

Análise de Viabilidade do Algoritmo 2SYN Baseada em uma Implementação OpenWrt

Diogo K. C. Yonemoto, Vinícius C. Silva, Paulo R. S. L. Coelho,
Luis F. Faina, Rafael Pasquini

Faculdade de Computação (FACOM)
Universidade Federal de Uberlândia (UFU) – Uberlândia – MG – Brazil

{diogo.yonemoto, vinicius.costa.silva, paulocoelho}@ufu.br

{luisfaina, rafael.pasquini}@ufu.br

Abstract. *Multihoming improves availability, but solutions such as SD-WAN introduce complexity and require control over both endpoints. In practice, most Internet traffic relies on TCP over IPv4 with NAT, where transparent path selection for arbitrary destinations remains a challenge. The 2SYN algorithm leverages the TCP handshake to select paths based on the initial RTT without requiring support from the destination. This work investigates the feasibility of implementing 2SYN in embedded environments using OpenWrt. Results demonstrate correct dynamic path selection in multihoming scenarios.*

Resumo. *O multihoming melhora a disponibilidade de redes, porém soluções como SD-WAN apresentam maior complexidade e dependem do controle dos extremos. Na prática, grande parte do tráfego utiliza TCP sobre IPv4 com NAT, onde a seleção transparente de caminhos para destinos arbitrários permanece um desafio. O algoritmo 2SYN utiliza o handshake TCP para selecionar dinamicamente o caminho com menor RTT inicial, sem suporte no destino. Este trabalho investiga sua viabilidade em ambientes embarcados por meio de uma implementação em OpenWrt. Resultados demonstram a seleção dinâmica de caminhos em cenários multihoming.*

1. Introdução

O uso de *multihoming* tem sido amplamente adotado como estratégia para aumentar a disponibilidade e o desempenho de redes, permitindo o uso de múltiplos enlaces de acesso. Abordagens tradicionais, como *failover* e balanceamento de carga estático [Bonald and Roberts 2012], não consideram as condições dinâmicas da rede, podendo resultar em escolhas subótimas de caminho.

Soluções baseadas em SD-WAN (*Software-Defined - Wide Area Network*) [Yalda et al. 2021, Yang et al. 2019] possibilitam o gerenciamento dinâmico de tráfego por meio de monitoramento contínuo e controle centralizado. No entanto, essas abordagens geralmente exigem maior complexidade de implantação, frequentemente baseadas em túneis entre pontos controlados, o que limita sua aplicação a cenários onde há domínio sobre os extremos da comunicação.

Na prática, grande parte do tráfego na Internet ainda utiliza TCP sobre IPv4 com NAT (*Network Address Translation*), onde os destinos são arbitrários e não há suporte

a mecanismos cooperativos de seleção de caminhos. Nesse contexto, torna-se relevante investigar soluções leves e transparentes que possam operar sem dependência do destino.

O algoritmo 2SYN [Toledo and Keslassy 2025] propõe uma alternativa nesse sentido, utilizando o próprio estabelecimento de conexões TCP para inferir dinamicamente o melhor caminho disponível. A abordagem consiste em enviar pacotes SYN por múltiplos enlaces e selecionar aquele cujo SYN-ACK é recebido primeiro, utilizando o RTT (*Round Trip Time*) inicial como critério de decisão, sem necessidade de suporte no destino.

Apesar de sua simplicidade e potencial prático, a implementação do 2SYN em ambientes embarcados ainda é pouco explorada, especialmente considerando restrições de *hardware* e integração com o plano de encaminhamento de pacotes.

Neste contexto, este trabalho investiga a viabilidade da implementação do algoritmo 2SYN em dispositivos embarcados, por meio de uma solução baseada em OpenWrt [OpenWrt Project 2024]. A proposta busca demonstrar que é possível integrar o algoritmo ao processamento de pacotes de forma transparente, permitindo a seleção dinâmica de caminhos em cenários multihoming reais.

2. Trabalhos Relacionados

Soluções para *multihoming* têm sido amplamente estudadas, com diferentes abordagens para seleção de caminhos. Arquiteturas baseadas em SD-WAN [Yalda et al. 2021, Yang et al. 2019] realizam seleção dinâmica a partir de métricas como latência e perda de pacotes, geralmente por meio de túneis e controle centralizado, sendo mais adequadas a cenários com domínio sobre os extremos da comunicação.

Outra linha de trabalho envolve protocolos *multipath*, como o MPTCP (*Multipath TCP*) [Ford et al. 2013], que permitem o uso simultâneo de múltiplos caminhos em uma mesma conexão. Essas abordagens, entretanto, requerem suporte no destino e modificações na pilha de protocolos, o que limita sua aplicação na Internet aberta.

