Cisco live!

January 29 – February 2, 2018 · Barcelona

BRKSPG-3012

# SP Security

## Leveraging **BGP FlowSpec** to protect your infrastructure

Nicolas Fevrier, Technical Leader Engineering
@CiscoIOSXR

Cisco live!

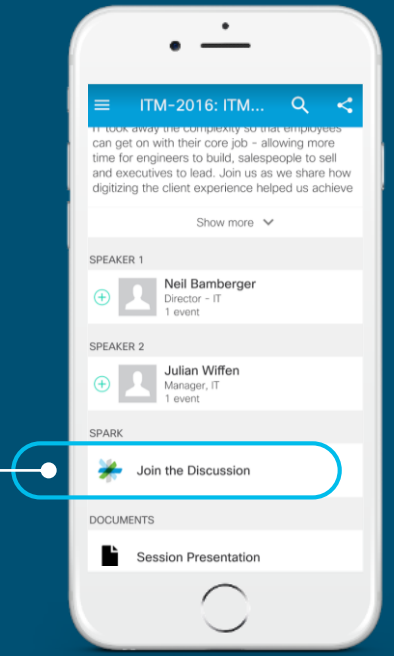
# Cisco Spark

## Questions?

Use Cisco Spark to communicate with the speaker after the session

## How

1. Find this session in the Cisco Live Mobile App
2. Click “Join the Discussion”
3. Install Spark or go directly to the space
4. Enter messages/questions in the space



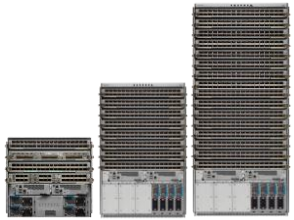

cs.co/ciscolivebot#BRKSPG-3012

# What We Hope To Achieve With This Session

- Introduce BGP Flowspec
- Clarify what it can do and where it fits
- DDoS Mitigation is not the only use-case in production
- Provide one more tool to your networking belt

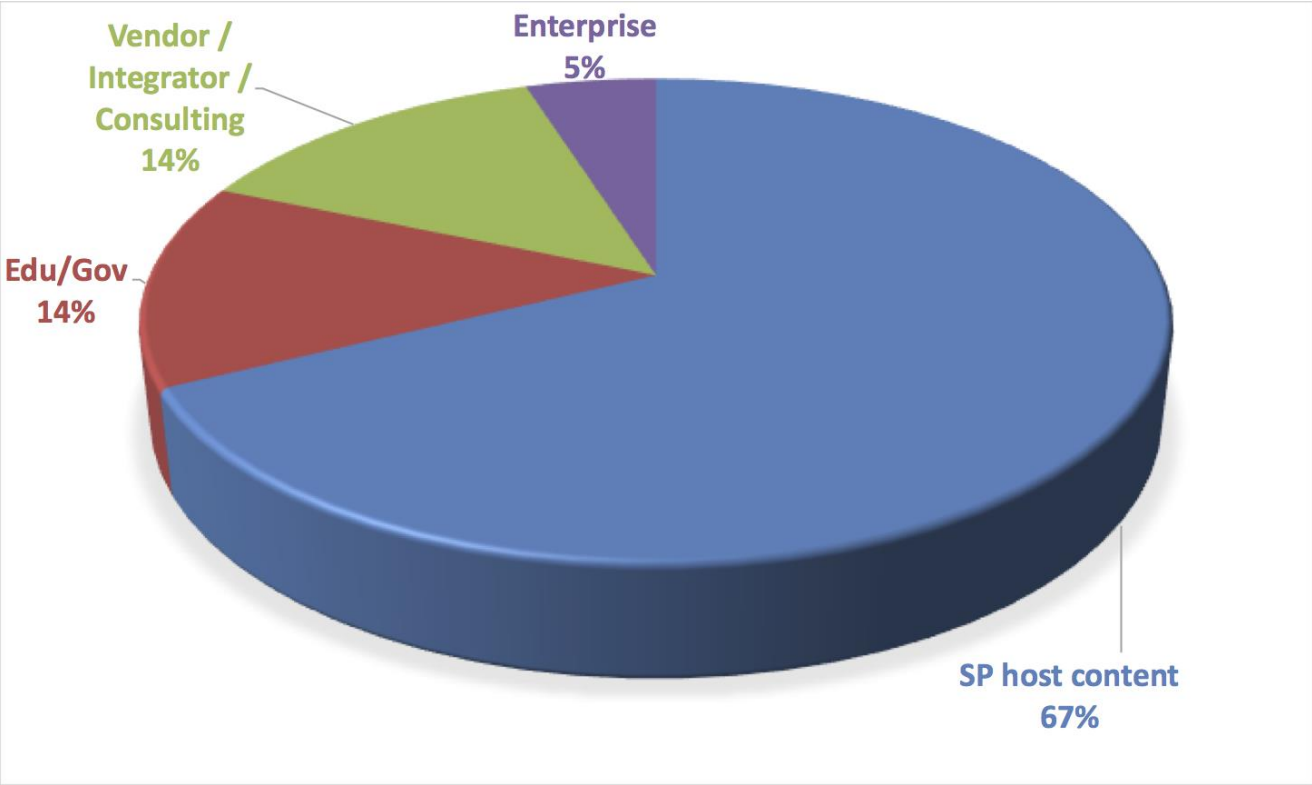
# Me ?

- Nicolas Fevrier
- TL / Technical Marketing Engineer based in Paris
- Service Providers BU
- In Cisco since 2004
  - Worked on all IOS XR Platforms
  - from CRS-1 to NCS5500
  - Worked in Services/Deployment and BU



# You ?

CiscoLive attendees registered to this session



# Agenda

- Introduction
- BGP FlowSpec Protocol Description
- Use-cases, Demo
  w/ DDoS Mitigation
- Configuring the Protocol
- Caveats and Limitations
- Conclusion

# Acknowledgements


- Andy Karch
- Bertrand Duvivier
- Gunter Van de Velde
- Brian Prater
- Kirill Kasavchenko
- Tomas Sundstrom

# Another 180+ Pages Slidedeck ?

- 90 Minutes
- Large “Back Up Slides” section
- Use of “For your reference” logo

# Introduction

# Introduction

- August 2009, IETF ratified “Dissemination of Flow Specification Rules”
- Separation of controlling and forwarding plane. Sounds familiar ?
- A powerful tool in the SP Security toolbox but Use-cases are expending way beyond Security

Secure https://tools.ietf.org/html/rfc5575

[Docs] [txt|pdf] [draft-ietf-idr-fl...] [Diff1] [Diff2] [IPR] [Errata]

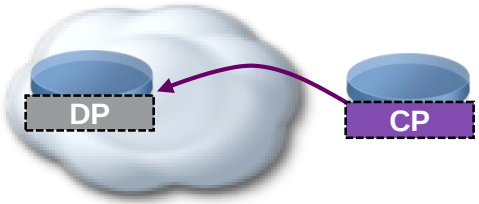
Updated by: 7674 PROPOSED STANDARD
Errata Exist

Network Working Group P. Marques
Request for Comments: 5575 Cisco Systems
Category: Standards Track N. Sheth
Juniper Networks
R. Raszuk
Cisco Systems
B. Greene
Juniper Networks
J. Mauch
NTT America
D. McPherson
Arbor Networks
August 2009

**Dissemination of Flow Specification Rules**

Abstract

This document defines a new Border Gateway Protocol Network Layer Reachability Information (BGP NLRI) encoding format that can be used to distribute traffic flow specifications. This allows the routing system to propagate information regarding more specific components of the traffic aggregate defined by an IP destination prefix.



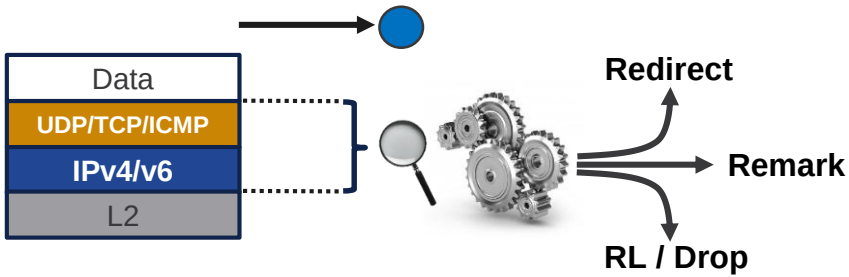
# Introduction

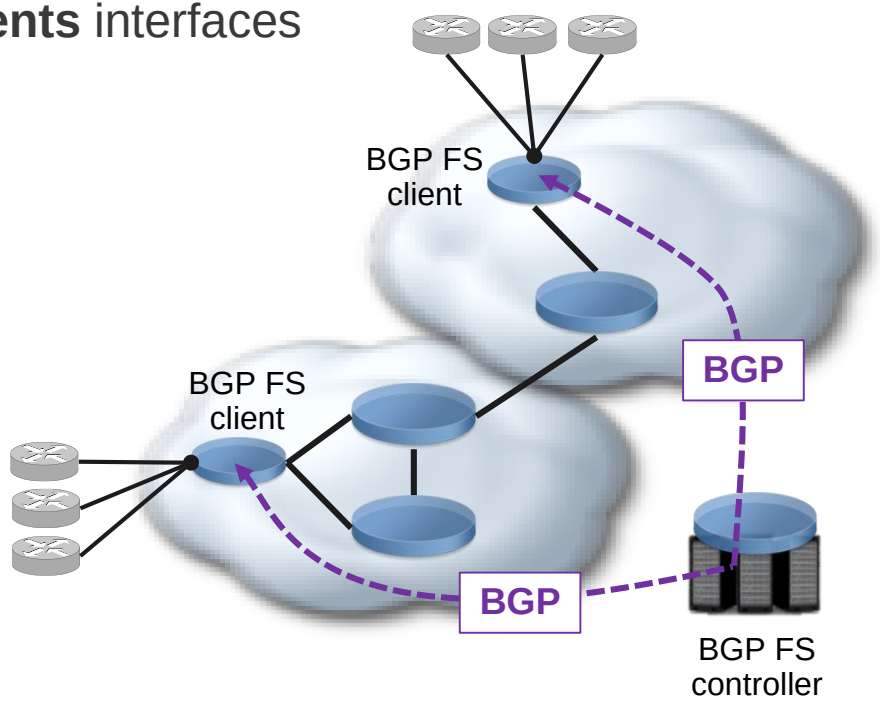
BGP FlowSpec is not:

- Netflow
  - Sample traffic and generate records from local table collector
- Openflow
  - But similarities exist
- Microflow Policing
  - Per user rate-limiting, some overlap

# Introduction

- A **Controller** programs remotely how packets should be treated when received on **Clients** interfaces
  - Remote PBR: redirect packet in VRF X
  - Remote PBR: redirect packet to @IP X
  - Remote QoS: DSCP Marking
  - Remote QoS: Policing (rate-limiter)
  - Remote ACL: Policing to 0 bps

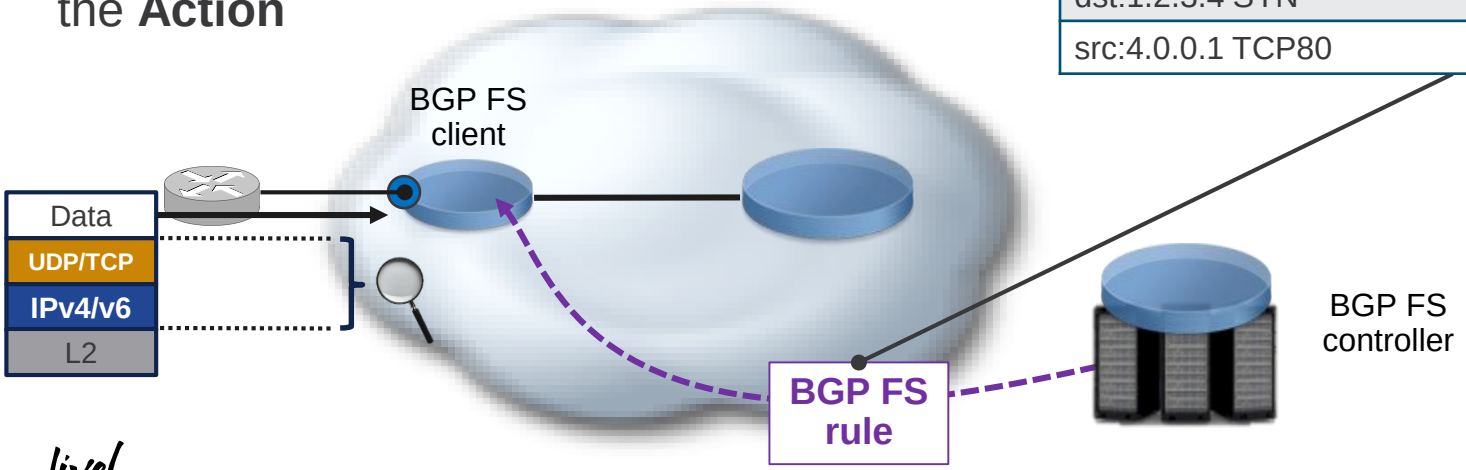


# Introduction: Rule is Description and Action

- BGP is used to program remotely a **rule** made of:
  - A traffic description (v4/v6 L3/L4)
  - An action
- Traffic received on client (ingress only today) matching the **Description** will be applied the **Action**

