BGP daemons

A Fun Battle

Here is where your peering begins...

By Claus Töpke - claus.topke@gmail.com - www.telcomanager.com

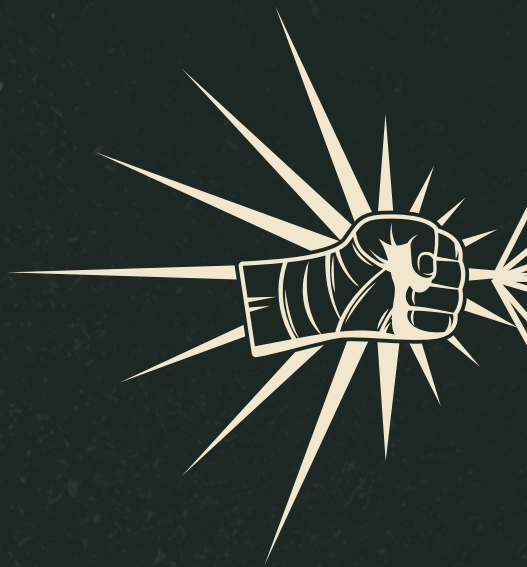
Who am I ?

20+ years experience
Network Operator and Engineer
Network Architect
Software Developer
Product Developer & Language explorer
Technology Writer
Network Automation

Today working as Product Director for IPs @ Telcomanager

Why selecting a BGP daemon matter ?

Use cases



SDN Controller & TE
Optimizing CDN traffic with peers. Allowing prioritization for VIP customers.

DDoS mitigation
With flowspec, among particular communities mitigate or limit large scale attacks

Looking Glass
Allow customers to check local BGP tables before announcing

Router Reflector
Remove load on border and PE routers

Border Router
Open source Linux routers

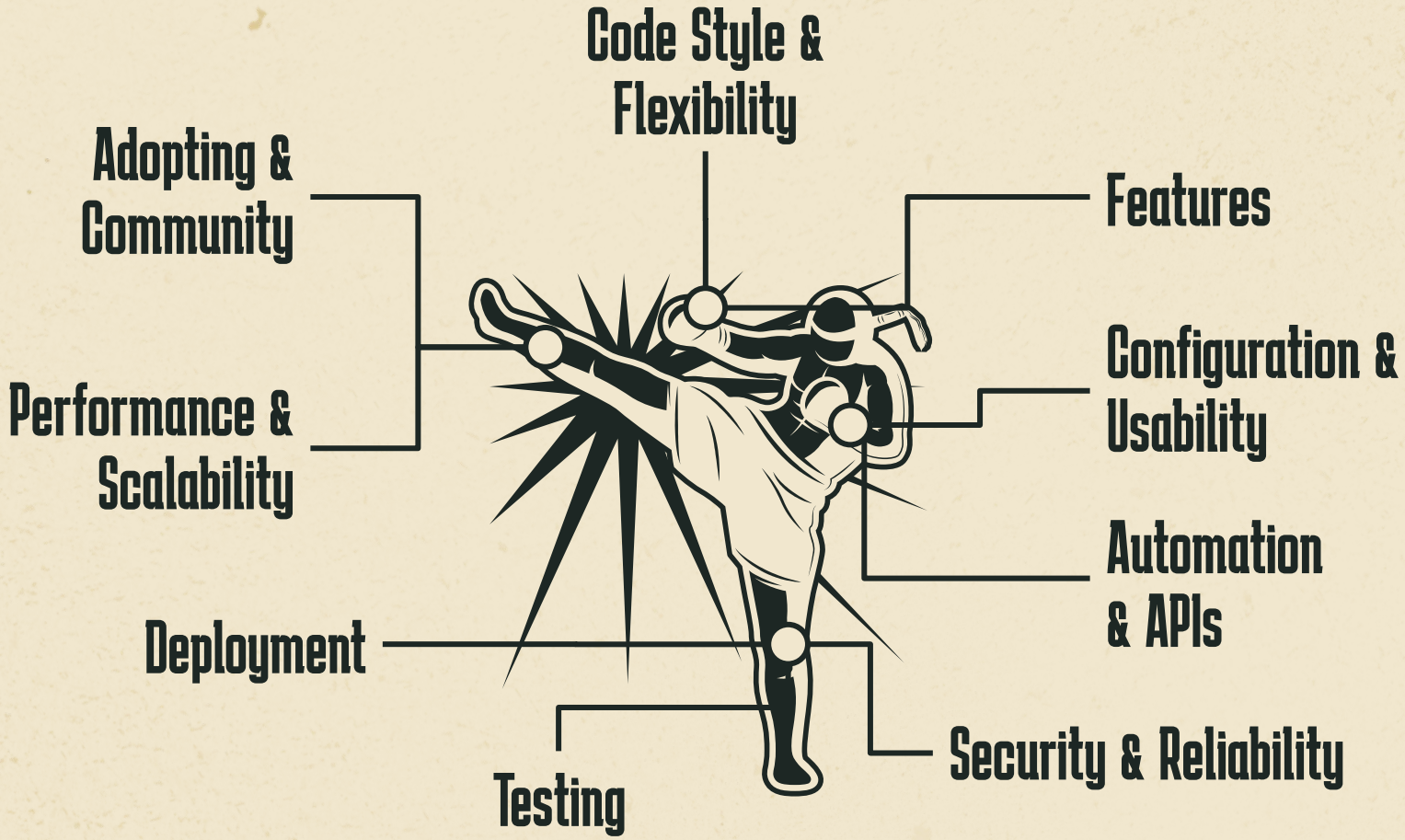
Data Fabric
Use EVPN to create large clos network data centers and routers

BGP Analyser
Evaluate and monitor peers, create reports of traffic exchange

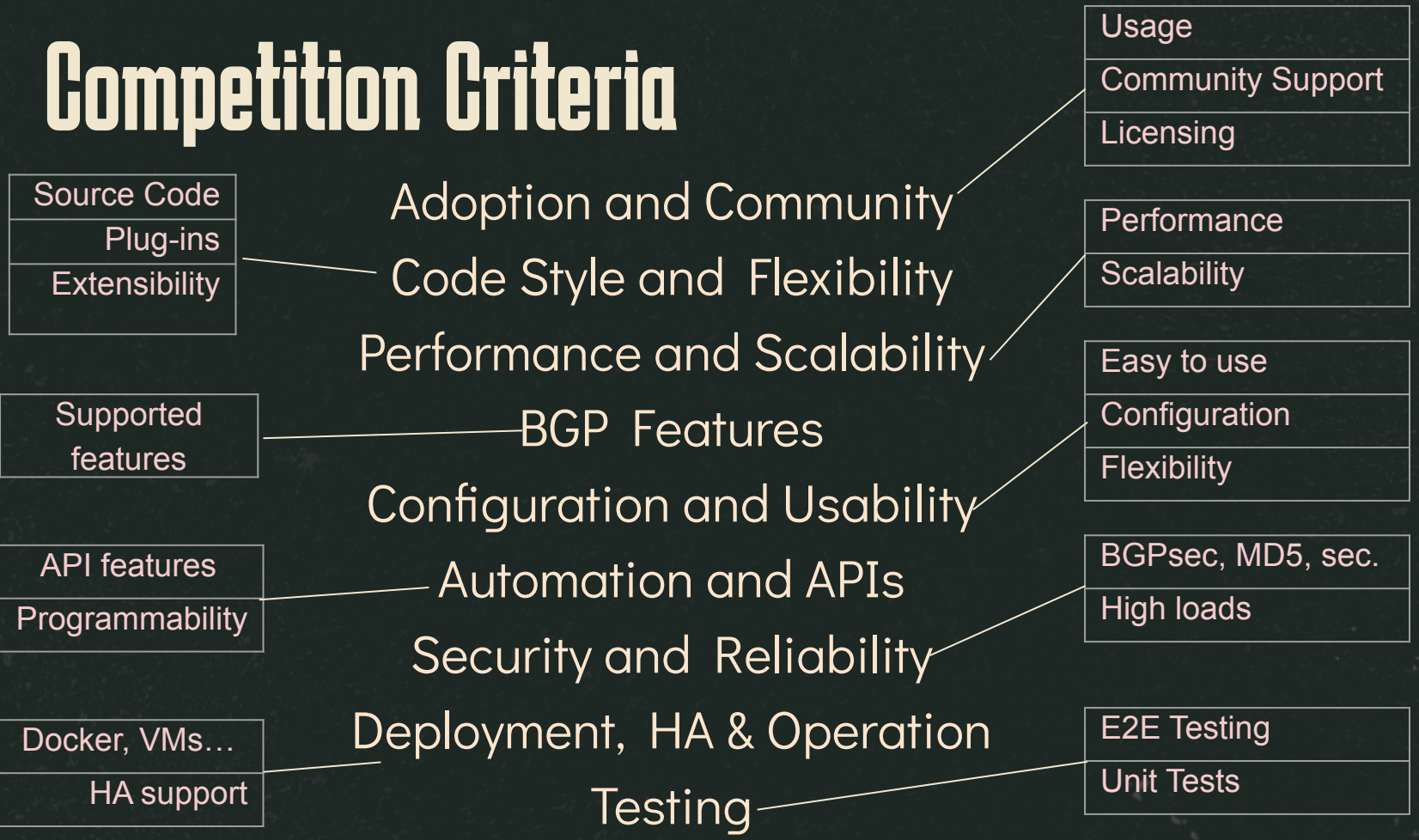
Why did the BGP router start therapy?

