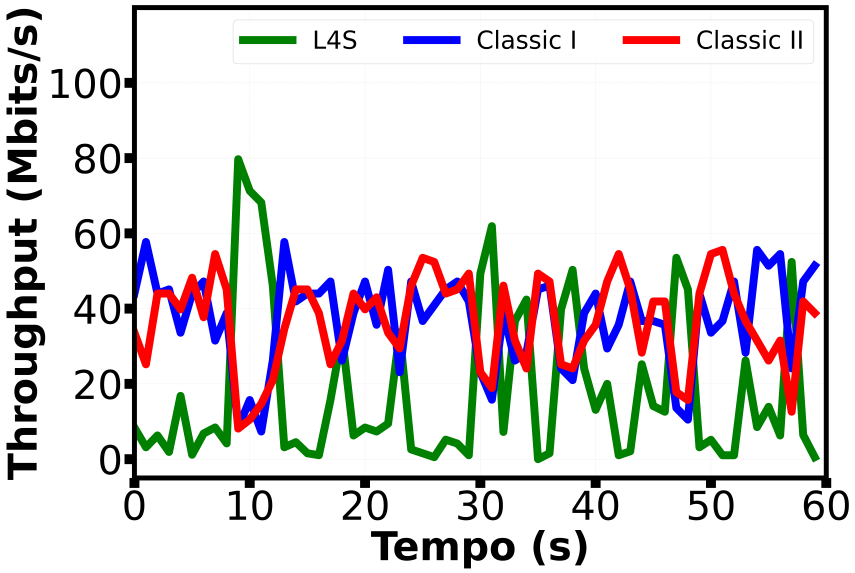


Figura 2. Comportamento temporal do *throughput* no cenário *baseline*, ilustrando os dois fluxos clássicos e o fluxo L4S ao longo de 60 s.

6.2. Execução do Ataque ECN Não-responsivo

Os resultados quantitativos da Figura 3 evidenciam uma inversão de justiça. Em vez de ser penalizado, o comportamento antissocial é premiado com maior alocação de banda. Por outro lado, os fluxos cooperativos são punidos justamente por obedecerem aos sinais do AQM. O resultado é um caso emblemático de falha em controle distribuído, no qual a validade do mecanismo depende do cumprimento voluntário das regras por parte dos sistemas finais.

No cenário de ataque, o emissor malicioso passou a monopolizar o enlace. Seu *throughput* médio aumentou de 36.5 Mbits/s no *baseline* para 76.5 Mbits/s, capturando a maior parte da capacidade disponível. Em termos práticos, o comportamento do atacante se traduziu em um platô de alta vazão ao longo dos 60 segundos de experimento, com baixa oscilação e ausência dos recuos característicos de um fluxo que responde ao congestionamento.

O efeito sobre os fluxos legítimos foi severo. A vítima L4S teve seu *throughput* reduzido de 18.0 Mbits/s para apenas 0.106 Mbits/s, o que representa uma queda de apro-

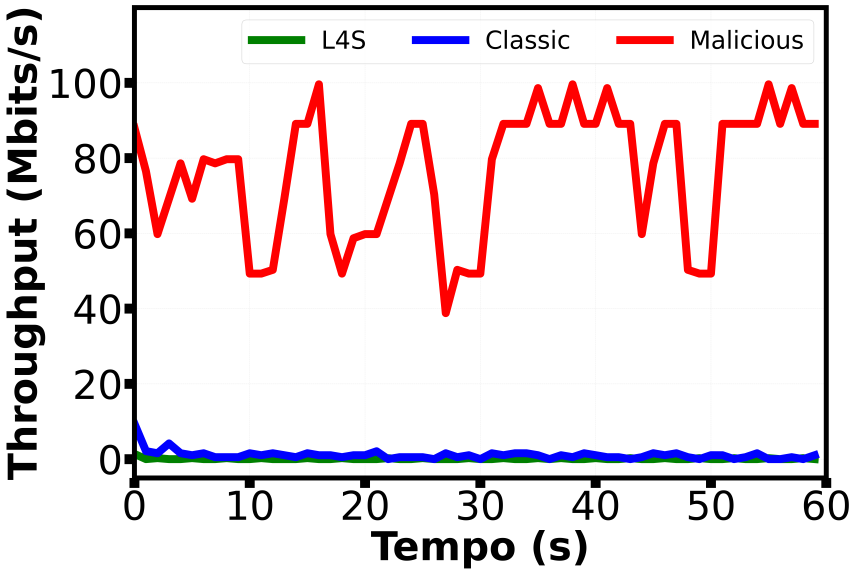


Figura 3. Comportamento temporal do *throughput* no cenário de ataque,com o atacante dominando a banda e os fluxos legítimos comprimidos junto ao eixo horizontal.

ximadamente 99.4%. Já a vítima clássica foi reduzida de 37.2 Mbits/s para 1.08 Mbits/s, correspondendo a uma perda de cerca de 97.1%. Esses valores mostram que o ataque não compromete apenas a fila L, mas também degrada fortemente o domínio clássico por meio do acoplamento probabilístico do AQM. A Tabela 1 sintetiza essa comparação entre os dois cenários.