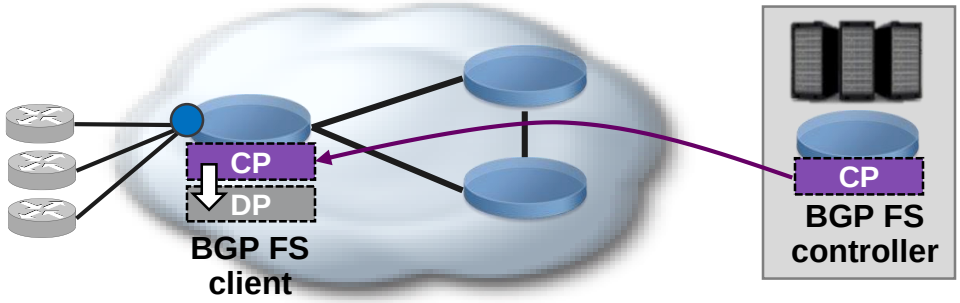
| Traffic Description | Action |
|---|---|
| dst:2001:4:5::23/128 | redirect-in-VRF Dirty |
| UDP:123 Size: 800-1500 | rate-limit 0 bps |
| dst:1.2.3.4 SYN | redirect-to-IP 20.2.3.4 |
| src:4.0.0.1 TCP80 | mark DSCP ef |



# BGP FlowSpec Components

## Controller

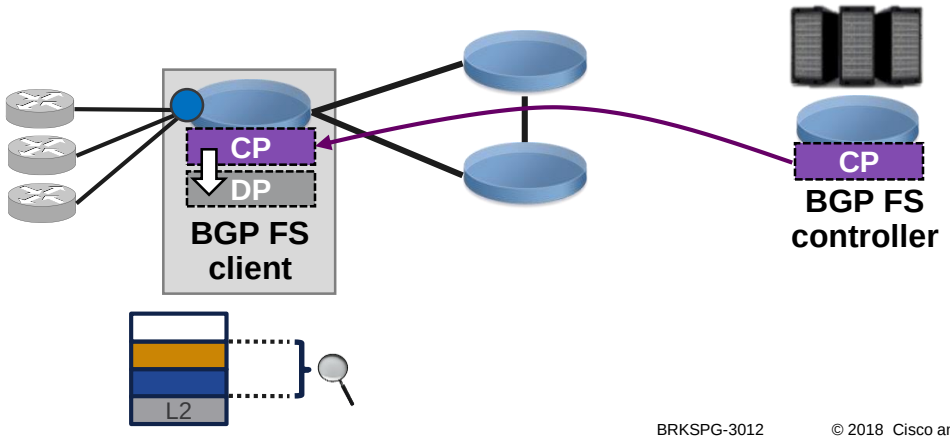
- Injects rules remotely in the clients
- Needs to implement **Control Plane** (CP) at the minimum
- Examples of BGP FS Controllers:
  - **router** (ASR9000, CRS, NCS 6000, XR 12000, …)
  - **server** (ExaBGP, YABGP, Open Day Light, Arbor SP, …)
  - **virtual router** (XRv 9000)



# BGP FlowSpec Components

## Client

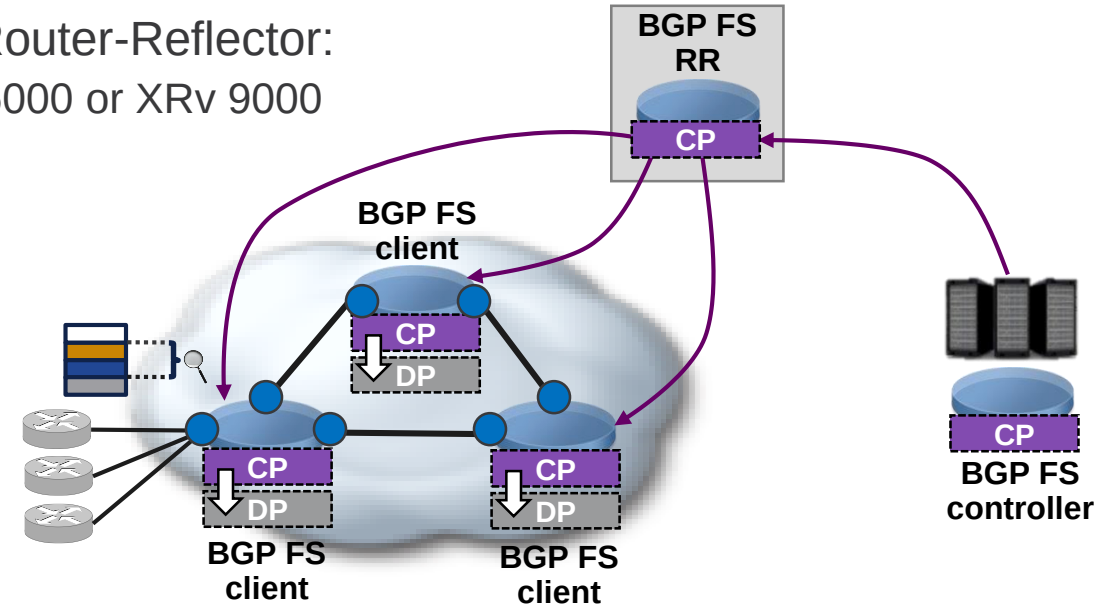
- Receives rules from Controller(s) and programs the match/actions in hardware
- Needs to implement both **Control Plane** (CP) and **Data Plane** (DP)
- Examples of BGP FS Clients:
  - **router** (ASR 9000, CRS, NCS 6000, ASR 1000, CSR 1000v…)



# BGP FlowSpec Components

## Route-Reflector

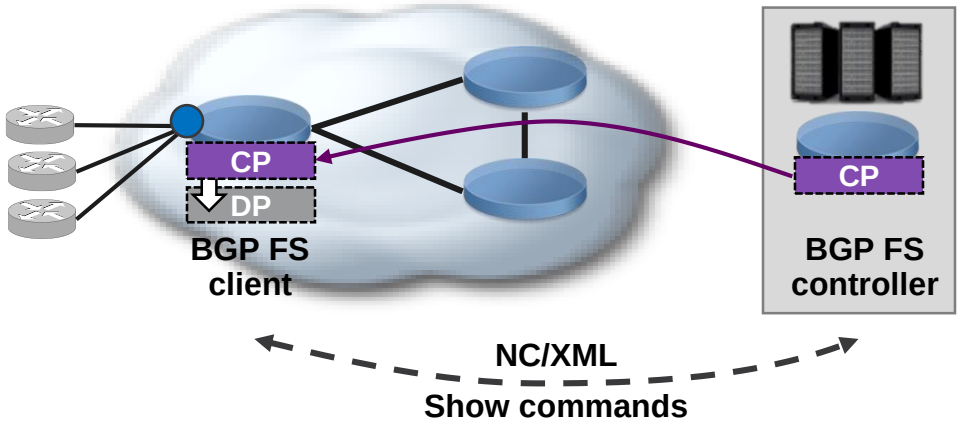
- Receives rules from Controller(s) and distributes them to Clients
- Usually **Control Plane** only, doesn't (need to) program the rules locally
- Examples of BGP FS Router-Reflector:
  - ASR 9000, CRS, NCS 6000 or XRv 9000
  - ASR 1000, CSR 1000v



# BGP FlowSpec

## Uni-Directional

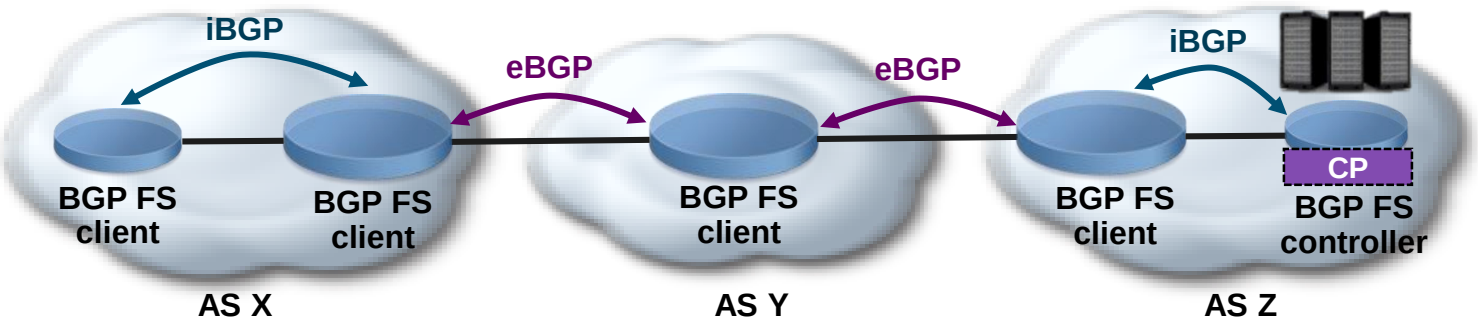
- BGP FS is not bi-directional
- One way arrow from Controller to Client → no feedback loop
- Need other mechanism to collect counters / stats and measure the impact of the rule on traffic



# BGP FlowSpec Session

## Internal / External

- BGP FlowSpec follows the same rules than “traditional” BGP
  - Rules received from eBGP are sent to other eBGP peers
  - Rules received from eBGP are sent to iBGP peers
  - Rules received from iBGP are sent to eBGP peers
  - Rules received from iBGP are not sent to other iBGP peers unless the router is configured as a route-reflector



# BGP FlowSpec Protocol Description

# RFC 5575

## Dissemination of Flow Specification Rules

- Why using BGP?
  - Simple to extend by adding a new NLRI
    - MP_REACH_NLRI / MP_UNREACH_NLRI
  - Already used for every other kind of technology
    - IPv4, IPv6, VPN, Multicast, Labels, MAC addresses, EVPN, …
  - Point to multipoint with Route-Reflectors
  - Inter-domain support
  - Networking engineers and architects understand perfectly BGP
- Why not Openflow or direct NetConf to the router ?
  - Strong framework exists with RR architecture, policies, HA, LLGR
  - Data can be spread at scale and beyond the AS boundaries

# RFC 5575

## Dissemination of Flow Specification Rules: Traffic Matching

- NLRI defined (AFI=1, SAFI=133) to describe the traffic of interest
  1. Destination IP Address
  2. Source IP Address
  3. IP Protocol
  4. Port
  5. Destination port
  6. Source Port
  7. ICMP Type
  8. ICMP Code
  9. TCP Flags
  10. Packet length
  11. DSCP
  12. Fragment

| Type | Length |
|---|---|
| Address Family Identifier (AFI) | 2 octets |
| Subsequent Address Family Identifier (SAFI) | 1 octet |
| Length of Next Hop Network Address | 1 octet |
| Network Address of Next Hop | Variable |
| Reserved | 1 octet |
| Network Layer Reachability Information (NLRI) | Variable |

The MP_REACH_NLRI – RFC 4760

# RFC 5575

## Dissemination of Flow Specification Rules: Traffic Matching

**IPv4**

| Version | IHL | ToS | Total Length | |
|---|---|---|---|---|
| Identification | | | Flags | Frag Offset |
| TTL | Protocol | Header Checksum | | |
| Source Address | | | | |
| Destination Address | | | | |
| Options | | | | Padding |

**Not matched:**

- MPLS labels number
- MAC address
- L5-7 data like
  - HTTP URL
  - Cookie
  - DNS requests…

**TCP**

| Source Port | | | Destination Port |
|---|---|---|---|
| Sequence Number | | | |
| Ack Number | | | |
| H lgh | Res | C bit | Window |
| Checksum | | | Urgent |
| Options | | | |
| Data | | | |

**UDP**

| Source Port | Destination Port |
|---|---|
| Length | Checksum |
| Data | |

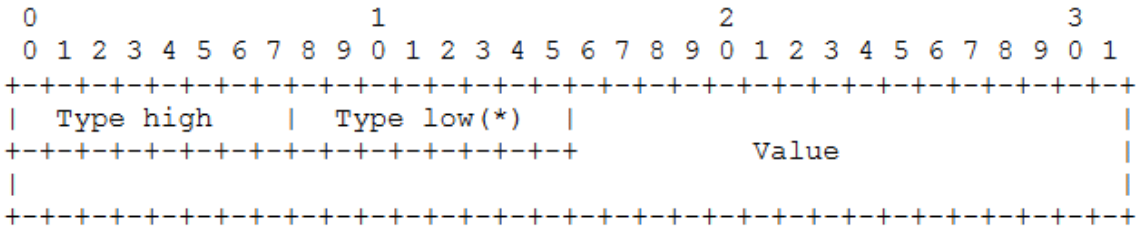
**ICMP**

| Type | Code | Checksum |
|---|---|---|
| Quench | | |
| Data | | |

# RFC 5575

## Dissemination of Flow Specification Rules: Traffic Actions

- Traffic Action is defined in extended-communities (RFC4360)




| Type | Description | Encoding |
|---|---|---|
| 0x8006 | Traffic-rate | 2 bytes ASN; 4 bytes as float |
| 0x8007 | Traffic-action | Bitmask |
| 0x8008 | Redirect | 6 bytes RT (Route Target) |
| 0x8009 | Traffic-marking | DSCP Value |