It was dealing with some serious route withdrawal issues.

Competition Criteria


Code Style &
Flexibility
Adopting &
Community
Features
Performance &
Scalability
Configuration &
Usability
Automation
& APIs
Deployment
Security & Reliability
Testing

Competition Criteria



Contenders

Quagga (Zebra)

BIRD

Bird

FRRouting (FRR)

goBGP

ExaBGP

exa networks

RustyBGP

holo-routing

OpenBGPD

GO AHEAD. EXPLORE.

OpenBGPD

iBGP

Scoring

Score	Quagga	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
Adoption & Community								
Code & Flexibility								
Configuration & Usability								
Automation & API								
Security & Reliability								

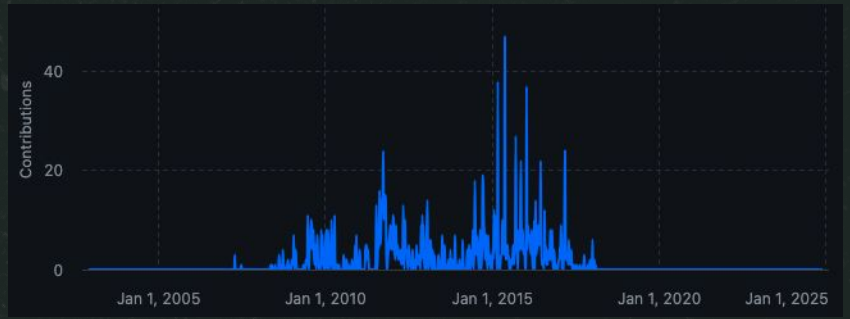
Score	Quagga	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
BGP Features								
Deployment, HA & Oper								
Performance & Scalability								

Adoption & Community Code & Flexibility

Quagga (Zebra)

JP 1996 as zebra and 2002 as Quagga then FRR 2017

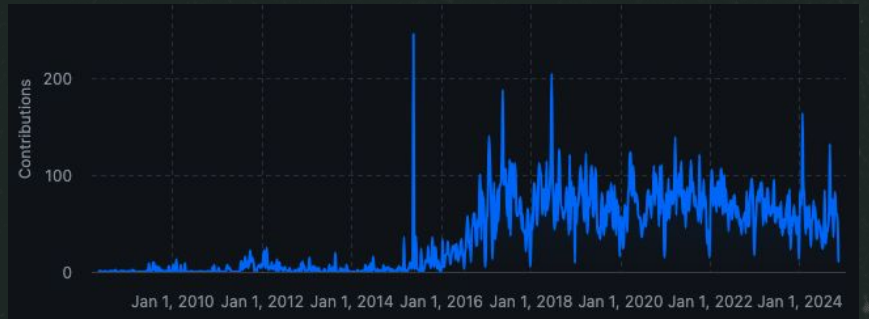
96 contributors
341 stars
35 watching and 154 forks
7 years last commit (in Dez 2024)
License GNU v2.0
C 98%



FRR

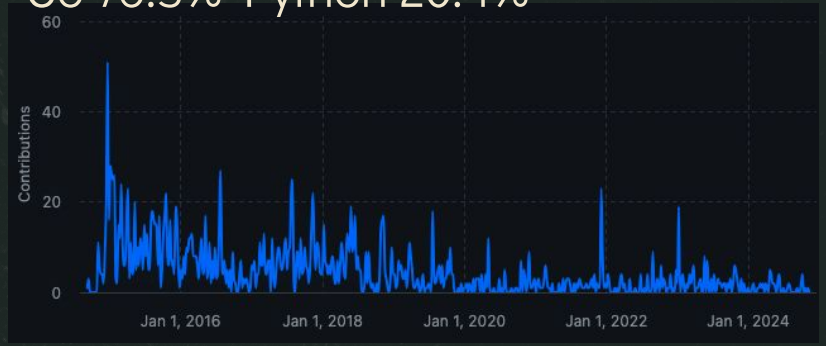
Fork Quagga 2017, but not only BGP

436 contributors & 354 issues
3.4k stars
155 watching and 1.3k forks
2 hours last commit (in Dez 2024)
License GNU v2.0
C 70.4% Python 27.6%



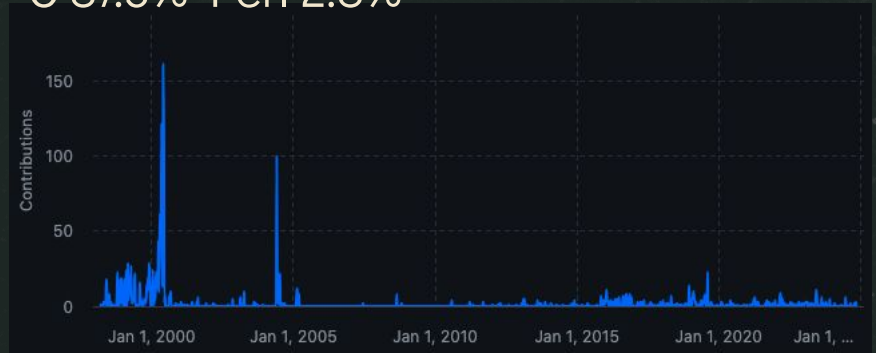
goBGP

JP 2014 Fujitsu, only BGP, no OSRG (Open Source Routing Group)
156 contributors and 171 issues
3.6k stars
121 watching and 697 forks
1 week last commit (in Dez 2024)
Apache 2.0
Go 76.5% Python 20.4%



Bird

CZ 1999 and BGP 2006, Bird 2.0 2018, not only BGP
35 contributors
92 stars
21 watching and 30 forks
5 days last commit (in Dez 2024)
GNU V2 License
C 87.3% Perl 2.8%



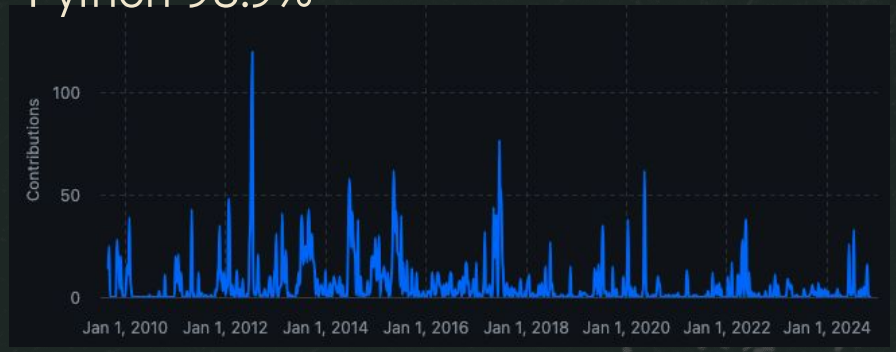
OpenBGPD

DE 2003 for OpenBSD, overcoming sec in Quagga, 2016 Docker.
6 contributors and 3 issues (*)
57 stars
11 watching and 15 forks
1 month last commit (in Dez 2024)
ISC License ~ BSD 2
C 89.5% M4 5.2%

ExaBGP

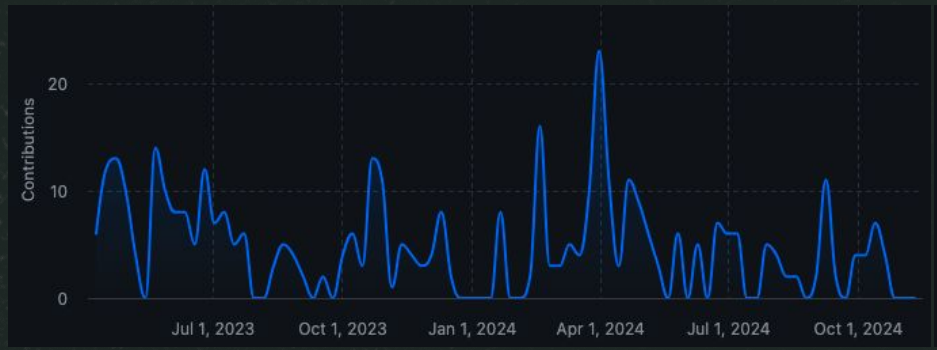
UK 2009 only BGP, API driven

109 contributors
2.1k stars
134 watching and 447 forks
1 week last commit (in Dez 2024)
Own license equivalent to BSD 3
Python 98.9%



holo-routing

BR 2023 by Renato Westphal
7 contributors and 4 issues
279 stars
15 watching and 17 forks
3 weeks last commit (in Dez 2024)
MIT License
Rust 100%



RustyBGP

2020 by OSRG (Open Source Routing Group -> goBGP)
9 contributors and 6 issues
485 stars
28 watching and 47 forks
4 months last commit (in Dez 2024)
Apache-2.0 License
Rust 91.9% Python 4.1%

Score

	Quagga	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
Adoption & Community								
Code & Flexibility								
Configuration & Usability								
Automation & API								
Security & Reliability								