O algoritmo 2SYN [Toledo and Keslassy 2025] propõe uma alternativa leve ao utilizar o estabelecimento de conexões TCP para inferir o melhor caminho com base no RTT inicial, sem necessidade de suporte no destino. No entanto, sua implementação em ambientes embarcados ainda é pouco explorada. Este trabalho diferencia-se ao investigar a viabilidade do 2SYN em dispositivos baseados em OpenWrt, com foco na integração ao processamento de pacotes em cenários reais de *multihoming*.

3. Arquitetura da Proposta

Esta seção descreve a arquitetura para implementação do algoritmo 2SYN em um roteador baseado em OpenWrt, com foco na integração ao processamento de pacotes e viabilidade em sistemas embarcados. A Figura 1 descreve o seu funcionamento, passo a passo, bem como traz detalhes experimentais a serem discutidos, posteriormente, junto da Figura 2.

A proposta considera um roteador atuando como *gateway*, conectado a múltiplos enlaces de saída. Para cada nova conexão TCP, o roteador intercepta pacotes SYN, replica-os por diferentes *interfaces* e seleciona o caminho com base no primeiro SYN-ACK recebido. A solução opera de forma transparente para aplicações e não requer suporte no destino, utilizando exclusivamente informações do protocolo TCP.

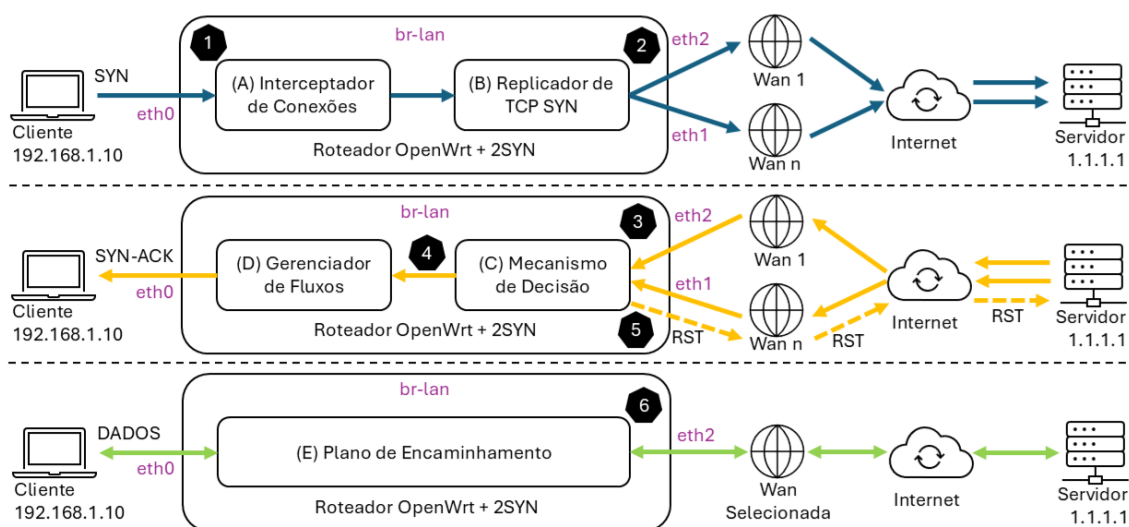


Figura 1. Etapas de operação do algoritmo 2SYN em ambiente OpenWrt: (1–2) interceptação e replicação do pacote SYN por múltiplos enlaces; (3–5) recepção dos pacotes SYN-ACK, seleção do caminho com menor RTT inicial e encerramento dos demais caminhos via RST; (6) encaminhamento bidirecional dos dados pelo enlace selecionado.

A arquitetura é composta pelos seguintes elementos:

(A) Interceptador de Conexões: responsável por identificar novos fluxos TCP a partir da captura de pacotes SYN no plano de dados, utilizando mecanismos do sistema, como *iptables/Netfilter*.

(B) Replicador de TCP SYN: realiza a duplicação do pacote SYN inicial e seu envio por múltiplas interfaces de saída, explorando os enlaces de *multihoming*.

(C) Mecanismo de Decisão: determina o enlace a ser utilizado com base no primeiro pacote SYN-ACK recebido, utilizando o RTT inicial como critério de seleção, e realiza o encerramento das demais tentativas por meio do envio de pacotes RST.

(D) Gerenciador de Fluxos: mantém o estado das conexões ativas e associa ao enlace selecionado, garantindo consistência no encaminhamento dos pacotes.

(E) Plano de Encaminhamento: responsável pelo envio dos pacotes de dados pelo enlace selecionado, delegando o encaminhamento ao sistema operacional após a definição do caminho.