# RFC 7674

## Clarification of the Flowspec Redirect Extended Community

- Following Redirect actions are supported since IOS XR 5.2.0

| Type | Description | Encoding |
|---|---|---|
| 0x8008 | Redirect 2B ASN RT | 2 Octets ASN , 4 Octets Value |
| 0x8108 | Redirect IPv4 RT | 4 Octets IPv4 address, 2 Octets Value |
| 0x8208 | Redirect 4B ASN RT | 4 Octets ASN, 2 Octets Value |

Note: the IPv4 RT ( a.b.c.d : value ) is not the the redirect to IP action

# IETF Drafts

## Extensions for RFC5575: IETF Drafts

- On top of the RFC implementation, IOS **XR supports**:
  - IPv6 extensions: *draft-ietf-idr-flow-spec-v6-03*
  - Redirect IP extension: *draft-simpson-idr-flowspec-redirect-02*
  - IBGP extension: *draft-ietf-idr-bgp-flowspec-oid-01*
  - Persistence Support: *draft-uttaro-idr-bgp-persistence-02* (in IOS XR5.2.2)
  - HA/NSR Support
  - Max-prefix

# IETF Drafts

## Extensions for RFC5575: IETF Drafts

- On top of the former list, IOS **XE supports**:
  - draft-ietf-idr-flowspec-interfaceset-03

  New Extended community to inform remote router where (interface) to apply the rule

  Not supported on XR

For Your Reference

# IETF Drafts

## Extensions for RFC5575: Unsupported IETF Drafts

- Other drafts are under work in the IDR group but not supported in IOS XR:
  - Carrying Label Information for BGP FlowSpec: *draft-ietf-idr-bgp-flowspec-label-01*
  - Dissemination of Flow Specification Rules for L2 VPN: *draft-ietf-idr-flowspec-l2vpn-05*
  - BGP Flow Specification Filter for MPLS Label: *draft-ietf-idr-flowspec-mpls-match-01*
  - BGP Flow Specification Packet-Rate Action: *draft-ietf-idr-flowspec-packet-rate-01*
  - Flowspec Indirection-id Redirect: *draft-ietf-idr-flowspec-path-redirect-01*
  - Dissemination of Flow Specification Rules: *draft-ietf-idr-rfc5575bis-01*
  - Inter-provider Propagation of BGP Flow specification Rules: *draft-bashir-idr-inter-provider-flowspec-actions-00*
  - Populate to FIB Action for FlowSpec: *draft-li-idr-flowspec-populate-to-fib-00*

# Cisco Routers BGP FS Implementation


| Platform Hardware | Support in Data Plane |
|---|---|
| ASR 9k – Typhoon LC (MOD80/160, 24-36x10G, 1-2x100G) | XR 5.2.0 |
| ASR 9k – SIP700 | XR 5.2.2 |
| ASR 9001(-S) | XR 5.2.2 |
| ASR 9k – Tomahawk (MOD200/400, 4-8-12x100G) | XR 5.3.0 |
| CRS-3 (Taiko) LC (1x100G, 14-20x10G, Flex) | XR 5.2.0 |
| CRS-X (Topaz) LC (4x100G, 40x10G, Flex) | XR 5.3.2 |
| NCS 6000 | XR 5.2.4 / 6.2.2 / roadmap* |
| XRv 9000 | 5.4.0 CP only / DP later |
| NCS 5000 / NCS 5500 | In the roadmap |
| ASR 1000 | IOS XE 3.15 |
| CSR 1000v | IOS XE 3.15 |
| **NCS 5500 (Jericho+ w/ eTCAM)** | **XR 6.5.1** |

Note: IOS XE introduced the support of BGP FS in 3.15 (but not as a controller role)

For Your Reference

# Cisco IOS XR Routers BGP FS Implementation

| NLRI type | Match fields | Value input method | XR PI | ASR9000 | CRS | NCS6000 |
|---|---|---|---|---|---|---|
| **Type 1** | IPv4 Destination address | Prefix length | ✓ | ✓ | ✓ | ✓ |
| **Type 2** | IPv4 Source address | Prefix length | ✓ | ✓ | ✓ | ✓ |
| **Type 3** | IPv4 protocol | Multi value range | ✓ | ✓ | ✓ | ✓ |
| **Type 4** | IPv4 source or destination port | Multi Value range | ✓ | ✓ | ✓ | ✓ |
| **Type 5** | IPv4 destination port | Multi Value range | ✓ | ✓ | ✓ | ✓ |
| **Type 6** | IPv4 Source port | Multi Value range | ✓ | ✓ | ✓ | ✓ |
| **Type 7** | IPv4 ICMP type | Multi value range | ✓ | ✓ | ✓ | ✓ |
| **Type 8** | IPv4 ICMP code | Multi value range | ✓ | ✓ | ✓ | ✓ |
| **Type 9** | IPv4 TCP flags (2 bytes include reserved bits) | Bit mask | ✓ | Only Lower byte Reserved and NS bit not supported | Only Lower byte Reserved and NS bit not supported | Only Lower byte Reserved and NS bit not supported |
| **Type 10** | IPv4 Packet length | Multi value range | ✓ | ✓ | ✓ | ✓ |
| **Type 11** | IPv4 DSCP | Multi value range | ✓ | ✓ | ✓ | ✓ |
| **Type 12** | IPv4 fragmentation bits | Bit mask | ✓ | Only indication of fragment | ✓ | ✓ |

For Your Reference

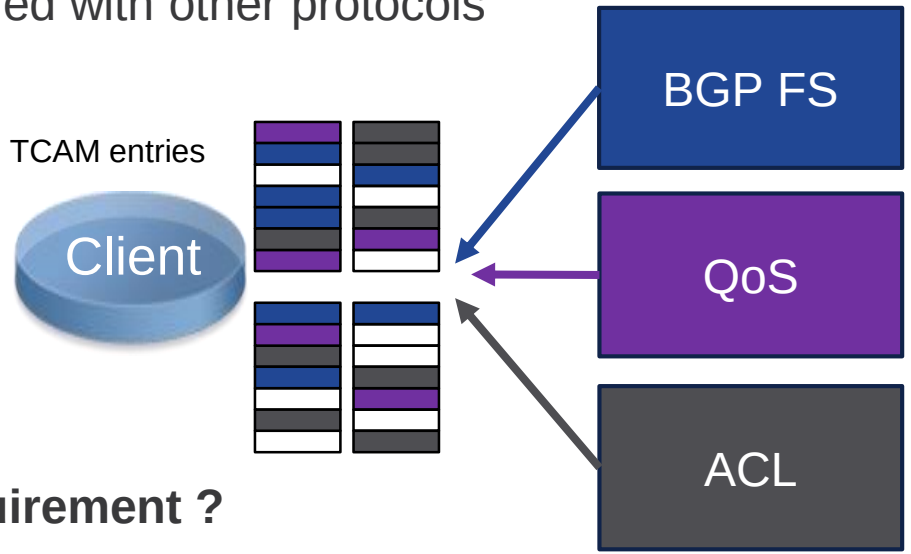
# Cisco IOS XR Routers BGP FS Implementation

| NLRI type | Match fields | Value input method | XR PI | ASR9000 | CRS | NCS6000 |
|---|---|---|---|---|---|---|
| **Type 1** | IPv6 Destination address | Prefix length | ✓ | ✓ | ✓ | ✓ |
| **Type 2** | IPv6 Source address | Prefix length | ✓ | ✓ | ✓ | ✓ |
| **Type 3** | IPv6 Next Header | Multi value range | ✓ | ✓ | ✓ | ✓ |
| **Type 4** | IPv6 source or destination port | Multi Value range | ✓ | ✓ | ✓ | ✓ |
| **Type 5** | IPv6 destination port | Multi Value range | ✓ | ✓ | ✓ | ✓ |
| **Type 6** | IPv6 Source port | Multi Value range | ✓ | ✓ | ✓ | ✓ |
| **Type 7** | IPv6 ICMP type | Multi value range | ✓ | ✓ | ✓ | ✓ |
| **Type 8** | IPv6 ICMP code | Multi value range | ✓ | ✓ | ✓ | ✓ |
| **Type 9** | IPv6 TCP flags (2 bytes include reserved bits) | Bit mask | ✓ | Only Lower byte Reserved and NS bit not supported | Only Lower byte Reserved and NS bit not supported | Only Lower byte Reserved and NS bit not supported |
| **Type 10** | IPv6 Packet length | Multi value range | ✓ | ✓ | ✓ | ✓ |
| **Type 11** | IPv6 Traffic Class | Multi value range | ✓ | ✓ | ✓ | ✓ |
| **Type 12** | Reserved | N/A | N/A | N/A | N/A | N/A |
| **Type 13** | IPv6 Flow Based (20 bytes) | Multi value range | ✗ | ✗ | ✗ | ✗ |

# IOS XR Implementation

## Resource Usage

- BGP Flowspec entries are stored in TCAM
  - Up to **3000 simple rules per line card** (limited on the controller today)
- Resource is finite and shared with other protocols

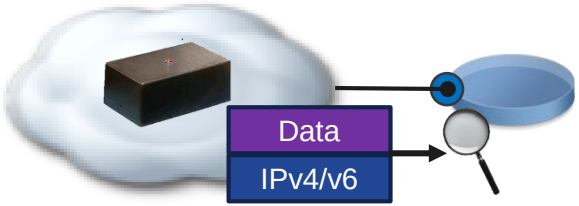


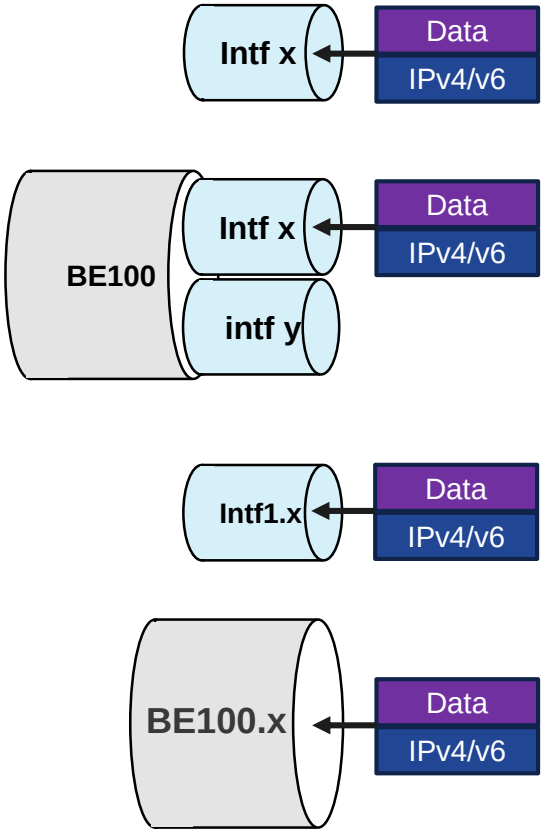

- **What is YOUR scale requirement ?**

# IOS XR Implementation

## Application on Interface

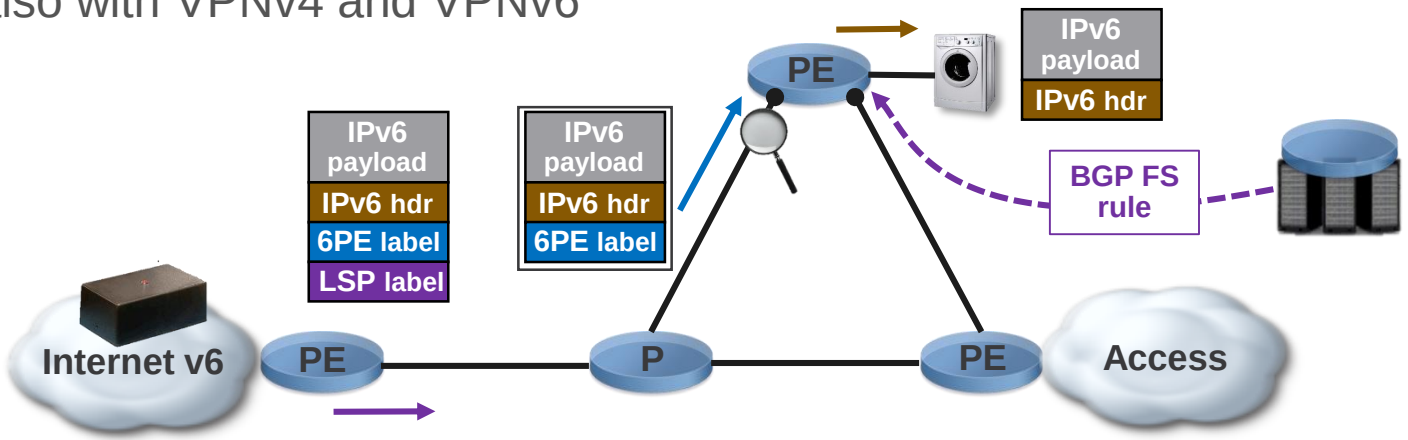
- In current implementation, rules are applied:
  - in ingress
  - on physical or logical interfaces (Link-bundles and dot1q)
  - but not on tunnels
  - with IPv4 and IPv6 traffic