Score

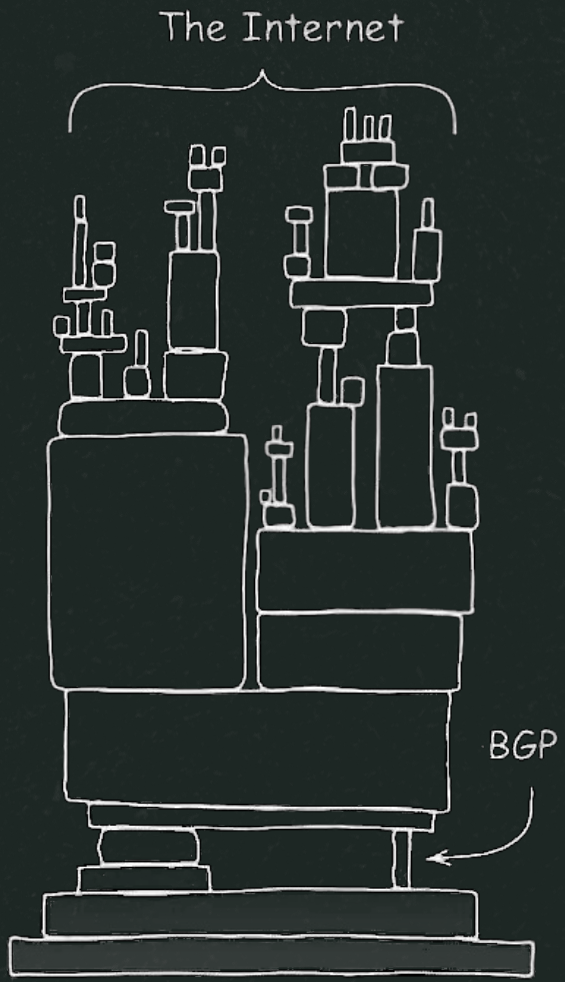
	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
Adoption & Community	6	10	7	7	6	9	8
Code & Flexibility							
Configuration & Usability							
Automation & API							
Security & Reliability							

Code Style & Flexibility

	Languages	Style	BGP only ?	Plugins?	Age in 2024	Flexibility	
FRR	C 70.4%, Python 27.6%, Perl 0.7% and M4 0.5%	C, complex	NO	Limited script based	7*	High, API support	4
goBGP	Go 76.5%, Python 20.4% and Shell 2.9%	Go, idiomatic	YES	YES	10	Very High, gRPC API	4.5
Bird	C 87.3, Yacc 6.2% and Perl 2.8%	C, efficient	NO	NO	18	Low, config-based	1
OpenBGPD	C 89.5%, M4 5.2% and Makefile 4.1%	C, secure	YES	NO	21	Low, config-based	1.5
ExaBGP	Python 98.9%	Python, simple	YES	YES	15	Very High, Python-based	3
holo-routing	Rust 100.0%	Rust, modular	NO	Limited via modules	2	Medium, extensible	4.5
RustyBGP	Rust 91.9%, Python 4.1% and Go 3.7%	Rust, modern	YES	NO	4	Medium, Rust API	2.5

Score

	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
Adoption & Community	5	10	7	7	5	9	8
Code & Flexibility	2	9	6	10	3	10	7
Configuration & Usability							
Automation & API							
Security & Reliability							


The Internet
BGP

Configuration & Usability

Configuration & Usability

	Config Format	Easy?	Doc	Tools	
FRR	CLI (vtysh), cisco text, JSON API	med/high	Extensive	Strong (gRPC, RESTCONF, Prom.)	4
goBGP	JSON, TOML, YAML, CLI, gRPC API	high	Excellent	Extensive (gRPC, APIs, Kubernetes)	4
Bird	Text based	medium	Good, but technical	Limited	1
OpenBGPD	Text based	High, simple	For OpenBSD	Limited	1.5
ExaBGP	Python-based scripts, JSON	High	Good, script focus	Strong Python integration	3.5
holo-routing	YAML/Rust-based config	Medium	Limited	Custom Rust modules	2
RustyBGP	YAML-based config	Medium	Limited	Minimum	1.5

Command Line Interface

	CLI Tool	Easy?	Syntax	Dynamic Changes	Remarks	
FRR	`vtysh`	yes	cisco-like	YES	Easy for network engineers with cisco-like configuration skills	4
goBGP	`gobgp`	yes	unix-style	YES	Easy for network engineers with unix-like configuration skils	4
Bird	`birdc`	medium	Custom, bird-style	Limited, need few restart	Adding a new BGP neighbor requires restarting the service	2.5
OpenBGPD	`bgpctl`	yes	unix-style	Limited, need few restart	Updating policy rules: to `filter`, `prefix-set`, or `rde filter` rules requires reload.	3.5
ExaBGP	No dedicated CLI	medium	script	YES	Scripts for everything	2
holo-routing	`holoctl`	medium	rust-style	Limited, module restart	(route policy) Need to reload the specific configuration module: `holoctl reload bgp`	2.5
RustyBGP	`rustybgp`	medium	rust-style	Limited, module restart	Adding neighbor requires a `rustybgp reload`	2.5

Score

	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
Adoption & Community	5	10	7	7	5	9	8
Code & Flexibility	2	9	6	10	3	10	7
Configuration & Usability	4	10	5	6	6	10	7
Automation & API							
Security & Reliability							

Automation & API

Automation & API

	API support	RESTfull	RESTCONF	YANG	Dynamic updates via API	
FRR	gRPC and RESTCONF API	via RESTCONF	YES	YES	YES	3
goBGP	gRPC and REST API	YES	NO	NO	YES	3
Bird	No external API	N/A	N/A	N/A	N/A	0
OpenBGPD	No external API	N/A	N/A	N/A	N/A	0
ExaBGP	Python API, JSON-based	NO	NO	NO	YES	1.5
holo-routing	gRPC API, gNMI support	NO	NO	YES	YES	2.5
RustyBGP	gRPC API	NO	NO	NO	YES	2

Score

	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
Adoption & Community	5	10	7	7	5	9	8
Code & Flexibility	2	9	6	10	3	10	7
Configuration & Usability	4	10	5	6	6	10	7
Automation & API	0	10	7	8	0	10	5
Security & Reliability							


ROUTE
BGP

Security & Reliability

Security & Reliability

	RKPI support	TCP MD5	TTL security	Vulnerability management	Reliability	Attention !	
FRR	YES	YES	YES	High (large community, fast patching)	High	none	5.5
goBGP	YES	YES	YES	High (active project, regular updates)	High	none	5.5
Bird	NO	YES	YES	Moderate (depends on community patches)	High	Basic filtering, relies on external security measures	3
OpenBGPD	YES	YES	YES	High (OpenBSD's security focus)	High	none	5.5
ExaBGP	NO	NO	NO	Moderate (depends on script hygiene)	High	Limited native security, relies on script-level controls	1
holo-routing	NO	YES	YES	High (cargo-audit, Rust's safety, no-crates.io)	High	none	5
RustyBGP	NO	YES	YES	High (cargo-audit, Rust's safety, no-creates.io)	High	none	5

Score

	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
Adoption & Community	5	10	7	7	5	9	8
Code & Flexibility	2	9	6	10	3	10	7
Configuration & Usability	4	10	5	6	6	10	7
Automation & API	0	10	7	8	0	10	5
Security & Reliability	5	10	9	9	10	10	2

BGP Features

IETF RFCs a Very very long list...

As of November 2024, there are around 150 RFCs directly focused on BGP. Including indirect RFCs related to SNMP, YANG, NETCONF, and MPLS, this adds an estimated 200-300 more. Altogether, the total number of direct and indirect BGP-related RFCs could easily exceed a thousand, depending on the scope considered.

What to look for ?

Security Enhancements	Default eBGP Filtering, RPKI, BGPsec, Route Leak Prevention	RFC 8212, RFC 6811, RFC 8205, RFC 9286, RFC 8674
Traffic Engineering & Policy Control	EVPN, Add-Path, Flowspec, Segment Routing, Large Communities	RFC 7432, RFC 8365, RFC 7911, RFC 8955, RFC 8956, RFC 8957, RFC 8669, RFC 9087, RFC 8092, RFC 7999
Scalability & Convergence Enhancements	AS Path Limit, Prefix Aggregation, Best External Route Selection	RFC 9107, RFC 7311, RFC 7806
Monitoring & Visibility	BMP, MRT Format, YANG Models, gNMI	RFC 7854, RFC 8050, RFC 9085, gNMI Specification
Automation and Dynamic Capabilities	Dynamic Capability, BGP-LS, YANG BGP	RFC 7313, RFC 7752, RFC 9085, RFC 9067, RFC 8349, RFC 8294

My list

Route Optimization	Includes efficient prefix aggregation and route leak prevention to reduce table size and improve convergence.	RFC 9107
Route Leak Prevention	Defines the BGP Role attribute to prevent route leaks by enforcing correct session roles.	RFC 7908, RFC 9286
BGP FlowSpec	Extends BGP for flow-based traffic filtering, enabling dynamic DDoS mitigation and traffic shaping.	RFC 8955, RFC 8956, RFC 8957
gNMI methods	A gRPC-based API for network configuration and telemetry, supporting real-time monitoring and automation.	Own gNMI Specification or YANG models
EVPN	Provides scalable L2/L3 VPN services using BGP, enabling multi-tenancy, VXLAN overlays, and data center interconnects.	RFC 7432, RFC 8365
Segment Routing	Enables BGP to advertise Segment Routing SIDs for efficient traffic engineering without traditional MPLS.	RFC 8669, RFC 9087

BGP Features

	Route Optimization	Leak Prevention	flowspec	gNMI	EVPN	Segment Routing	
FRR	YES	YES	YES	Supports Capabilities, Get, and Subscribe; limited or no Set support	YES	YES	5.5
goBGP	YES	YES	YES	YES, fully supports all four methods: Capabilities, Get, Set, and Subscribe	YES	YES	6
Bird	YES	YES	YES	NO	NO	NO	3
OpenBGPD	YES	YES	NO	NO	NO	NO	2
ExaBGP	YES	YES	YES	NO	NO	NO	3
holo-routing	YES	YES	NO	YES, fully supports all four methods: Capabilities, Get, Set, and Subscribe	YES	YES	5
RustyBGP	YES	YES	NO	NO	NO	NO	2