Tabela 1. Resumo comparativo do *throughput* médio nos cenários *baseline* e ataque.

Fluxo	Baseline	Ataque	Variação Absoluta	Variação %
Atacante	36.5	76.5	+40.0	+109.6%
Vítima L4S	18.0	0.106	-17.9	-99.4%
Vítima Clássica	37.2	1.08	-36.1	-97.1%

6.3. Interpretação dos Resultados

Os resultados sugerem que a implementação padrão do DualQ, embora eficiente sob hipóteses de cooperação, não dispõe de mecanismos intrínsecos para verificar a responsividade real do emissor que utiliza ECT(1). Em outras palavras, a rede consegue observar a marcação no cabeçalho, mas não garante que o remetente esteja semanticamente aderente ao comportamento implícito do L4S [Briscoe et al. 2023b, Schepper et al. 2023].

Essa constatação tem implicações diretas para a segurança de arquiteturas de baixa latência. Em ambientes abertos ou multiusuário, a possibilidade de um único emissor malicioso degradar drasticamente o desempenho de fluxos legítimos torna o problema relevante não apenas do ponto de vista de desempenho, mas também de disponibilidade de serviço.

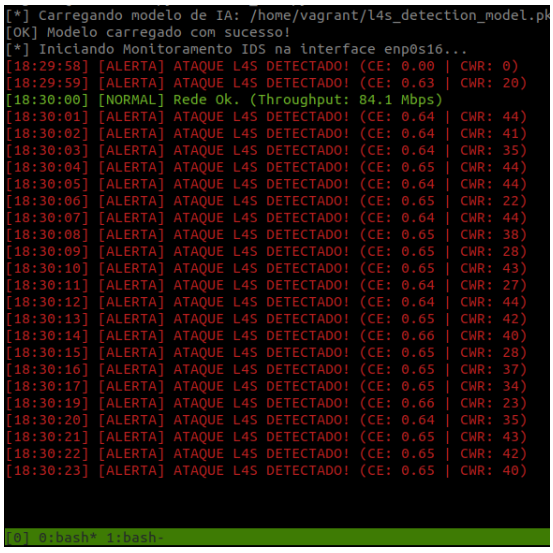
7. Validação do IDS

Esta seção valida o sistema de detecção proposto, analisando se os padrões observados permitem distinguir, de forma consistente, tráfego legítimo de comportamento malicioso em ambientes L4S. Além de avaliar o desempenho do classificador supervisionado, a seção discute as regras inferidas pelo modelo e suas implicações para o monitoramento de redes de baixa latência no contexto de ataques ECN não responsivos.

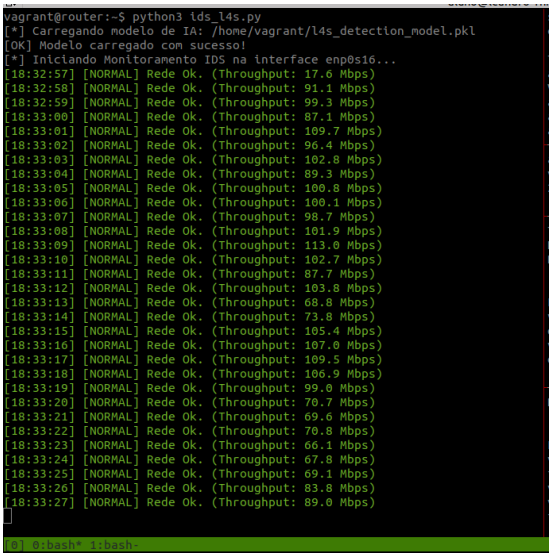
7.1. Assinatura Comportamental do Ataque

O IDS executado no roteador central foi capaz de associar três indícios principais ao comportamento malicioso: *(i)* alta taxa de marcação CE na fila L, *(ii)* baixa variação temporal do *throughput* do fluxo suspeito e *(iii)* correlação entre persistência do fluxo invasor e colapso dos demais emissores. Em conjunto, esses sinais compõem uma assinatura consistente do ataque ECN não-responsivo.