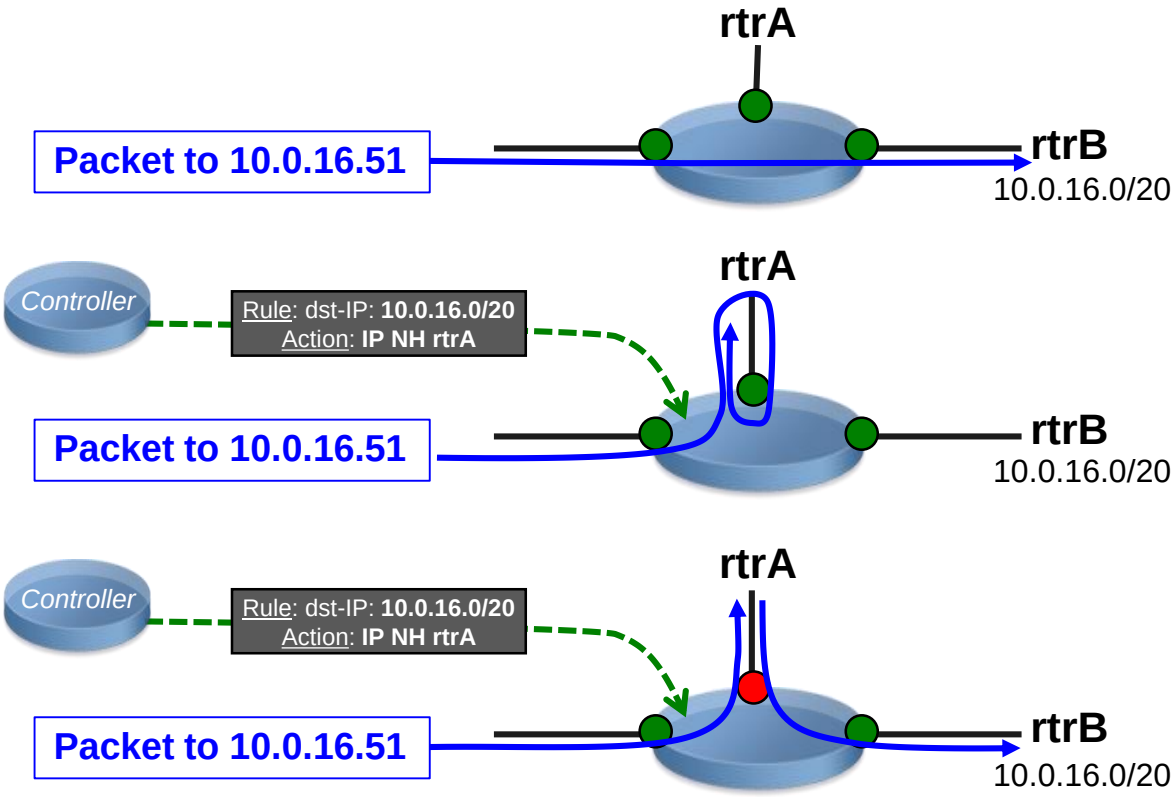



# BGP FlowSpec with 6PE

- Network with legacy devices not supported dual-stack are leveraging 6PE to transport IPv6 over MPLS
- When packets are received on PE routers, they are encapsulated in MPLS labels
- ASR9000 will be able to apply BGP FS rules on the P-PE interface receiving 6PE labelled packets and match in the IPv6 Header (L3 and L4)
- Works also with VPNv4 and VPNv6



# IOS XR Interface Disabled




BGP FlowSpec Enabled

BGP FlowSpec Disabled

BGP FS is applied to the whole router but can be activated or deactivated on particular interfaces via CLI configuration. Particularly useful in Distributed DDoS mitigation architecture.

# IOS XE Implementation

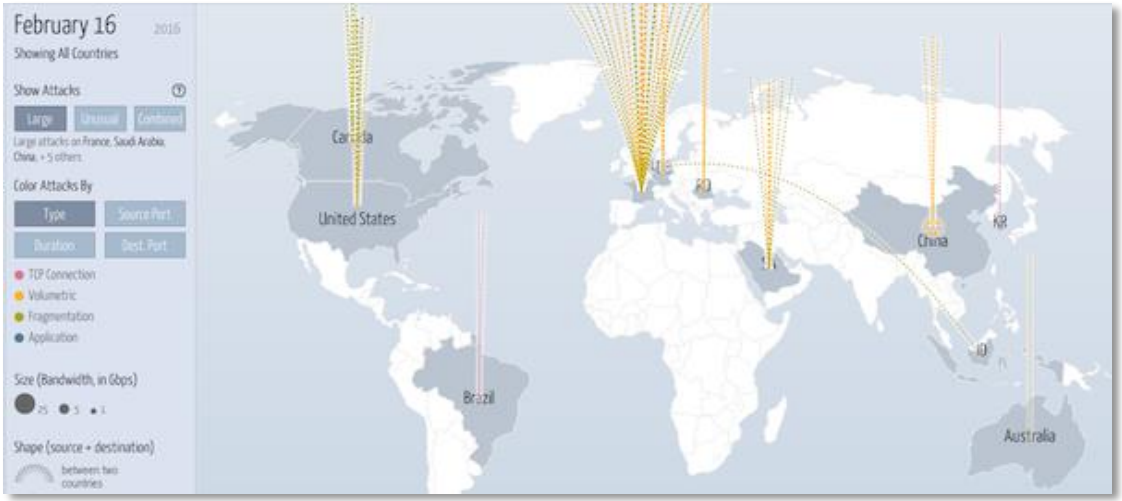
- Implementation on IOS XE is very similar than the IOS XR one (sharing a lot of code), hence the features are almost identical but with a different scale support

| | ASR1000 | CSR1000v | ISR4400 |
|---|---|---|---|
| Max rules per system | 4000 | 250 | 4000 |
| Max rules per VRF | 1000 | 32 | 250 |

# Use-cases:
# DDoS Mitigation

# DDoS Attacks

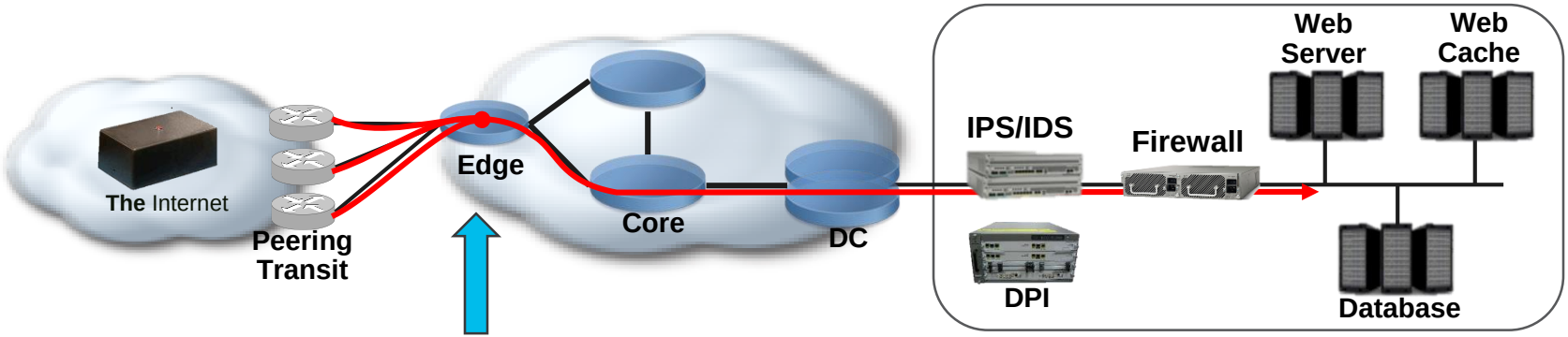
- No longer necessary to explain the risk
  - Distributed Denial of Service (DDoS) is a lucrative activity for attackers
  - ISP, Hosting Services, Enterprises: it can jeopardize your business
    Everyone is at risk
  - 2017:
    - More sophisticated
    - Less volumetric
    - But still very high




Source: [http://www.digitalattackmap.com/](http://www.digitalattackmap.com/)

# DDoS Attacks

- Denial of Service attacks are of different natures:
  - Application-layer attacks
    - Detected and handled by Firewalls, IDS or at the Server level
  - Volumetric attacks (including Protocols attacks)
    - Can NOT be mitigated in datacenter or server farm (too late)
    - Should be handled in the backbone or at the border

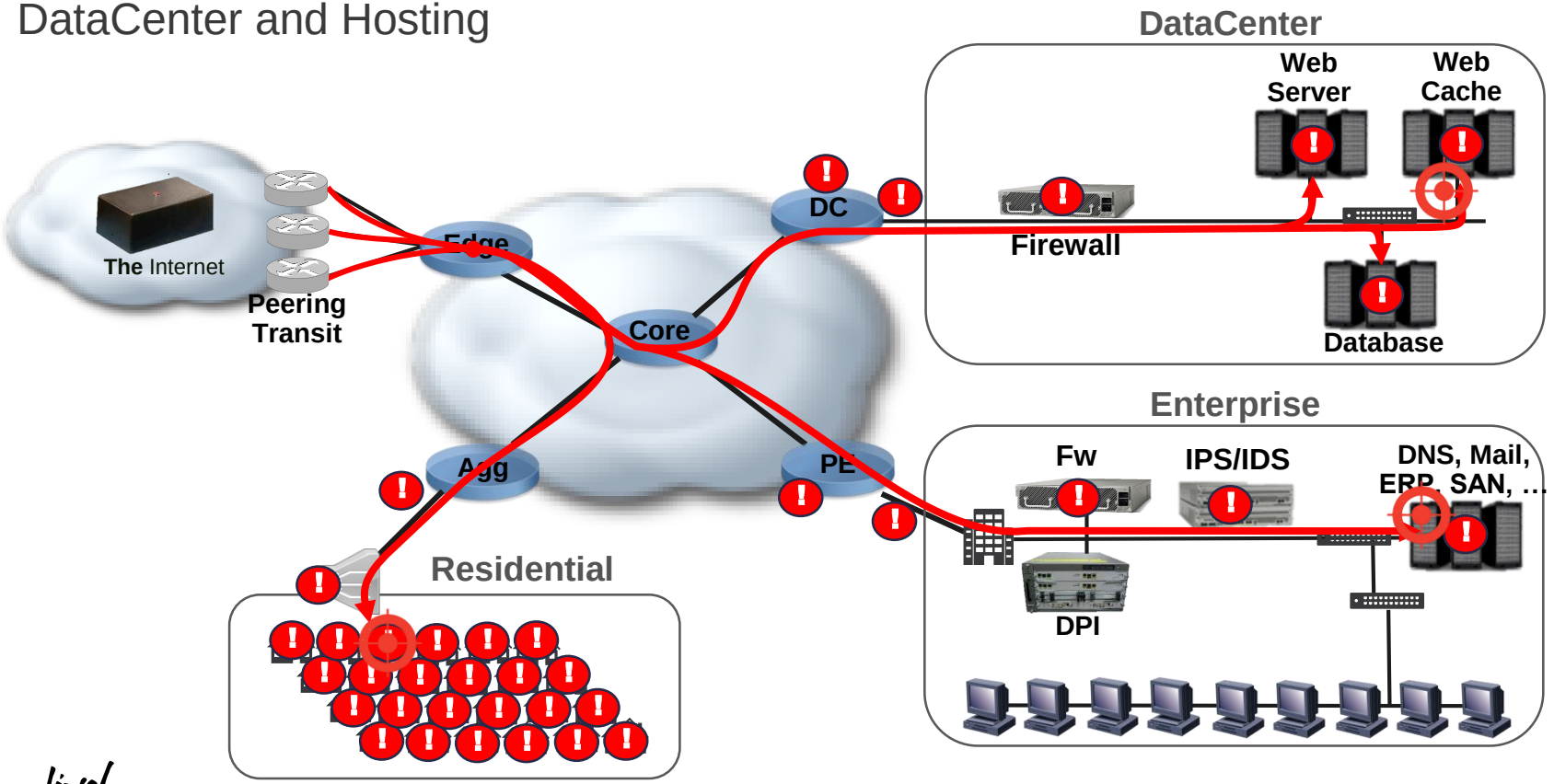


# DDoS Attacks Mitigation

- BGP FS was initially designed with DDoS Mitigation use-case in mind
- Distributed attack received from all transit and peering points
- We can use a mitigation system in a ASR9000/VSM card or an appliance connected to your IOS-XR router
- We differentiate arbitrarily three DDoS attack families:
  - Stateless Amplification
  - Stateless L3 / L4 / others
  - Stateful / up-to-L7 on application resources