Score

	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
BGP Features	5	9	2	8	2	10	5
Deployment, HA & Oper							
Performance & Scalability							

Deployment, HA & Operation

Deployment, HA & Operation

	Binary structure	Resource footprint	BFD	Graceful restart	Dynamic config	Active Passive	Native Docker	
FRR	Multiple binaries	Medium	YES	Yes (Full support)	Yes (Modular, no restart)	Yes (Via VRRP or Keepalived)	Yes (Official Docker images)	6.5
goBGP	Single binary	Medium	YES	Yes (Full support)	Yes (APIs for live updates)	Yes (APIs enable setups)	Yes (Official Docker images)	6.5
Bird	Single binary	Low	NO	Basic (Limited support)	No (Restart required)	NO	NO	2.5
OpenBGPD	Single binary	Very low	NO	Partial (Retains routes)	No (Restart required)	NO	NO	2.5
ExaBGP	Python based	High	NO	NO	Yes (External scripting)	NO	Yes (Docker-ready Python)	1
holo-routing	Single binary	Low	YES	Yes (Full support)	Yes (Supports live updates)	Yes (Via orchestration tools)	Yes (Docker support provided)	7
RustyBGP	Single binary	Low	NO	Partial (Limited state retention)	Yes (Dynamic updates supported)	NO	NO	2.5

Score

	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
BGP Features	5	9	2	8	2	10	5
Deployment, HA & Oper	4	9	4	10	4	9	2
Performance & Scalability							

Interesting fact

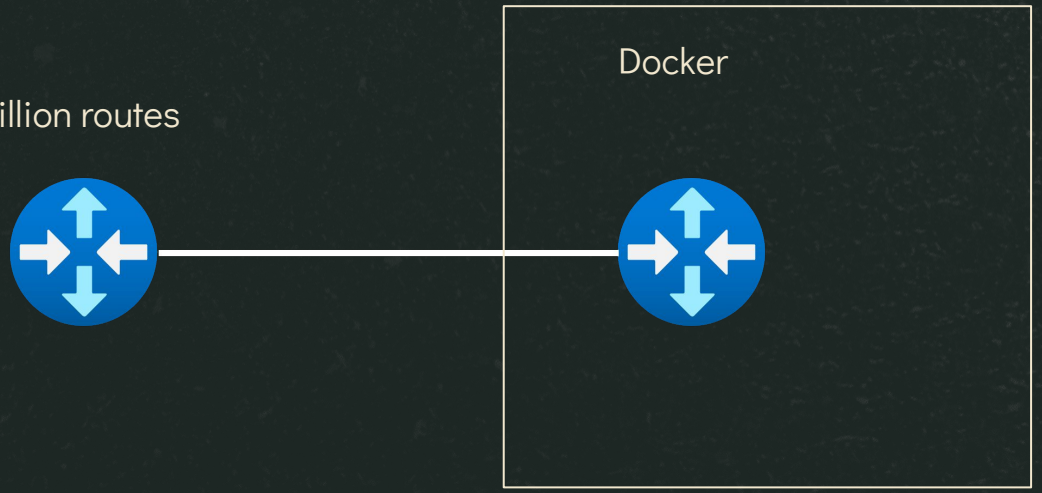
In early network standardization, the ISO and the Internet community competed for dominance (IPX and IP).

ISO's **IDRP** (Inter-Domain Routing Protocol), developed in **1993**, introduced key ideas for inter-domain routing, many of which were borrowed by the IETF in **1994** with the successful implementation of **BGP-4,** which remains today.

Performance

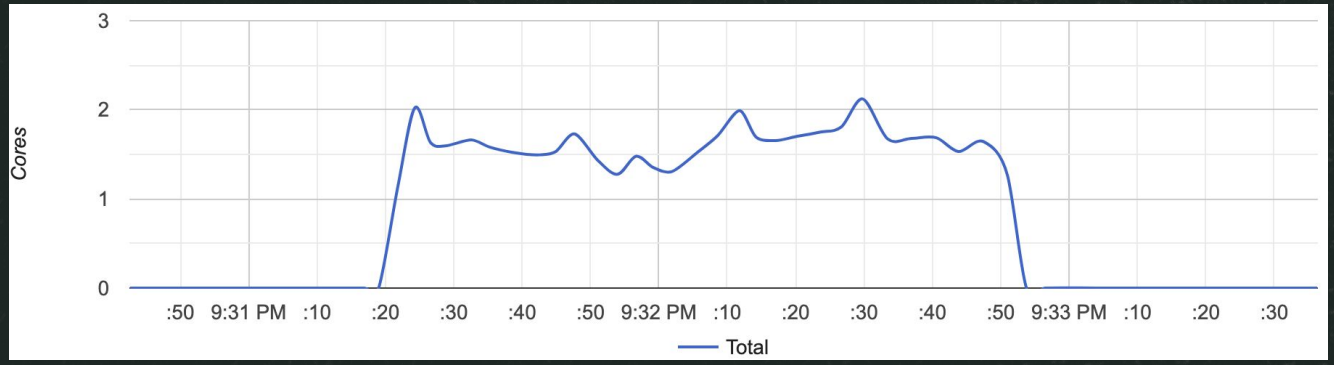
Setup

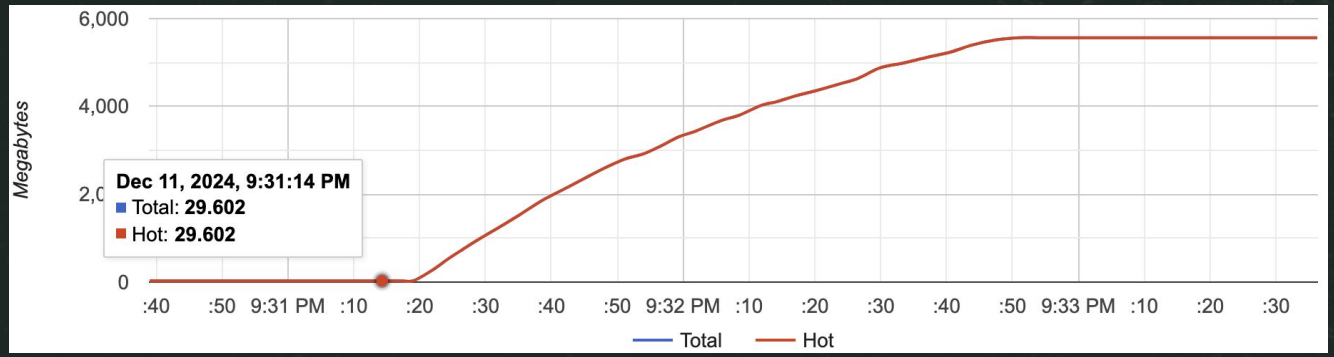
24 Cores
256GB RAM

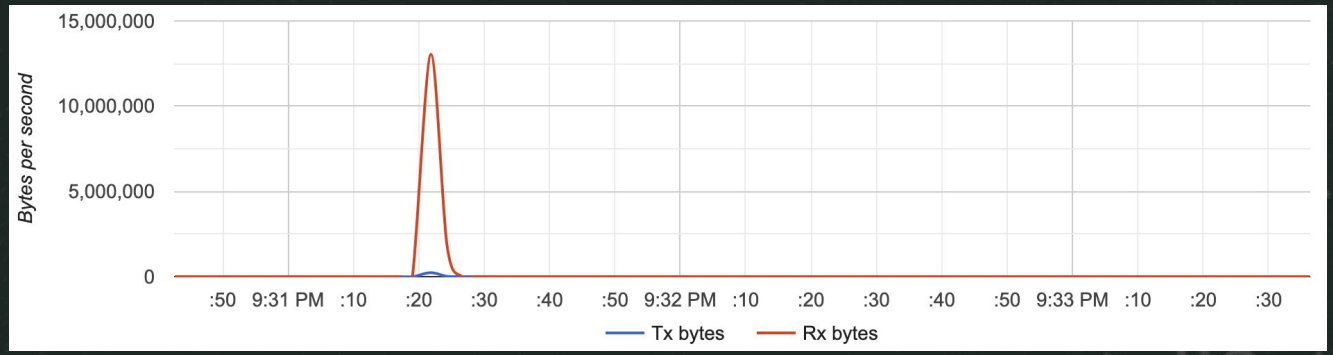



```
% docker images
REPOSITORY                      TAG                 IMAGE ID       CREATED          SIZE
frr                             latest              06855a86cce2   11 hours ago     1.53GB
openbgpd                        latest              62eff1393420   2 days ago       395MB
rustybgp                        latest              d79449331a46   2 days ago       98.1MB
bird                            latest              4864bcb6ae9e   3 days ago       449MB
exabgp                          latest              af407d72bf89   5 days ago       266MB
gobgp                           latest              01c7a0c584e7   5 days ago       130MB
holo                            latest              ce62f7c32843   8 days ago       125MB
```