O funcionamento do sistema ocorre da seguinte forma:

1. Interceptação de um novo pacote SYN;
2. Replicação do SYN pelas *interfaces* disponíveis;
3. Recebimento dos pacotes SYN-ACK;
4. Seleção do enlace associado ao primeiro retorno;
5. Encerramento dos demais caminhos via RST;
6. Encaminhamento do fluxo pelo enlace selecionado.

A implementação foi realizada em OpenWrt [OpenWrt Project 2024], utilizando mecanismos nativos do Linux para manipulação de pacotes. A versão final utiliza componentes em linguagem C baseados em *Netfilter* para interceptação e controle de fluxos,

#	Timestamp	Interface	Direção	IP de Origem	IP de Destino	Flags TCP	Ação	Etapa
1	20:25:33.985945	eth0	In	192.168.1.10.52752	1.1.1.1.80	[S]	SYN original	1
2	20:25:33.985953	br-lan	In	192.168.1.10.52752	1.1.1.1.80	[S]	SYN capturado na bridge (br-lan)	
3	20:25:33.986389	eth2	Out	192.168.150.167.52752	1.1.1.1.80	[S]	SYN replicado - enlace rápido (eth2)	2
4	20:25:33.999791	eth2	In	1.1.1.1.80	192.168.150.167.52752	[S.]	retorno 1º SYN ACK - eth2 vencedora!	3
5	20:25:34.000519	br-lan	Out	1.1.1.1.80	192.168.1.10.52752	[S.]	SYN ACK capturado na bridge (br-lan)	4
6	20:25:34.000530	eth0	Out	1.1.1.1.80	192.168.1.10.52752	[S.]	SYN ACK entregue ao nó de origem	
7	20:25:34.006343	eth0	In	192.168.1.10.52752	1.1.1.1.80	[.]	ACK do nó de origem ao servidor (three way handshake pela eth2)	6
8	20:25:34.006352	br-lan	In	192.168.1.10.52752	1.1.1.1.80	[.]		
9	20:25:34.006425	eth2	Out	192.168.150.167.52752	1.1.1.1.80	[.]		
10	20:25:34.006481	eth0	In	192.168.1.10.52752	1.1.1.1.80	[P.]	HTTP: GET / HTTP/1.1	
11	20:25:34.006486	br-lan	In	192.168.1.10.52752	1.1.1.1.80	[P.]		
12	20:25:34.006519	eth2	Out	192.168.150.167.52752	1.1.1.1.80	[P.]		
13	20:25:34.018970	eth2	In	1.1.1.1.80	192.168.150.167.52752	[.]	ACK (servidor > nó de origem)	
14	20:25:34.019047	br-lan	Out	1.1.1.1.80	192.168.1.10.52752	[.]		
15	20:25:34.019057	eth0	Out	1.1.1.1.80	192.168.1.10.52752	[.]		
16	20:25:34.019333	eth2	In	1.1.1.1.80	192.168.150.167.52752	[P.]	HTTP: HTTP/1.1 301 Moved Permanently	
17	20:25:34.019375	br-lan	Out	1.1.1.1.80	192.168.1.10.52752	[P.]		
18	20:25:34.019380	eth0	Out	1.1.1.1.80	192.168.1.10.52752	[P.]		
19	20:25:34.022390	eth0	In	192.168.1.10.52752	1.1.1.1.80	[.]	ACK (nó de origem > servidor)	
20	20:25:34.022397	br-lan	In	192.168.1.10.52752	1.1.1.1.80	[.]		
21	20:25:34.022464	eth2	Out	192.168.150.167.52752	1.1.1.1.80	[.]		
[22 .. 30] Fechamento da conexão TCP do enlace vencedor (eth2) no período entre [20:25:34.024053 .. 20:25:34.038158]- pacotes omitidos em virtude do espaço total neste artigo								
31	20:25:34.086351	eth1	Out	192.168.122.181.52752	1.1.1.1.80	[S]	SYN replicado - enlace atraso 100ms (eth1)	2
32	20:25:34.096439	eth1	In	1.1.1.1.80	192.168.122.181.52752	[S.]	retorno 2º SYN ACK - eth1 perdedora!	3
33	20:25:34.100089	eth1	Out	192.168.122.181.52752	1.1.1.1.80	[R.]	RST conexão perdedora (eth1)	5

Figura 2. Captura *tcpdump* ilustrando as fases de operação do algoritmo 2SYN em ambiente OpenWrt entre o host local 192.168.1.10 e o servidor 1.1.1.1.

permitindo maior eficiência na fase inicial da conexão. Após a seleção do caminho, o encaminhamento dos pacotes é delegado ao sistema operacional. Uma versão inicial em Python foi utilizada para validação funcional da lógica do algoritmo.