# Different Business, Different Targets

## DataCenter and Hosting

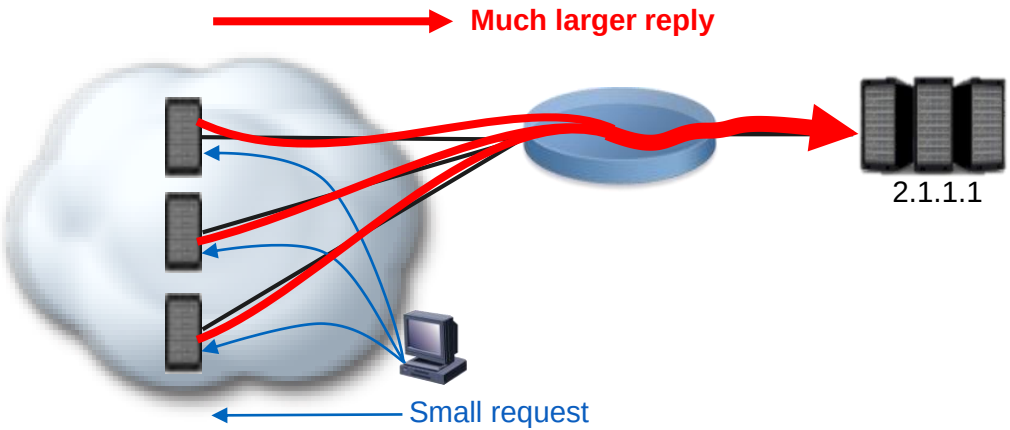


# Use-cases:
# DDoS Mitigation
# Amplification Attacks

# DDoS Mitigation with BGP FS

## Amplification Attacks 101

- Stateless attacks are not using a full handshake and are based on spoofed source addresses
- Amplification attacks using vulnerable protocols on high bandwidth servers




- DNS
- NTP
- CharGen
- SNMP
- SSDP
- RIPv1
- Port Mapper

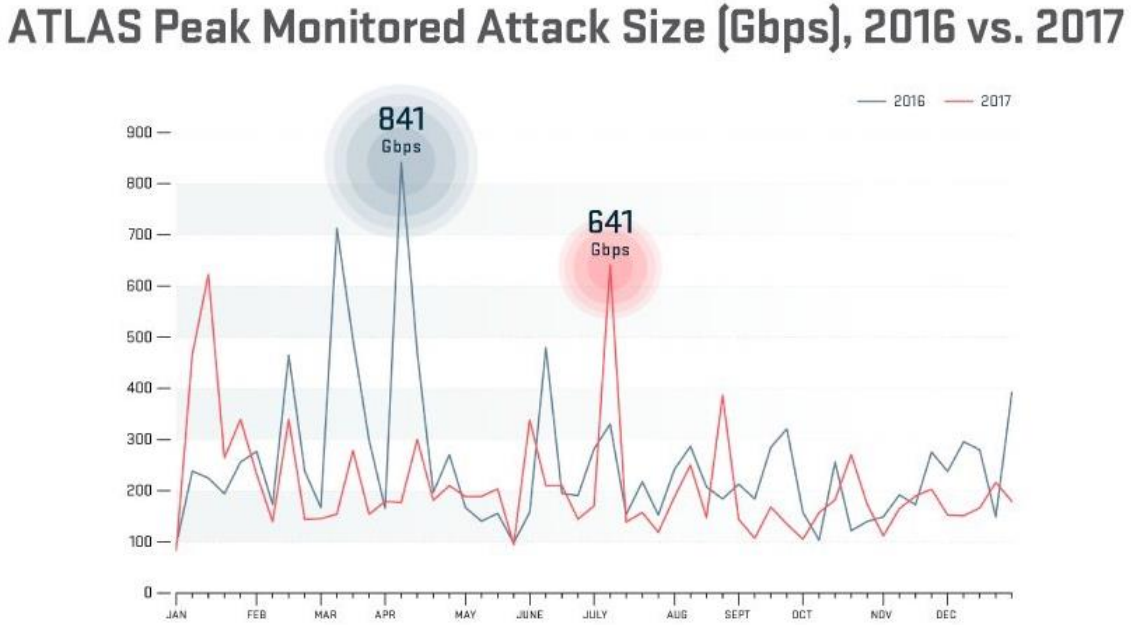
# DDoS Mitigation with BGP FS

## Amplification Attacks Always Trendy

- 2015/2016 → raise of Amp attacks
- 2016/2017 → botnets (Mirai, Satori, …)

- Victims
  - #1 Online Gaming
  - #2 Criminal demonstration
  - #3 Extortion

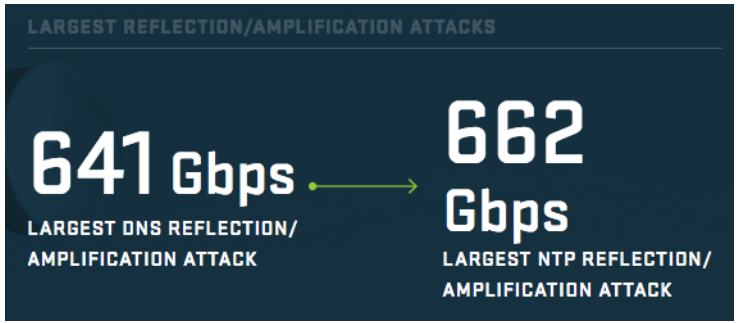
Source: Arbor WISR 2018



# DDoS Mitigation with BGP FS

## Amplification Attacks Always Trendy

- But Amplification attacks didn't disappear
- UDP Frag, DNS and NTP still in the top 3




Source: Arbor WISR 2018

Source: Akamai State of the Internet 2017

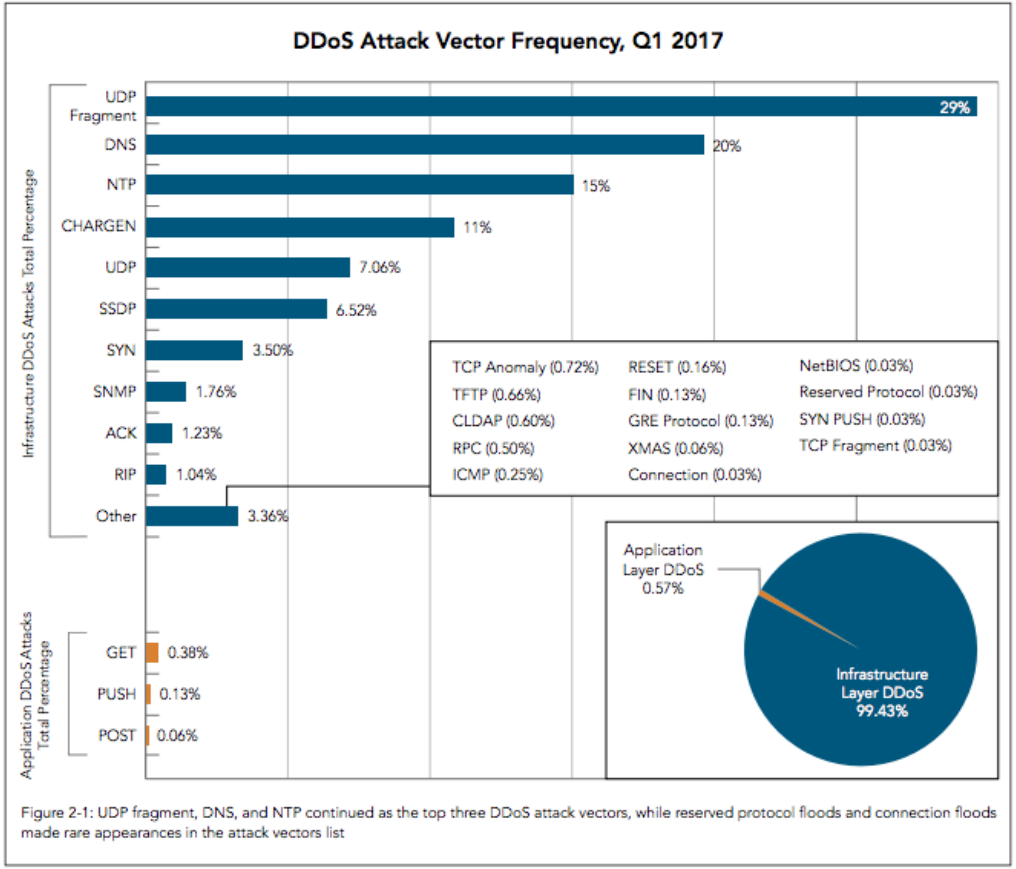
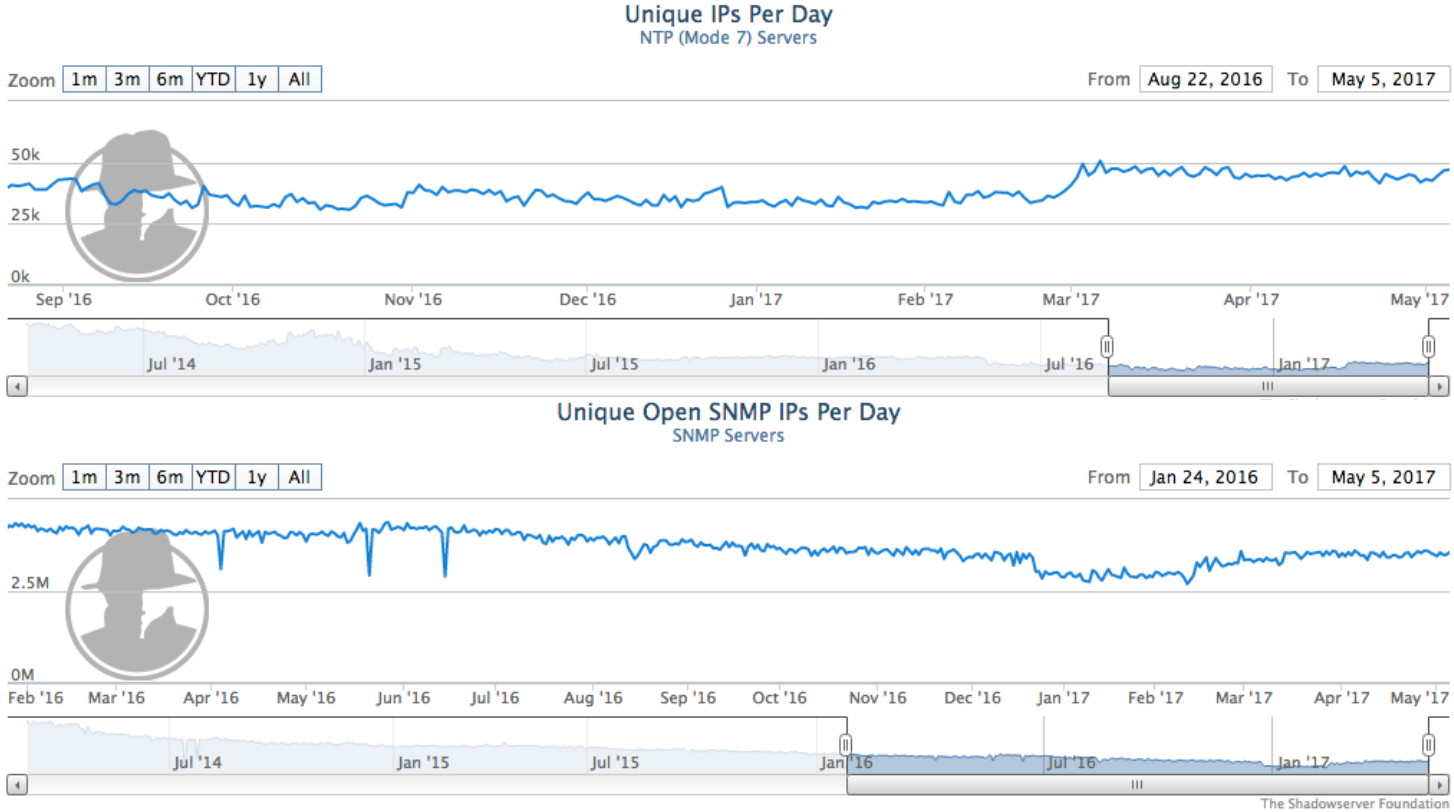



Figure 2-1: UDP fragment, DNS, and NTP continued as the top three DDoS attack vectors, while reserved protocol floods and connection floods made rare appearances in the attack vectors list