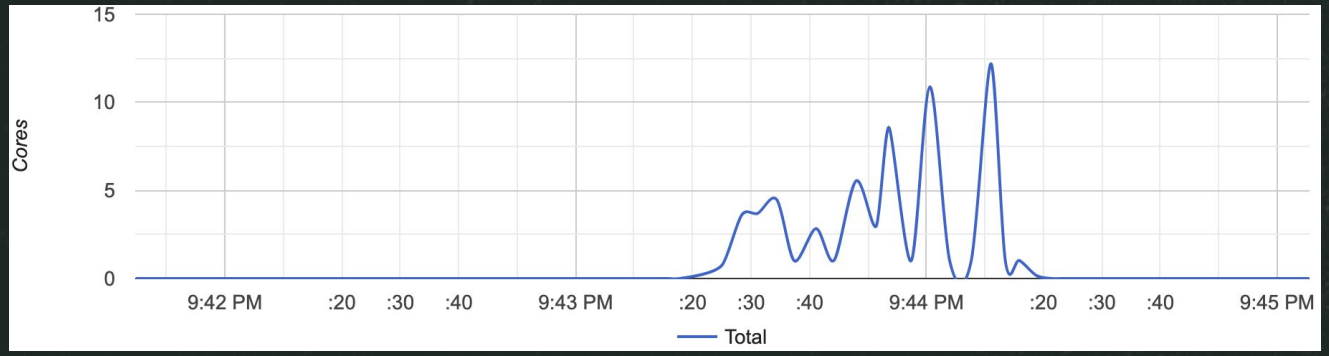
Frr

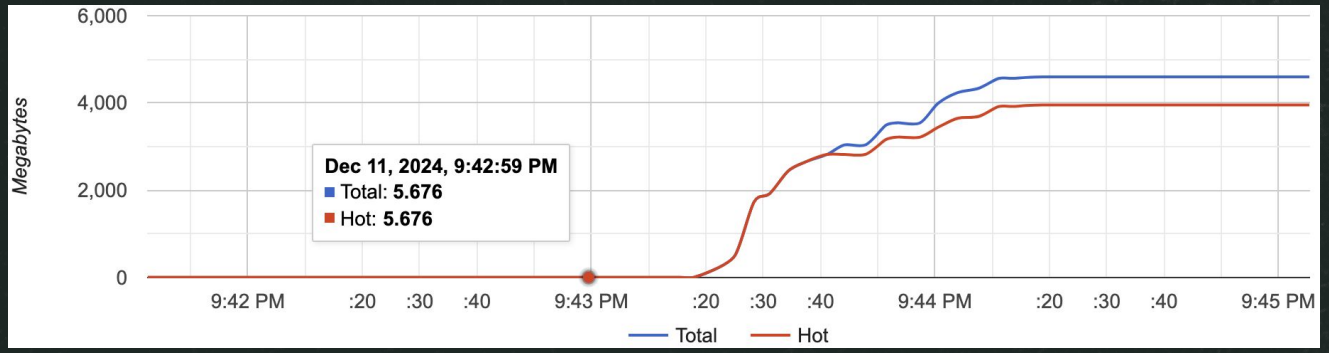

Cores
3
2
1
0
:50 9:31 PM :10 :20 :30 :40 :50 9:32 PM :10 :20 :30 :40 :50 9:33 PM :10 :20 :30
Total

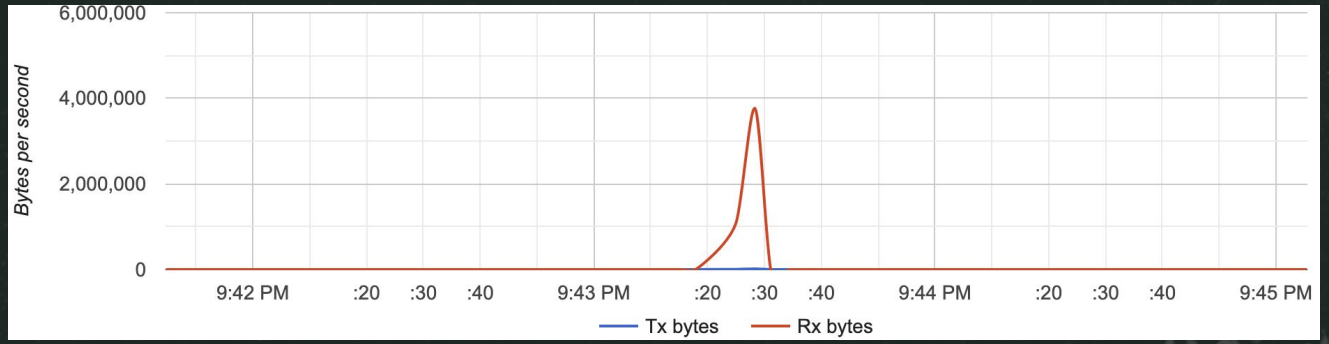

Megabytes
6,000
4,000
2,0
0
Dec 11, 2024, 9:31:14 PM
Total: 29.602
Hot: 29.602
:40 :50 9:31 PM :10 :20 :30 :40 :50 9:32 PM :10 :20 :30 :40 :50 9:33 PM :10 :20 :30
Total
Hot


Bytes per second
15,000,000
10,000,000
5,000,000
0
:50 9:31 PM :10 :20 :30 :40 :50 9:32 PM :10 :20 :30 :40 :50 9:33 PM :10 :20 :30
Tx bytes
Rx bytes

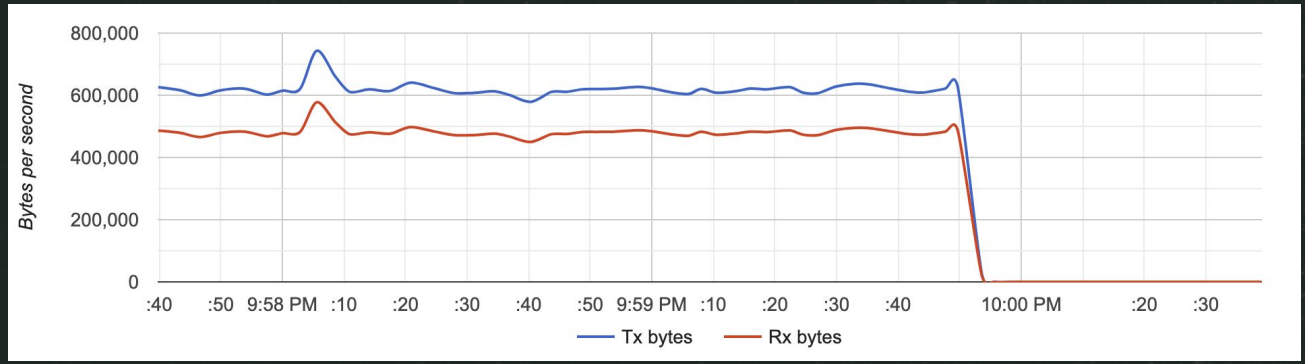
goBGP


Cores
15
10
5
0
9:42 PM
:20
:30
:40
9:43 PM
:20
:30
:40
9:44 PM
:20
:30
:40
9:45 PM
Total


Megabytes
6,000
4,000
2,000
0
Dec 11, 2024, 9:42:59 PM
Total: 5.676
Hot: 5.676
9:42 PM
:20
:30
:40
9:43 PM
:20
:30
:40
9:44 PM
:20
:30
:40
9:45 PM
Total
Hot