A arquitetura proposta permite a integração do 2SYN diretamente no plano de dados do roteador, mantendo baixo acoplamento com aplicações e transparência para o usuário final. Além disso, a abordagem limita o processamento adicional à fase inicial da conexão, reduzindo o impacto no desempenho do sistema.

4. Implementação e Aspectos de Sistema

A implementação foi desenvolvida em linguagem C, explorando mecanismos *Netfilter* no *kernel* Linux para interceptação, processamento e encaminhamento de pacotes. A abordagem concentra o processamento na fase inicial do estabelecimento da conexão TCP, permitindo que o restante do fluxo seja tratado diretamente pelo sistema operacional. A Figura 2 apresenta uma captura com *tcpdump* que ilustra as fases de operação da solução implementada no OpenWrt. Os detalhes das *interfaces*, *bridge*, cliente e servidor foram apresentados na Figura 1, como forma de ajudar na interpretação dos resultados.

A interceptação dos pacotes é realizada por meio de regras *iptables*, que direcionam pacotes SYN para uma fila controlada pela aplicação utilizando a biblioteca *libnetfilter_queue*. Os pacotes capturados (linhas 1 e 2) permanecem na fila até serem processados, permitindo controle explícito sobre seu tratamento antes de qualquer encaminhamento. Ao identificar um novo fluxo TCP, o sistema realiza a replicação do pacote SYN e o envia por múltiplas *interfaces* de saída (linhas 3 e 31), explorando os enlaces disponíveis no cenário de *multihoming*. As respostas são monitoradas, e o primeiro pacote SYN-ACK (linha 4) recebido é utilizado como critério para seleção do enlace (linhas 5 e 6), com base no RTT inicial observado. Após a seleção do caminho, o sistema promove o encerramento das demais tentativas de conexão por meio do envio de pacotes RST (linhas 32 e 33), garantindo que apenas o fluxo correspondente ao enlace selecionado permaneça

ativo (linhas 7 a 30).

O gerenciamento das conexões é realizado com o apoio da biblioteca *libnetfilter_conntrack*, permitindo a criação e manutenção de entradas associadas ao fluxo selecionado. Conforme observado na Figura 2 a rota pela *interface* `eth2` foi a mais rápida, sendo utilizada para a troca efetiva de dados entre Cliente e Servidor. A rota pela `eth1` foi configurada com um atraso de 100ms para fins de experimentação, sendo a perdedora.

5. Conclusão

Este trabalho apresentou a implementação do algoritmo 2SYN em ambiente OpenWrt, com foco na viabilidade de sua utilização em dispositivos embarcados para seleção dinâmica de caminhos em cenários *multihoming*. A abordagem proposta explora o próprio estabelecimento de conexões TCP para inferir o melhor enlace disponível, dispensando suporte no destino e mecanismos adicionais de controle.

Como principal contribuição, foi desenvolvida uma implementação eficiente em linguagem C, baseada no *framework Netfilter*, permitindo a integração do algoritmo ao processamento de pacotes no sistema operacional. A solução demonstrou ser capaz de operar de forma transparente às aplicações, limitando o processamento à fase inicial da conexão e delegando o encaminhamento subsequente ao plano de dados do sistema.

Os resultados obtidos indicam que a abordagem é viável e apresenta potencial como alternativa leve às soluções tradicionais, especialmente em cenários com restrições de custo e complexidade. Como trabalhos futuros, pretende-se avaliar o desempenho da solução em cenários mais amplos, incluindo medições quantitativas, validação em *hardware* embarcado dedicado e a extensão da abordagem para suportar outros protocolos e ambientes, como QUIC e IPv6.

Agradecimentos

Este trabalho contou com o apoio da Universidade Federal de Uberlândia (UFU) e da Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG).

Referências

- Bonald, T. and Roberts, J. (2012). Internet and the erlang formula. *ACM SIGCOMM*.
- Ford, A., Raiciu, C., Handley, M., and Bonaventure, O. (2013). TCP extensions for multipath operation with multiple addresses. *RFC 6824*.
- OpenWrt Project (2024). OpenWrt: A Linux operating system targeting embedded devices. <https://openwrt.org>.
- Toledo, K. and Keslassy, I. (2025). 2SYN: Congestion-aware multihoming. In *IEEE/IFIP Network Operations and Management Symposium (NOMS)*.
- Yalda, K. G., Hamad, D. J., and Tapus, N. (2021). A survey on Software-Defined Wide Area Network (SD-WAN) architectures. *IEEE Access*.
- Yang, Z., Cui, Y., Li, B., Liu, Y., and Xu, Y. (2019). Software-Defined Wide Area Network (SD-WAN): Architecture, advances and opportunities. *IEEE Communications Surveys & Tutorials*.