# DDoS Mitigation with BGP FS

## Amplification Attacks Always Trendy

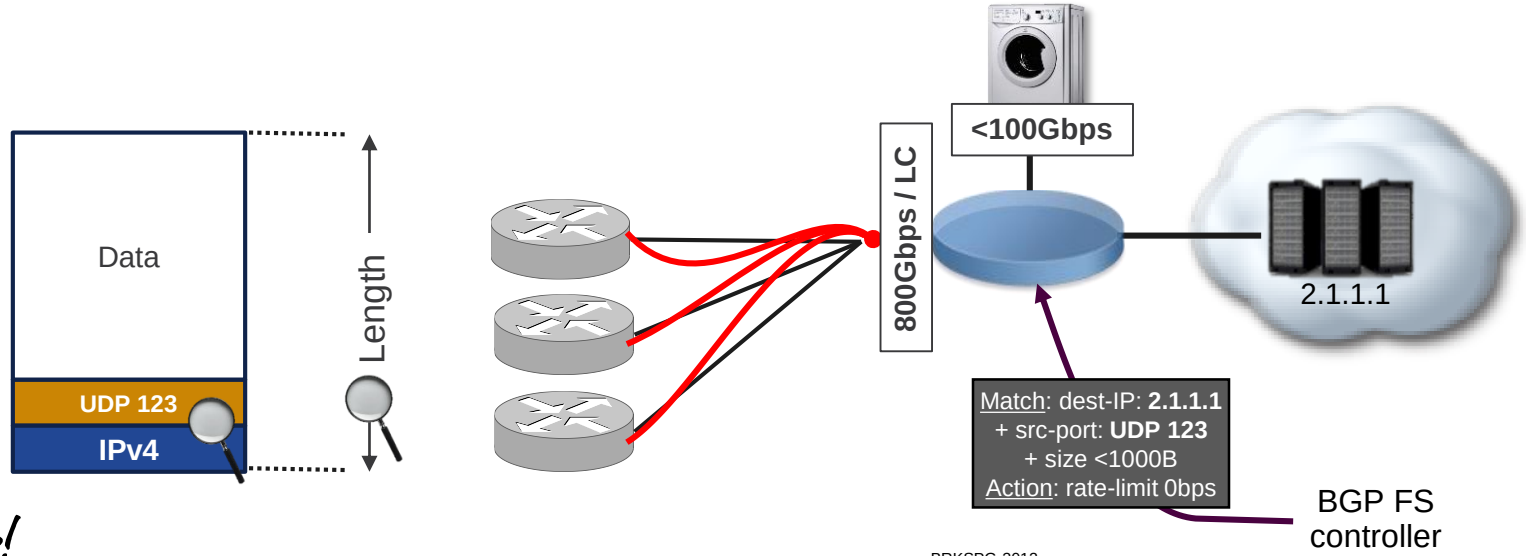



Source: https://www.shadowserver.org/

# DDoS Mitigation with BGP FS

## Rate-limiting / Filtering Amplification Attacks

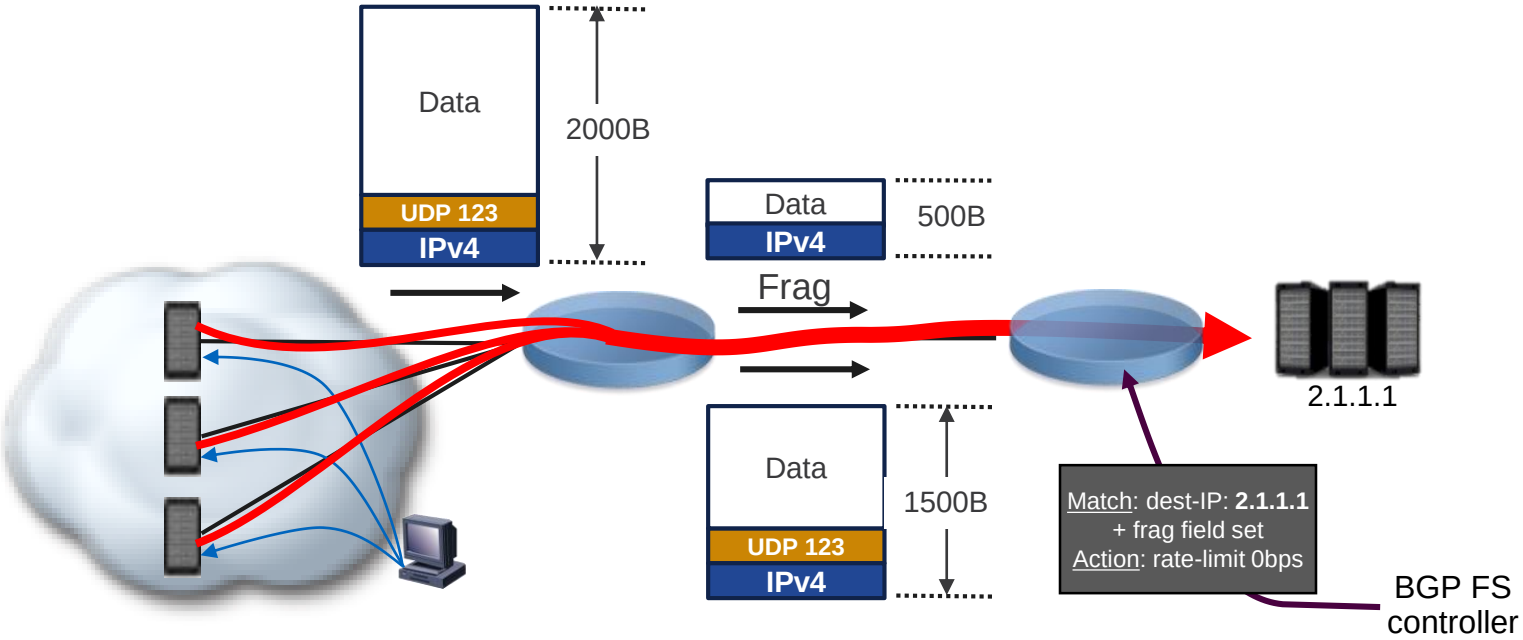
- Amplification attacks, example NTP
  - Don’t need to be handled by a “sophisticated” scrubbing system to be mitigated
  - Can be filtered at the router line card level → much higher performance
  - Identified by precisely matching the traffic pattern and filtered at the edge router level



# DDoS Mitigation with BGP FS

## Fragments

- Very often seen with amplification attacks (packets larger than the path MTU)

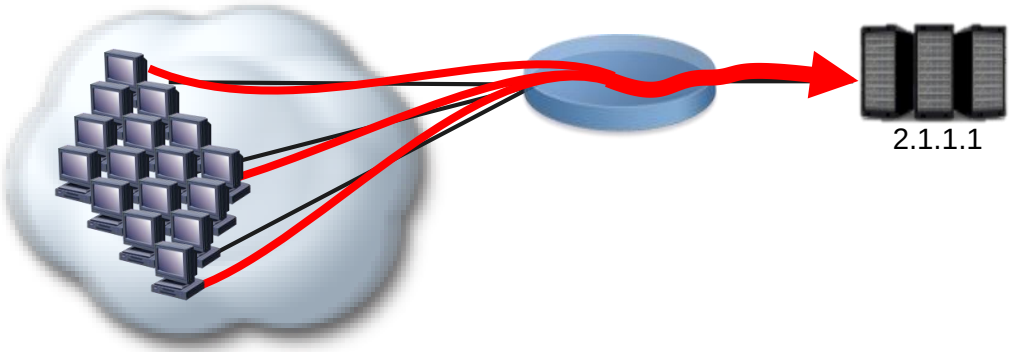


# Use-cases: DDoS Mitigation L3/L4 Attacks

# DDoS Mitigation with BGP FS

## Rate-limiting / Filtering Stateless Attacks: L3/L4 Protocol Attacks

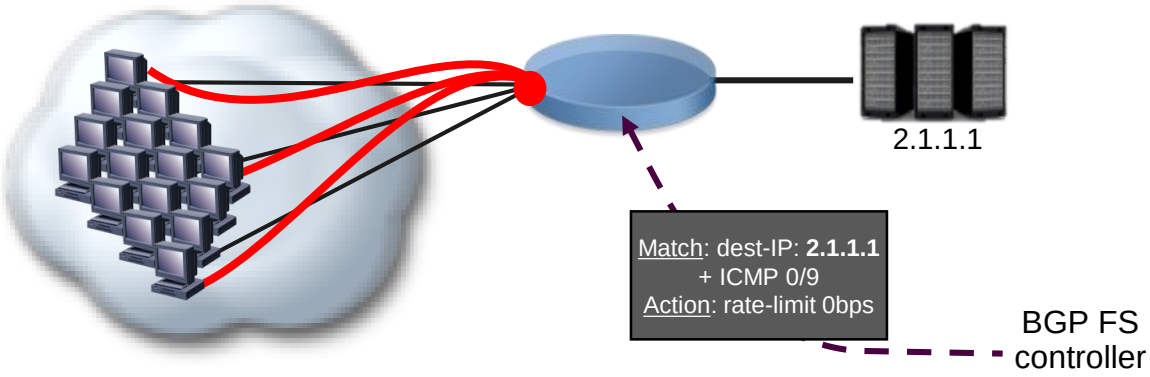
- Generic family covering non-amplified stateless streams like ICMP flood
- Source address could be forged or not (botnet members are corrupted hosts)



# DDoS Mitigation with BGP FS

## Rate-limiting / Filtering Stateless Attacks: L3/L4 Protocol Attacks

- L3/L4 attacks can be also filter at the edge router via BGP FS
- Same principles than previous use-case

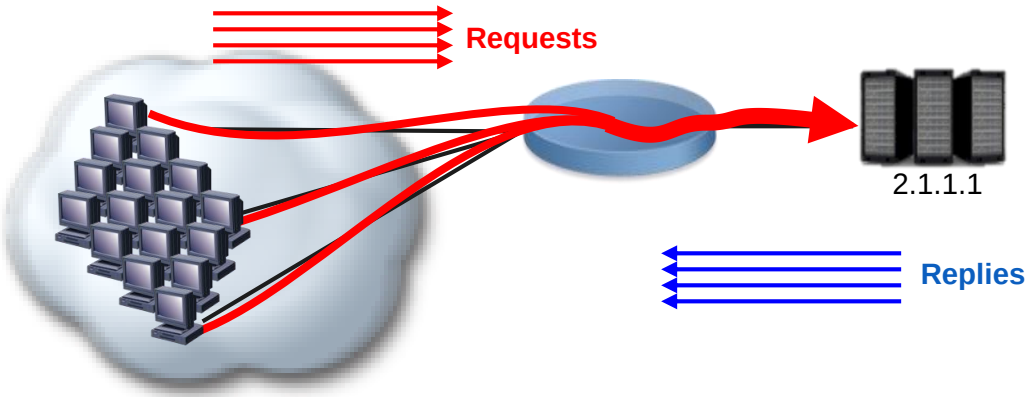


# Use-cases: DDoS Mitigation Stateful Attacks

# DDoS Mitigation with BGP FS

## Addressing Stateful Attacks

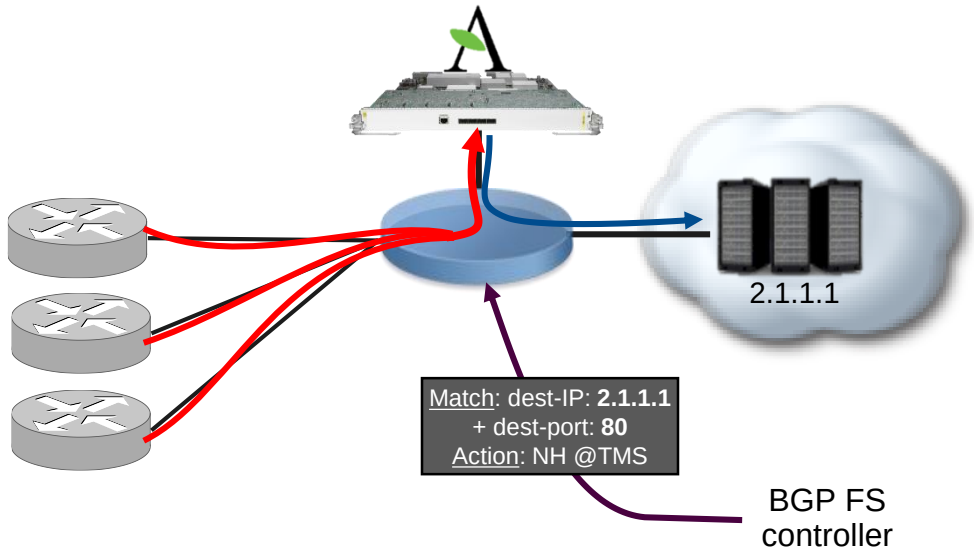
- More advanced attacks using Botnets or even real users (LOIC) needs to be addressed differently by a specific scrubbing device. Examples:
  - HTTP: bots mimicking the behavior of a real web browser
  - TCP SYN
  - SSL
  - SIP
  - …



# DDoS Mitigation with BGP FS

## Addressing Stateful Attacks

- BGP FlowSpec will be used to program a different action here
  - Diversion to a next-hop address
  - Diversion to a different VRF

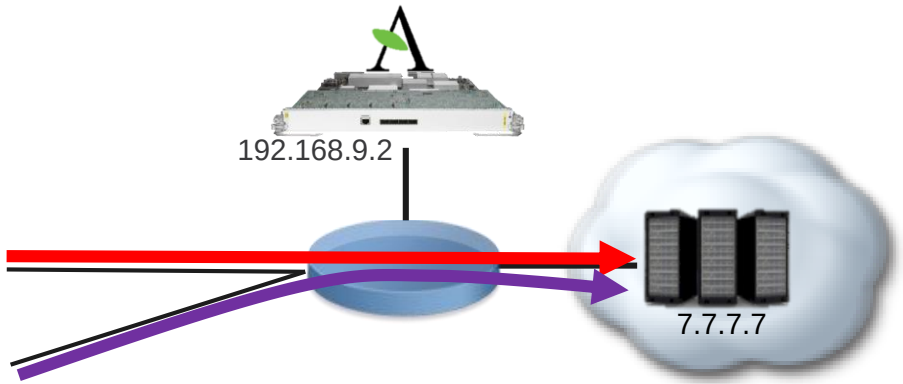


# DDoS Mitigation Demo with Arbor Solution

# Demo

## Rate-limiting and Redirect Attacks Traffic w/ BGP FlowSpec

- Edited version of a recording from Tomas Sundstrom
- Using Arbor TMS as a controller and ASR9000 as client