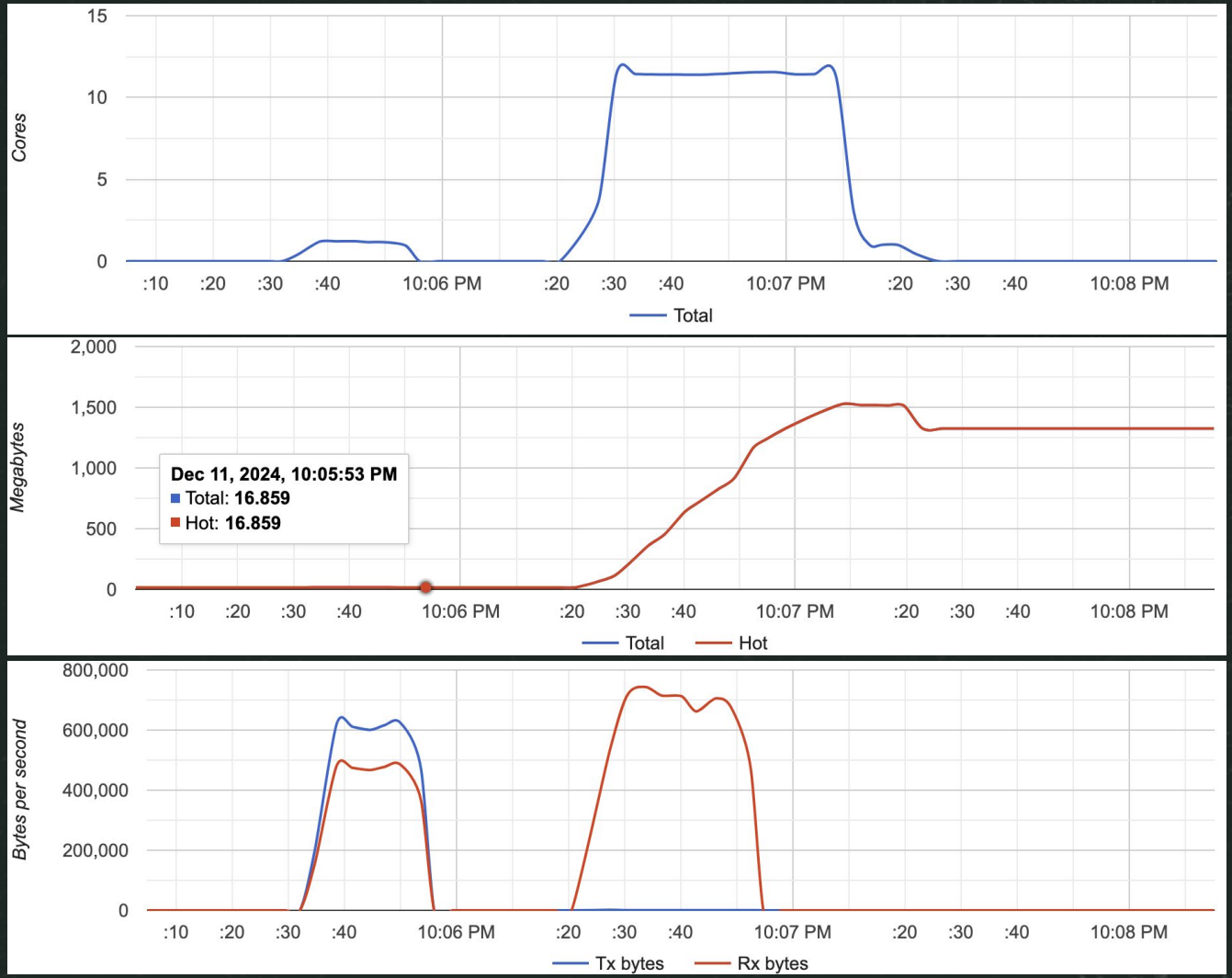

Bytes per second
6,000,000
4,000,000
2,000,000
0
9:42 PM
:20
:30
:40
9:43 PM
:20
:30
:40
9:44 PM
:20
:30
:40
9:45 PM
Tx bytes
Rx bytes

rustyBGP

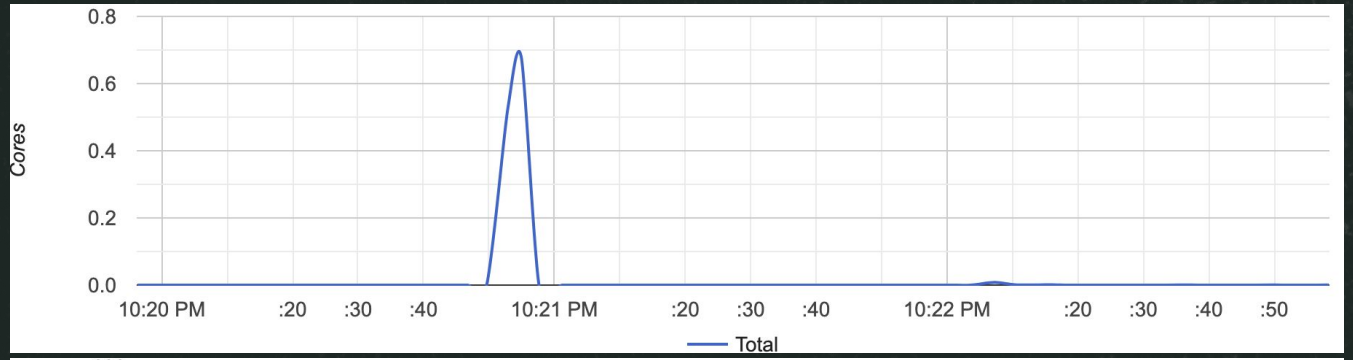


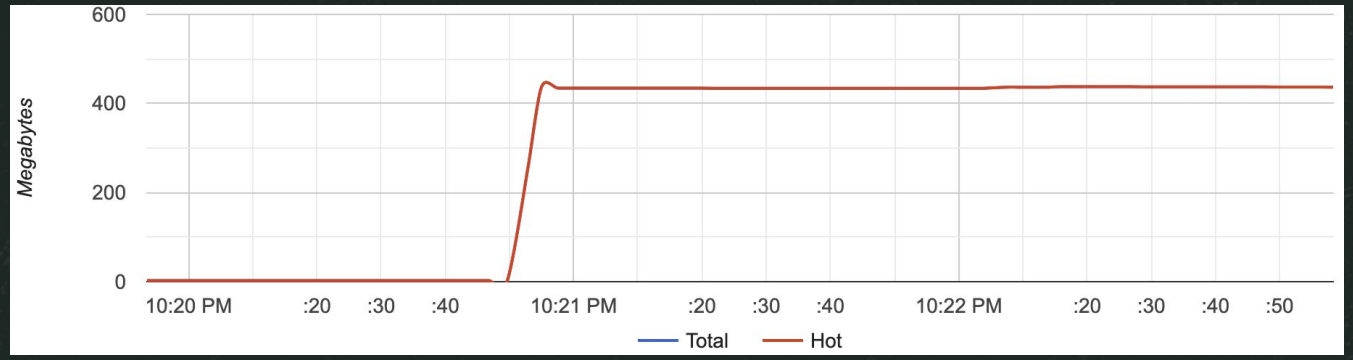

```
21:58:27.595954 IP 192.168.200.10.49368 > 192.168.200.1.179: Flags [S], seq 2890054387, win 64240, options [mss
1460,sackOK,TS val 2001568115 ecr 0,nop,wscale 7], length 0
21:58:27.596261 IP 192.168.200.10.49376 > 192.168.200.1.179: Flags [S], seq 484301566, win 64240, options [mss
1460,sackOK,TS val 2001568115 ecr 0,nop,wscale 7], length 0
21:58:27.596612 IP 192.168.200.10.49388 > 192.168.200.1.179: Flags [S], seq 1146106257, win 64240, options [mss
1460,sackOK,TS val 2001568115 ecr 0,nop,wscale 7], length 0
21:58:27.597045 IP 192.168.200.10.49392 > 192.168.200.1.179: Flags [S], seq 680279469, win 64240, options [mss
1460,sackOK,TS val 2001568116 ecr 0,nop,wscale 7], length 0
21:58:27.597500 IP 192.168.200.10.49398 > 192.168.200.1.179: Flags [S], seq 193656389, win 64240, options [mss
1460,sackOK,TS val 2001568116 ecr 0,nop,wscale 7], length 0
21:58:27.597927 IP 192.168.200.10.49414 > 192.168.200.1.179: Flags [S], seq 3419692504, win 64240, options [mss
1460,sackOK,TS val 2001568117 ecr 0,nop,wscale 7], length 0
21:58:27.598341 IP 192.168.200.10.49424 > 192.168.200.1.179: Flags [S], seq 2601758478, win 64240, options [mss
1460,sackOK,TS val 2001568117 ecr 0,nop,wscale 7], length 0
21:58:27.598791 IP 192.168.200.10.49438 > 192.168.200.1.179: Flags [S], seq 3477881123, win 64240, options [mss
1460,sackOK,TS val 2001568118 ecr 0,nop,wscale 7], length 0
21:58:27.599146 IP 192.168.200.10.49446 > 192.168.200.1.179: Flags [S], seq 979313717, win 64240, options [mss
1460,sackOK,TS val 2001568118 ecr 0,nop,wscale 7], length 0
```

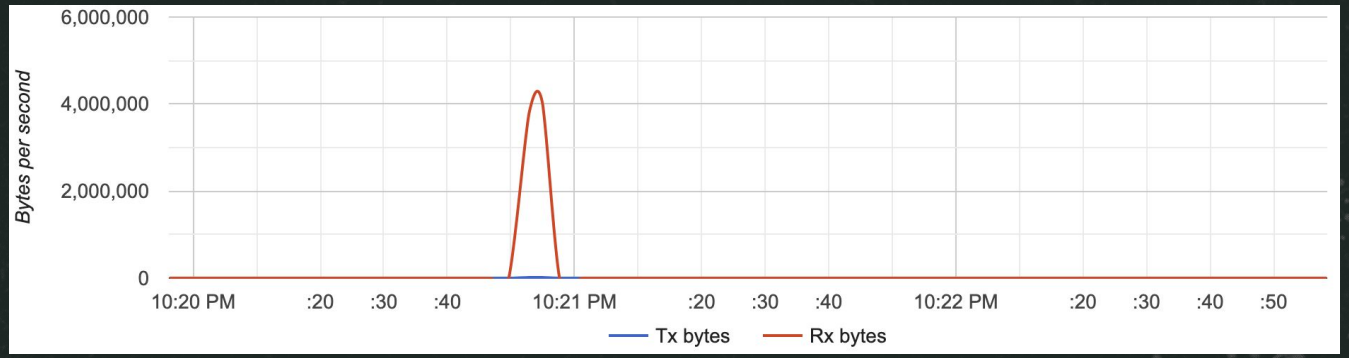
rustyBGP


Cores
15
10
5
0
:10
:20
:30
:40
10:06 PM
:20
:30
:40
10:07 PM
:20
:30
:40
10:08 PM
Total
Megabytes
2,000
1,500
1,000
500
0
Dec 11, 2024, 10:05:53 PM
Total: 16.859
Hot: 16.859
Total
Hot
Bytes per second
800,000
600,000
400,000
200,000
0
Tx bytes
Rx bytes

Bird


Cores
0.8
0.6
0.4
0.2
0.0
10:20 PM
:20
:30
:40
10:21 PM
:20
:30
:40
10:22 PM
:20
:30
:40
:50
Total


Megabytes
600
400
200
0
10:20 PM
:20
:30
:40
10:21 PM
:20
:30
:40
10:22 PM
:20
:30
:40
:50
Total
Hot