As regras extraídas pelo classificador reforçam essa interpretação. Em particular, a combinação entre `ratio_ce` elevado, `ratio_cwr` reduzido, *throughput* sustentado e janela TCP média alta descreve exatamente o padrão esperado de um fluxo que recebe feedback de congestionamento, mas decide ignorá-lo. A Figura 4 apresenta evidências visuais desse comportamento ao longo da execução do experimento.



(a) Alertas iniciais durante o ataque.



(b) Persistência de alertas e classificação.

Figura 4. Evidências visuais da detecção em tempo real do ataque ECN não-responsivo pelo IDS, detalhadas em dois subgráficos: (a) alertas iniciais e (b) persistência dos alertas ao longo da execução.

Após o treinamento, a árvore de decisão permitiu extrair regras interpretáveis que descrevem padrões comportamentais associados à identificação de fluxos maliciosos. O Algoritmo 2 resume a lógica de decisão aprendida pelo modelo.

O Algoritmo 2 representa, em forma estruturada, a assinatura comportamental aprendida pelo modelo de classificação. A primeira regra identifica como malicioso o fluxo que apresenta alta proporção de marcações CE, baixa taxa de resposta com CWR, vazão elevada e janela TCP média alta, o que sugere um comportamento agressivo diante

Algorithm 2 Regras de classificação extraídas da árvore de decisão

Require: Valores observados para $ratio\_ce$, $ratio\_cwr$, $flow\_throughput\_bps$, tcp_win_mean, $ratio\_ect1$, iat_mean e $flag\_cwr$
Ensure: Classe do fluxo: *Malicioso* ou *Benigno*

if $ratio\_ce > 0.80$ **then**
  if $ratio\_cwr < 0.10$ **then**
    if $flow\_throughput\_bps > 70$ Mbps **then**
      if $tcp\_win\_mean > 150000$ **then**
        return *Malicioso* ▷ confiança de 95%
      end if
    end if
  end if
else if $ratio\_ect1 > 0.95$ **then**
  if $iat\_mean < 0.001$ **then**
    if $flag\_cwr = 0$ **then**
      return *Malicioso* ▷ confiança de 92%
    end if
  end if
end if
return *Benigno*

dos sinais de congestionamento. A segunda regra complementa essa análise ao classificar como malicioso o fluxo com predominância de marcação ECT(1), intervalo médio entre pacotes muito pequeno e ausência do sinalizador CWR, indicando baixa responsividade ao mecanismo de controle de congestionamento. Caso nenhuma dessas condições seja satisfeita, o fluxo é classificado como benigno. Dessa forma, o pseudo-código evidencia que o modelo não apenas realiza a classificação, mas também fornece regras interpretáveis que ajudam a compreender quais combinações de atributos são mais relevantes para distinguir tráfego legítimo de tráfego suspeito.

7.2. Implicações práticas

A detecção correta do ataque não resolve, por si só, o problema de disponibilidade do enlace, mas demonstra que o comportamento malicioso deixa uma pegada estatística suficientemente nítida para ser reconhecida por mecanismos supervisionados leves. Isso abre espaço para evoluções naturais do sistema, como rebaixamento de prioridade, *rate limiting*, quarentena de fluxos suspeitos ou integração com políticas de policiamento por fluxo.

Além disso, a interpretabilidade da árvore de decisão favorece sua adoção em cenários de rede programável e depuração operacional, nos quais a capacidade de explicar o motivo de uma classificação é tão importante quanto o próprio acerto da detecção.

8. Limitações e Trabalhos Futuros

Apesar dos resultados promissores, algumas limitações devem ser destacadas. Em primeiro lugar, o estudo concentrou-se em um conjunto específico de experimentos, com

topologia controlada e tráfego sintético. Embora esse desenho seja adequado para isolar o fenômeno investigado, ele não esgota a variabilidade de redes reais com múltiplas aplicações concorrentes.

Em segundo lugar, o trabalho enfatizou a detecção, e não a mitigação. Assim, permanece em aberto a integração do classificador a mecanismos reativos que possam conter o ataque sem produzir penalizações indevidas sobre tráfego legítimo. Em terceiro lugar, o uso de árvore de decisão foi motivado por interpretabilidade e eficiência, mas outros algoritmos, como *Random Forest*, *Gradient Boosting* ou redes neurais leves, podem ser explorados em avaliações futuras.