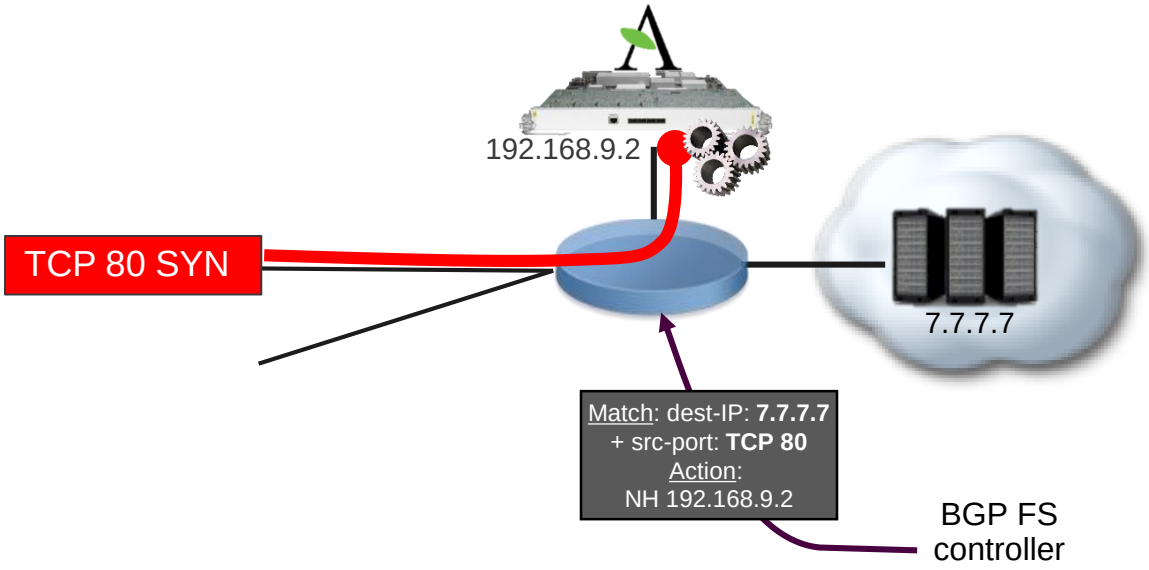



- Detection of the attack itself is out of the scope of this short demo

# Demo

## Rate-limiting and Redirect Attacks Traffic w/ BGP FlowSpec

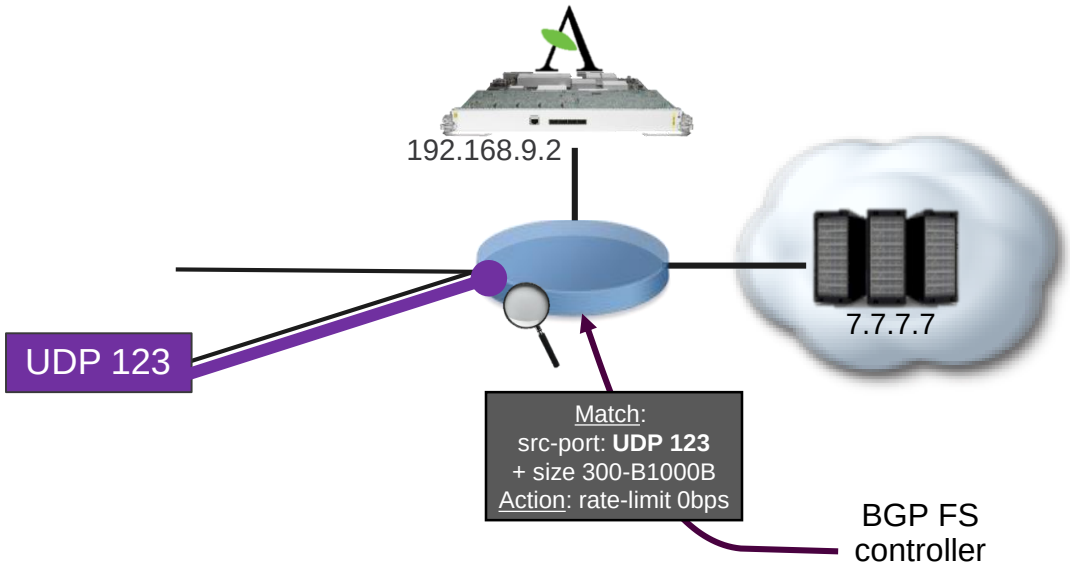
- First attack is identified as a TCP 80 SYN with very large packet size
- We will use BGP FS to divert the TCP 80 traffic targeted to 7.7.7.7 into the TMS



# Demo

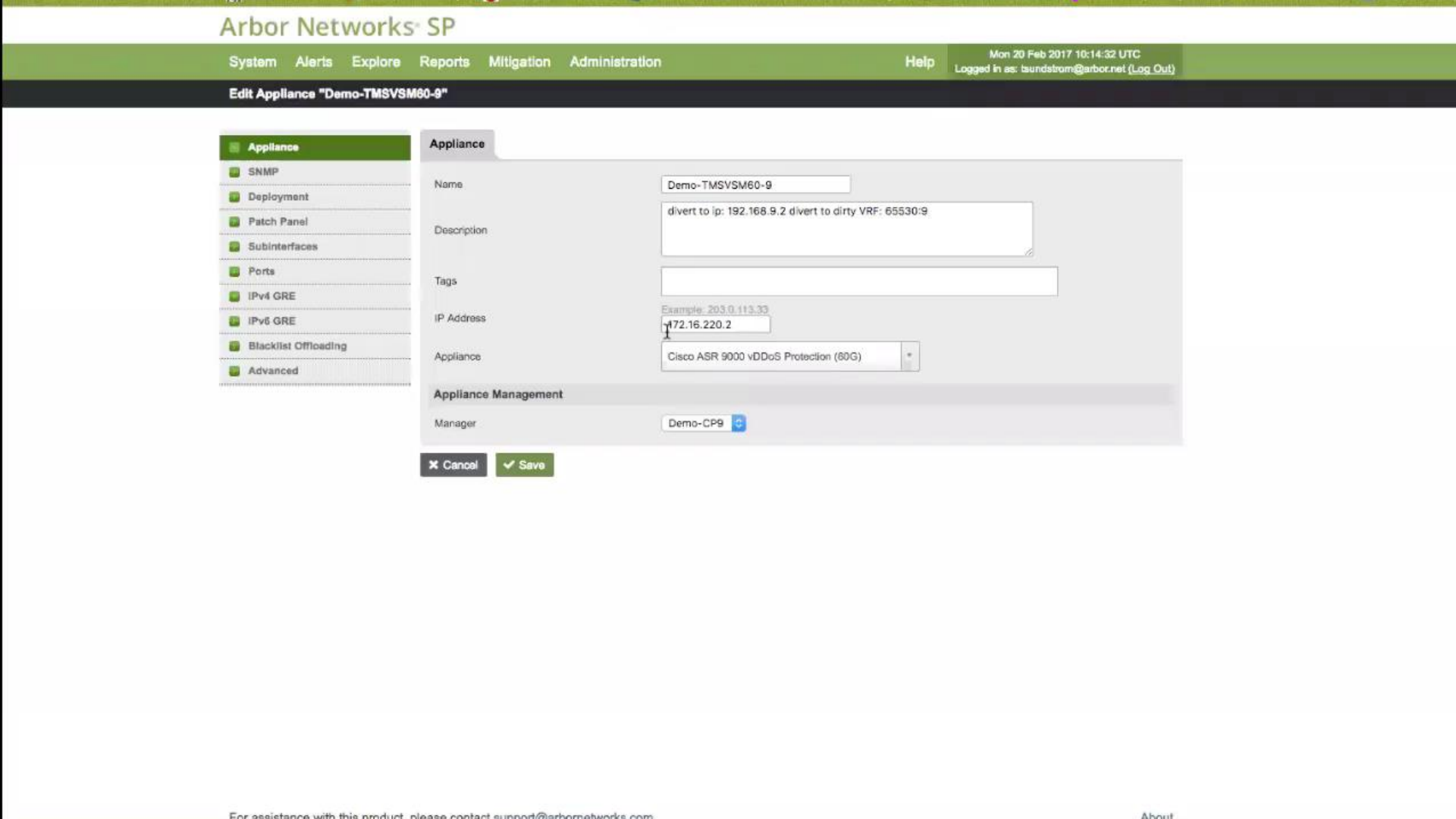
## Rate-limiting and Redirect Attacks Traffic w/ BGP FlowSpec

- Second attack is identified as a NTP Amplification (abnormal packet size)
- We will use BGP FS to drop UDP 123 packets from 300 to 1000 bytes



# Demo
# Rate-limiting and Redirect Attacks w/ BGP FlowSpec

http://bit.ly/2rYSKY9


Arbor Networks SP
System
Alerts
Explore
Reports
Mitigation
Administration
Help
Mon 20 Feb 2017 10:14:32 UTC
Logged in as: tsundstrom@arbor.net (Log Out)
Edit Appliance "Demo-TMSVSM60-9"
Appliance
SNMP
Deployment
Patch Panel
Subinterfaces
Ports
IPv4 GRE
IPv6 GRE
Blacklist Offloading
Advanced
Appliance
Name
Demo-TMSVSM60-9
Description
divert to ip: 192.168.9.2 divert to dirty VRF: 65530:9
Tags
IP Address
Example: 203.0.113.33
172.16.220.2
Appliance
Cisco ASR 9000 vDDoS Protection (60G)
Appliance Management
Manager
Demo-CP9
Cancel
Save
For assistance with this product, please contact support@arbornetworks.com
About

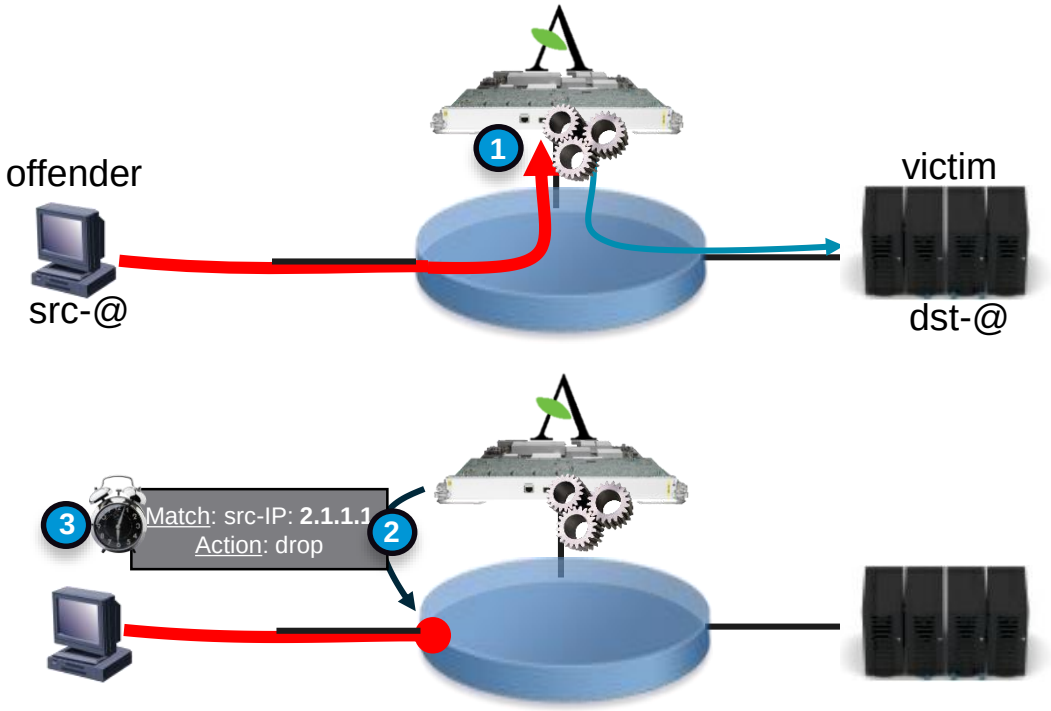
# Arbor SP Solution

## Dynamic Black-list Offload with BGP FlowSpec

1. A countermeasure is activated and detects an offender
2. TMS instructs the ASR9000 via Flowspec to program an ACL for the src-@ or the pair src-@+dst-@

→ For one minute

3. After 1min, the ACL is removed. If the offender is seen by the countermeasure again, ACL will be programmed for 5min, and then 5 min, again and again



- No “drop” in BGP Flowspec actions, just a policer to 0 bps
- In DDoS attack context what could be the benefits of rate-limiting to X bps instead of 0 bps
- X bps will drop packets randomly (legitimate or malicious ones equally), creating difficult troubleshooting situation
- 0 bps is advised

# DDoS Mitigation with BGP FS

## Benefits

- **Single point of control** to program rules in many clients
- **Granularity**: Allows a very precise description/matching of the attack traffic
- Can be used for **both mitigation and diversion** of the attack traffic, without impact the course of the rest of the traffic targeted to the victim
- **Off-Load Mitigation system**: Filtering stateless attacks on the edge route permits mitigation of millions of PPS of dirty traffic while liberating precious CPU cycle on the scrubbing device for more advanced mitigation needs
- Useful but not “Magic”: can not be do much for stateful attacks (Mirai, etc)