Bytes per second
6,000,000
4,000,000
2,000,000
0
10:20 PM
:20
:30
:40
10:21 PM
:20
:30
:40
10:22 PM
:20
:30
:40
:50
Tx bytes
Rx bytes

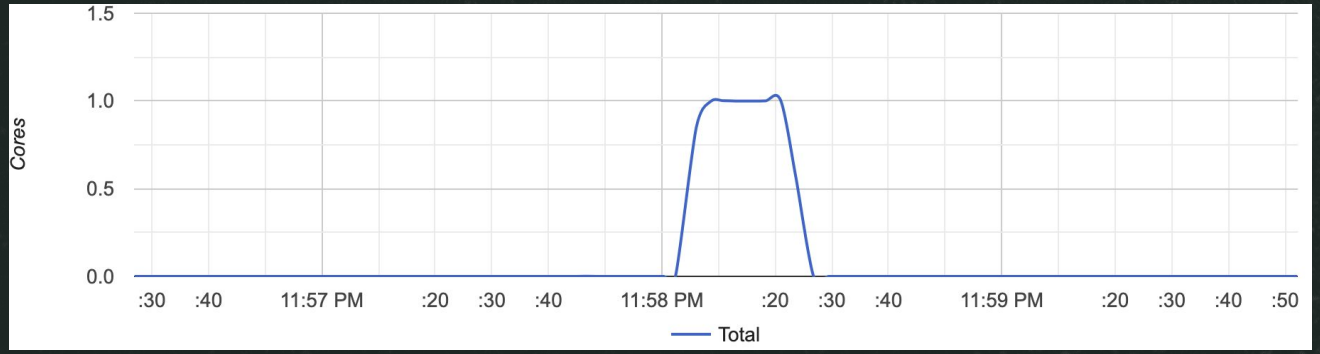
Bird

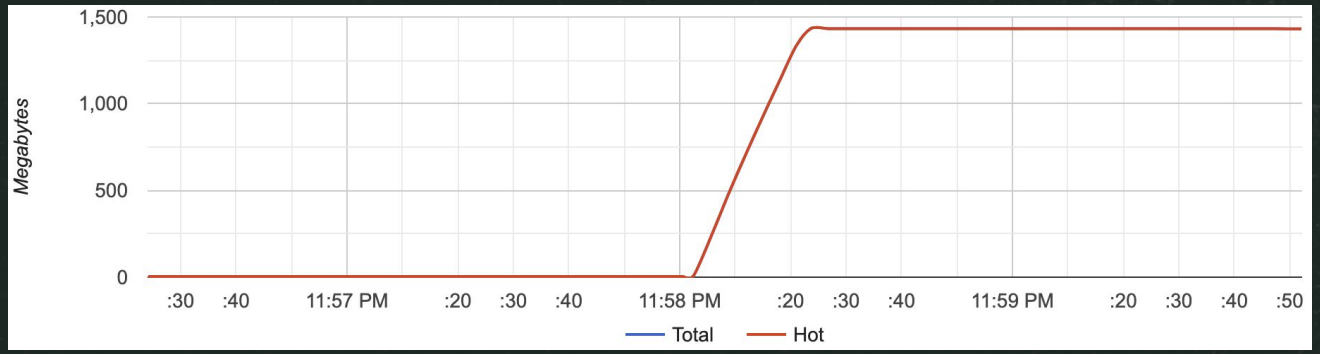
```
Channel ipv4
  State:          UP
  Table:          master4
  Preference:     100
  Input filter:   ACCEPT
  Output filter:  ACCEPT
  Routes:         4665406 imported, 0 exported, 4665406 preferred
  Route change stats:     received   rejected   filtered    ignored   accepted
    Import updates:        5000000     334594          0          0    4665406
    Import withdraws:            0          0        ---          0          0
    Export updates:        4665406    4665406          0        ---          0
    Export withdraws:            0        ---        ---        ---          0
  BGP Next hop:   192.168.200.10
  IGP IPv4 table: master4
```

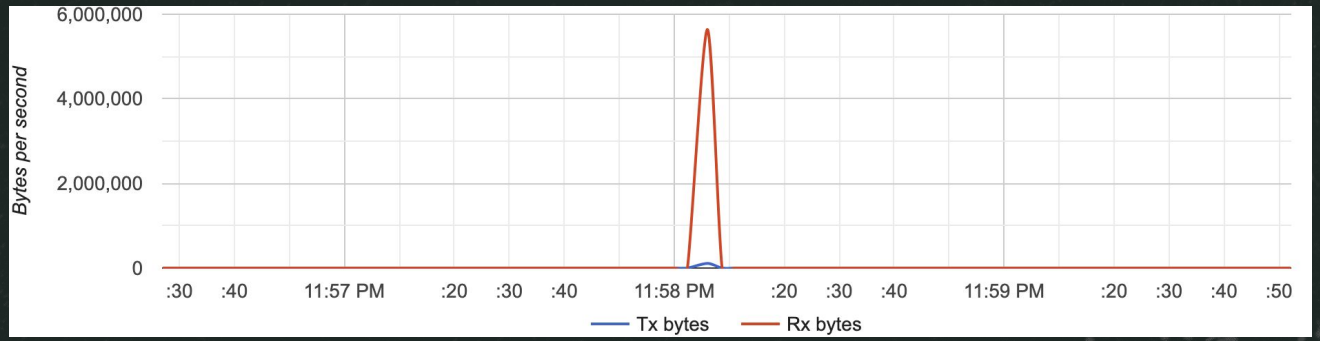
Holo

```
holo# show running
!
interfaces interface eth0
 type iana-if-type:ethernetCsmacd
 ipv4
!
routing control-plane-protocols control-plane-protocol ietf-bgp:bgp teste
 !
 bgp neighbors neighbor 192.168.200.1
  peer-as 65000
  local-as 65000
  transport local-address 192.168.200.10
!
routing control-plane-protocols control-plane-protocol ietf-bgp:bgp test
 bgp global
 bgp global as 65000
 bgp global identifier 192.168.200.10
 !
 bgp neighbors neighbor 192.168.200.1
  peer-as 65000
  !
  afi-safis afi-safi iana-bgp-types:ipv4-unicast
   enabled true
   apply-policy default-import-policy accept-route
   apply-policy default-export-policy accept-route
!
```

OpenBGP


Cores
1.5
1.0
0.5
0.0
:30
:40
11:57 PM
:20
:30
:40
11:58 PM
:20
:30
:40
11:59 PM
:20
:30
:40
:50
Total

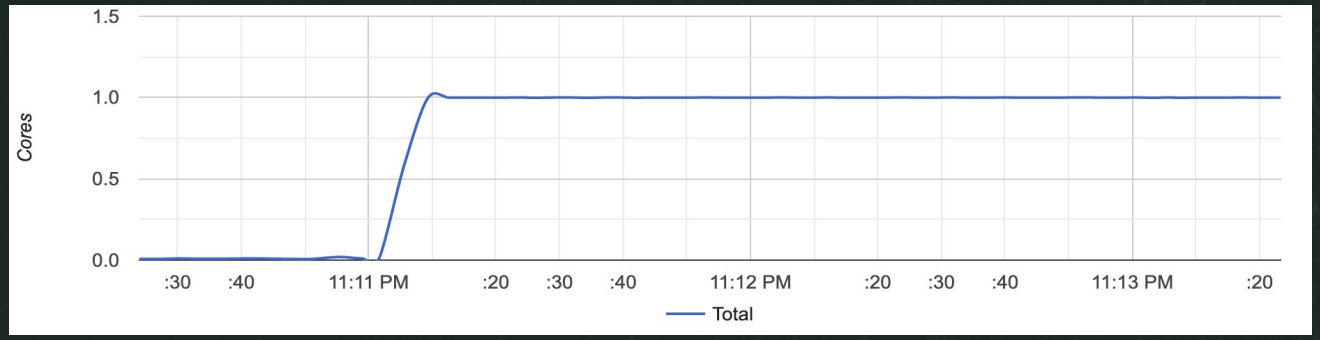

Megabytes
1,500
1,000
500
0
:30
:40
11:57 PM
:20
:30
:40
11:58 PM
:20
:30
:40
11:59 PM
:20
:30
:40
:50
Total
Hot

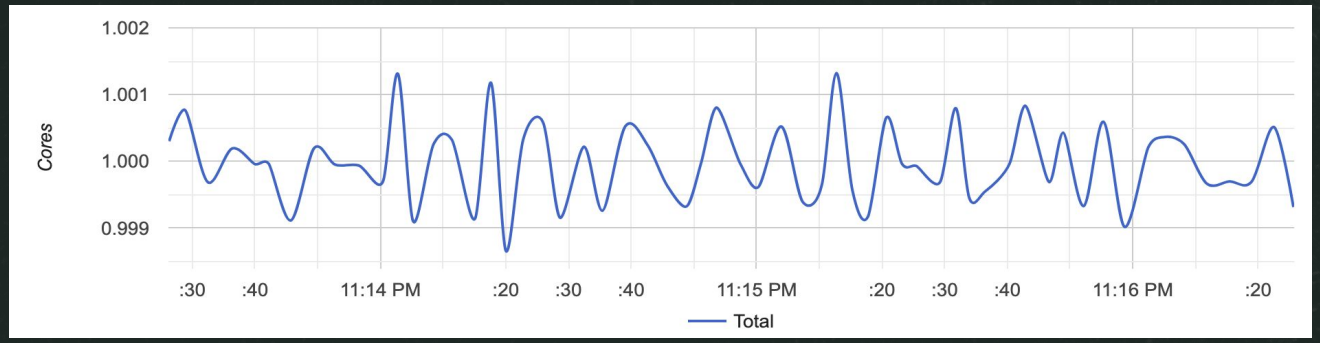

Bytes per second
6,000,000
4,000,000
2,000,000
0
:30
:40
11:57 PM
:20
:30
:40
11:58 PM
:20
:30
:40
11:59 PM
:20
:30
:40
:50
Tx bytes
Rx bytes