Por fim, ataques adaptativos capazes de simular maior variabilidade temporal ou padrões mais próximos do comportamento benigno exigirão ciclos contínuos de reavaliação e refinamento do conjunto de atributos empregado.

9. Conclusão

Este artigo apresentta a fundamentação e os resultados de um estudo sobre ataques ECN não-responsivos em arquiteturas L4S. A análise mostrou que a exploração do bit ECT(1) sem adesão à semântica de controle de congestionamento compromete a operação do DualQ e provoca uma degradação extrema sobre fluxos legítimos L4S e Clássicos. Os resultados experimentais evidenciaram que um único emissor malicioso é suficiente para capturar a maior parte da largura de banda do enlace de gargalo, reduzindo em mais de 99% o *throughput* da vítima L4S e em mais de 97% o da vítima clássica. Tal comportamento confirma que a ausência de verificação ativa da responsividade do emissor representa uma vulnerabilidade crítica para a adoção segura do L4S.

Adicionalmente, o estudo indicou que um IDS supervisionado baseado em árvore de decisão pode identificar o ataque a partir de uma assinatura comportamental clara, combinando marcação CE elevada, baixa reação via CWR e manutenção de alta taxa de transmissão. Como desdobramento, o trabalho destaca a necessidade de incorporar mecanismos inteligentes de segurança ao plano de dados ou ao roteador de borda em ambientes L4S, de modo a preservar não apenas a baixa latência prometida pela arquitetura, mas também a justiça e a disponibilidade do serviço.

Agradecimentos

Os autores agradecem à Chamada Interconecta IFPB para apoio a projetos de pesquisa, inovação, desenvolvimento tecnológico e social da Pró-Reitoria de Pesquisa, Inovação e Pós-Graduação (PRPIPG) do Instituto Federal da Paraíba (IFPB), pelo apoio institucional e pelo fomento que viabilizaram o desenvolvimento desta pesquisa.

Referências

Amor, N. B., Benferhat, S., and Elouedi, Z. (2004). Naive Bayes vs decision trees in intrusion detection systems. In *Proceedings of the 2004 ACM Symposium on Applied Computing*, SAC '04, page 420–424, New York, NY, USA. Association for Computing Machinery.

Baker, F. and White, G. (2015). IETF Recommendations Regarding Active Queue Management. RFC 7567.

Bishop, C. (2006). *Pattern recognition and machine learning*, volume 4. Springer New York.

Black, D. L. (2018). Relaxing Restrictions on Explicit Congestion Notification (ECN) Experimentation. RFC 8311.

Briscoe, B., Schepper, K. D., Bagnulo, M., and White, G. (2023a). Low Latency, Low Loss, and Scalable Throughput (L4S) Internet Service: Architecture. RFC 9330.

Briscoe, B., Schepper, K. D., Kuehlewind, M., and White, G. (2023b). The Explicit Congestion Notification (ECN) Protocol for Low Latency, Low Loss, and Scalable Throughput (L4S). RFC 9331.

Messaoudi, S., Ksentini, A., Messaoudi, F., and Bonnet, C. (2024). SDN-based L4S Congestion Control in Beyond 5G. In *IEEE International Conference on High Performance Switching and Routing (HPSR)*, pages 99–105.

Monteiro, L. V., Simão, V. S., de B. Lira, R., de Almeida, L. C., Gomes, R. D., and Maciel, P. D. (2024). L4S in Private 5G Industrial Networks: A Case Study for Real-Time Video Transmission in Programmable Networks. In *IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, pages 1–4.

Nichols, K. and Jacobson, V. (2018). Controlled Delay Active Queue Management. RFC 8289.

Oljira, D. B., Grinnemo, K.-J., Brunstrom, A., and Taheri, J. (2020). Validating the Sharing Behavior and Latency Characteristics of the L4S Architecture. *ACM SIGCOMM Computer Communication Review*, 50(2):37–44.

Ramakrishnan, K., Floyd, S., and Black, D. (2001). The Addition of Explicit Congestion Notification (ECN) to IP. RFC 3168.

Sarpkaya, F. B., Srivastava, A., Fund, F., and Panwar, S. (2024). To switch or not to switch to TCP Prague? Incentives for adoption in a partial L4S deployment. In *Proceedings of the 2024 Applied Networking Research Workshop*, ANRW '24, page 45–52, New York, NY, USA. Association for Computing Machinery.

Schepper, K. D., Briscoe, B., and White, G. (2023). Dual-Queue Coupled Active Queue Management (AQM) for Low Latency, Low Loss, and Scalable Throughput (L4S). RFC 9332.