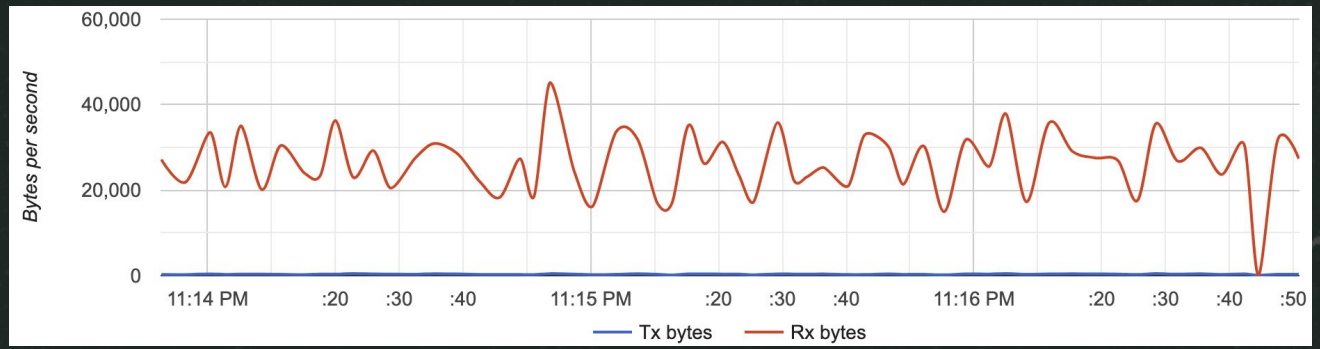
OpenBGP

```
%bgpctl show neighbor
BGP neighbor is 192.168.200.1, remote AS 65000
  BGP version 4, remote router-id 192.168.200.1
  BGP state = Established, up for 00:01:12
  Last read 00:00:26, holdtime 90s, keepalive interval 30s
  Last write 00:00:12
  Neighbor capabilities:
    Multiprotocol extensions: IPv4 unicast
    4-byte AS numbers
    Route Refresh
    Add-path: IPv4 unicast bidir
  Negotiated capabilities:
    Multiprotocol extensions: IPv4 unicast
    4-byte AS numbers
    Route Refresh

  Update statistics:
                  Sent       Received   Pending
  Prefixes           0    5000000
  Updates            0    5000000          0
  Withdraws          0          0          0
  End-of-Rib         0          0
  Route Refresh statistics:
  Request            0          0
  Begin-of-RR        0          0
  End-of-RR          0          0

  Local host:        192.168.200.10, Local port:    179
  Remote host:        192.168.200.1, Remote port: 47093
```

ExaBGP


1.5
1.0
0.5
0.0
Cores
:30
:40
11:11 PM
:20
:30
:40
11:12 PM
:20
:30
:40
11:13 PM
:20
Total

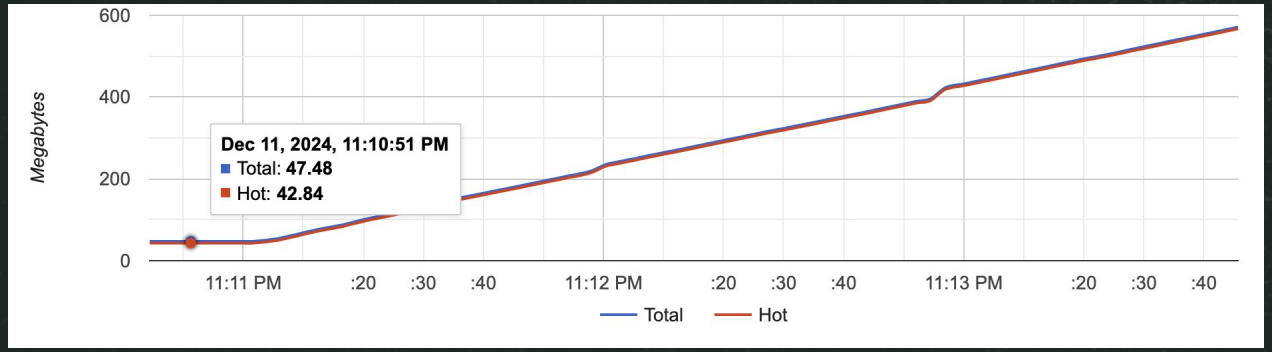

1.002
1.001
1.000
0.999
Cores
:30
:40
11:14 PM
:20
:30
:40
11:15 PM
:20
:30
:40
11:16 PM
:20
Total

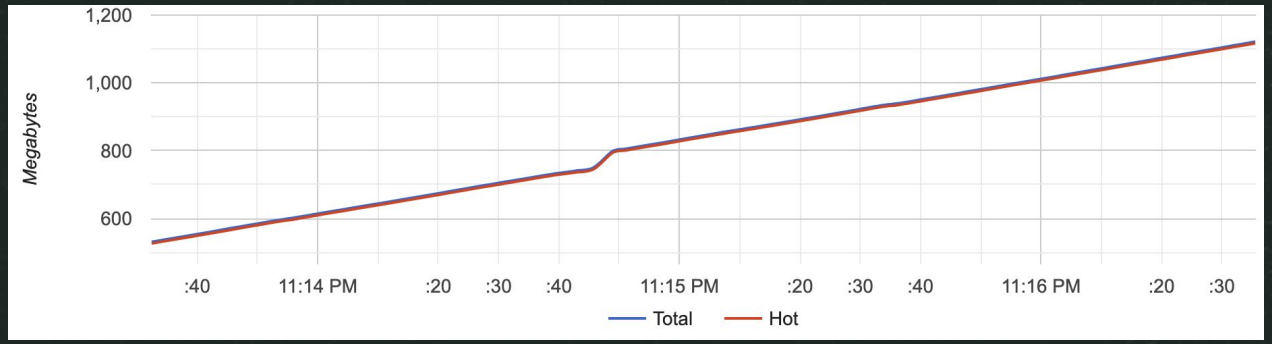

60,000
40,000
20,000
0
Bytes per second
11:14 PM
:20
:30
:40
11:15 PM
:20
:30
:40
11:16 PM
:20
:30
:40
:50
Tx bytes
Rx bytes

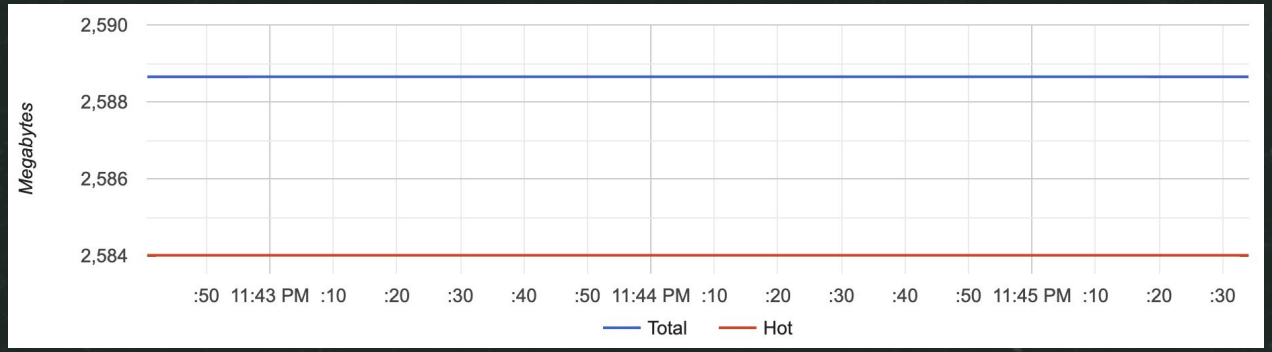
ExaBGP

13 min total

200MB/min







Score

	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
BGP Features	5	9	2	8	2	10	5
Deployment, HA & Oper	4	9	4	10	4	9	2
Performance & Scalability	10 ?	6	7	?	9 ?	8	1

Results

Final Score

	Bird	FRR	RustyBGP	holo-routing	OpenBGPD	goBGP	ExaBGP
No Performance	25	67	40	58	30	68	36
With Performance ??	35	73	47	58	39	76	37

So what ?

Do you like Python and Python is your uncle and only limited injections ? ***ExaBGP***

Fast and small footprint DDoS mitigation with flowspec ? ***Bird & GoBGP***

Do you wanna play safe, easy support and community ? ***FRR, GoBGP & Bird***

Do you want use gRPC to inject and collect updates ? ***GoBGP & Holo***

Are your developers as fluent in C as fish are in water ? ***Bird, OpenBGP & FRR***

Do you want to build your own router ? ***FRR, Holo & Bird***

Thank you

Let's talk about Traffic and BGP Analysis.

www.telcomanager.com

Network Automation Go/Python book : `a.co/d/iO7iXMe`

Github : `github.com/brnuts`

Linkedin : `www.linkedin.com/in/claus-topke/`

Work email : `claus@telcomanager.com`

Personal email : `claus.topke@gmail.com`