# Improving Existing DDoS Mitigation Models

# DDoS Mitigation Models

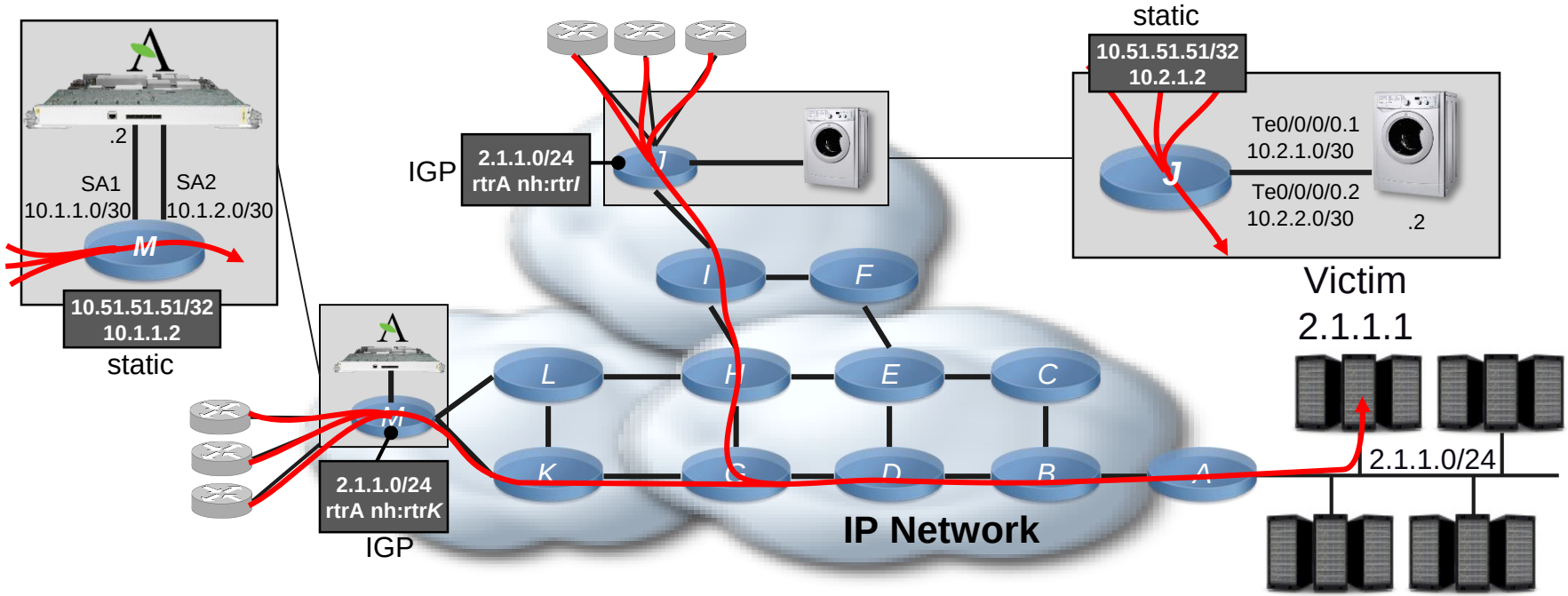
## Network Design

- Several approaches exist in the design of a DDoS mitigation solution
- No real “best practices” in this field, it mainly depends on
  - The topology
  - The protocols and services: IP only, MPLS transport, L2/L3VPN
- They all consist in:
  - Diverting the traffic targeted to the victim to push it into scrubbing devices
  - Performing an analysis of the packets to discriminate legit packets from attack packets
  - Re-injecting the legit traffic into the network
- Following examples are real-case used in very large production networks

# IP-only Network w/ Distributed TMS

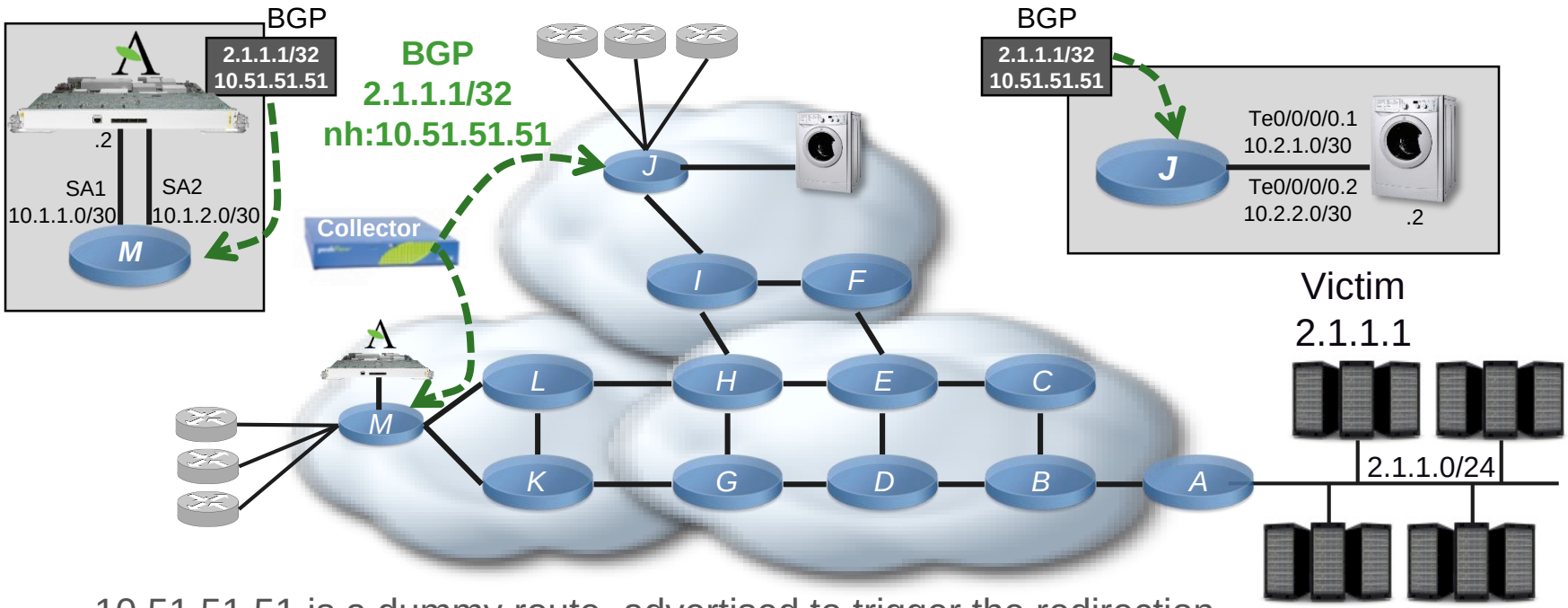
## Currently deployed

- A static route for 10.51.51.51 is defined on routers M and J pointing to local TMS



# IP-only Network w/ Distributed TMS

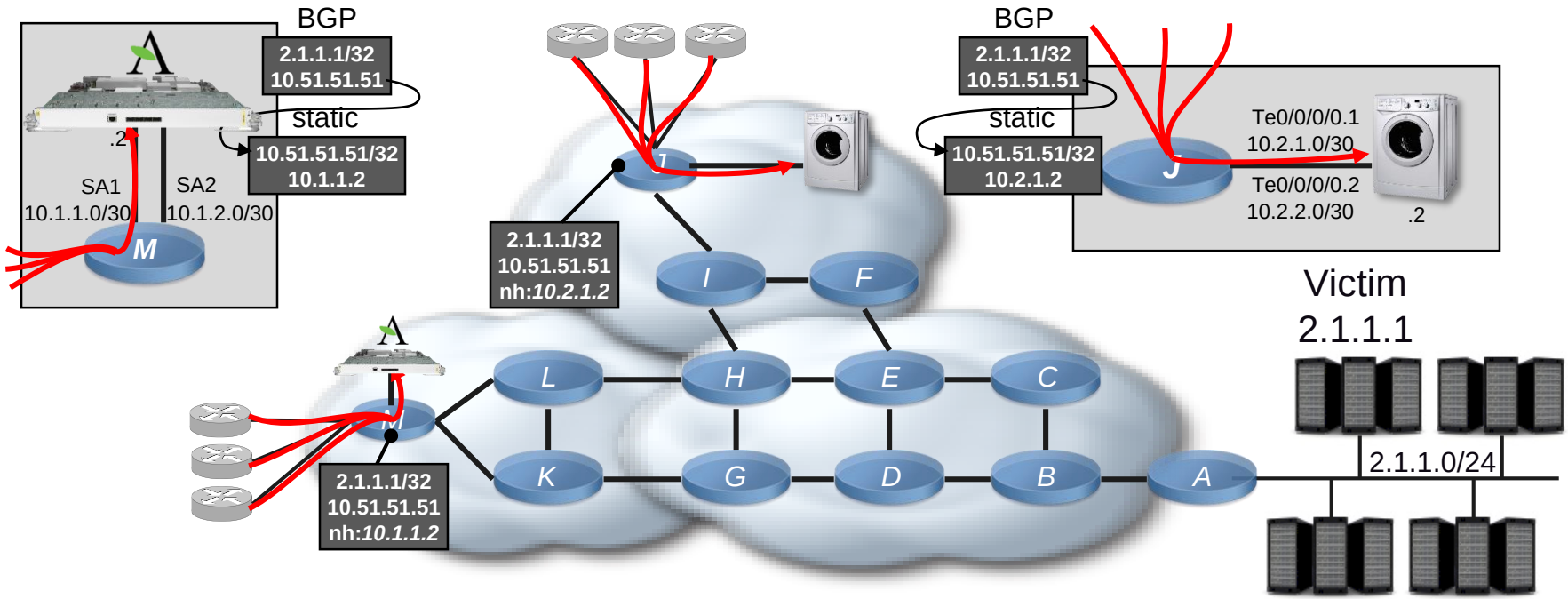
## Currently deployed




10.51.51.51 is a dummy route, advertised to trigger the redirection

# IP-only Network w/ Distributed TMS

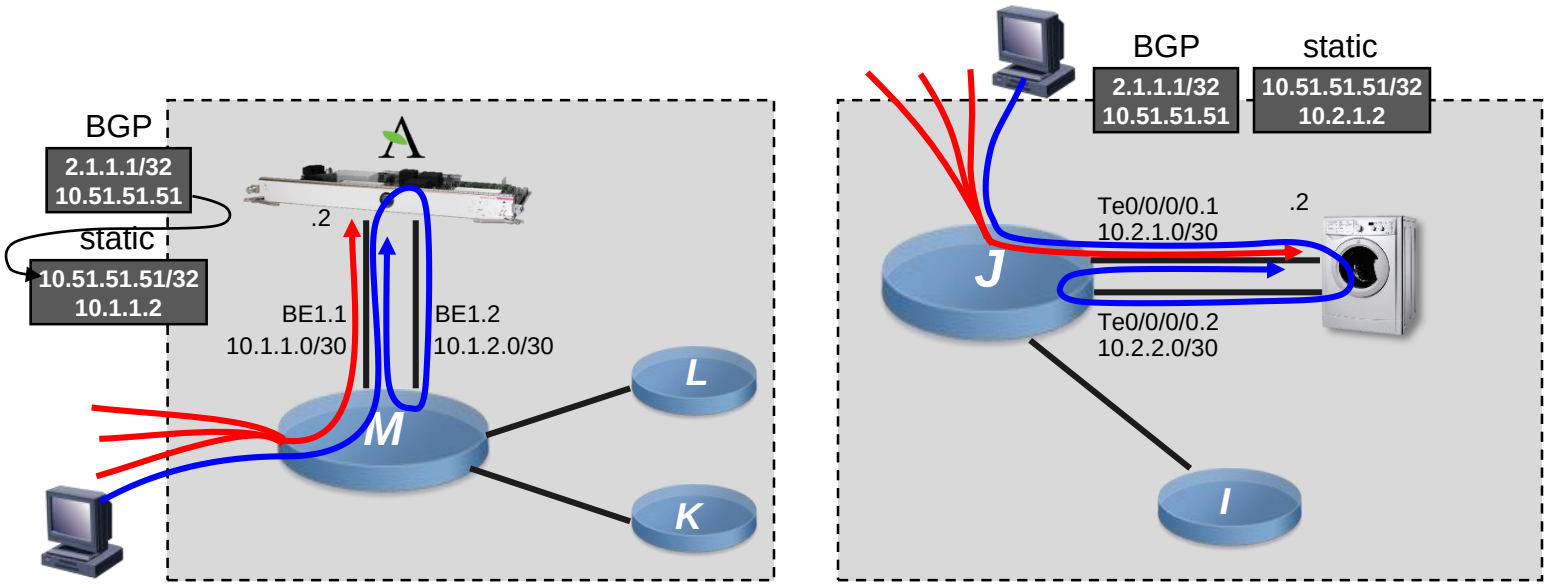
## Currently deployed



# IP-only Network w/ Distributed TMS

## Currently deployed

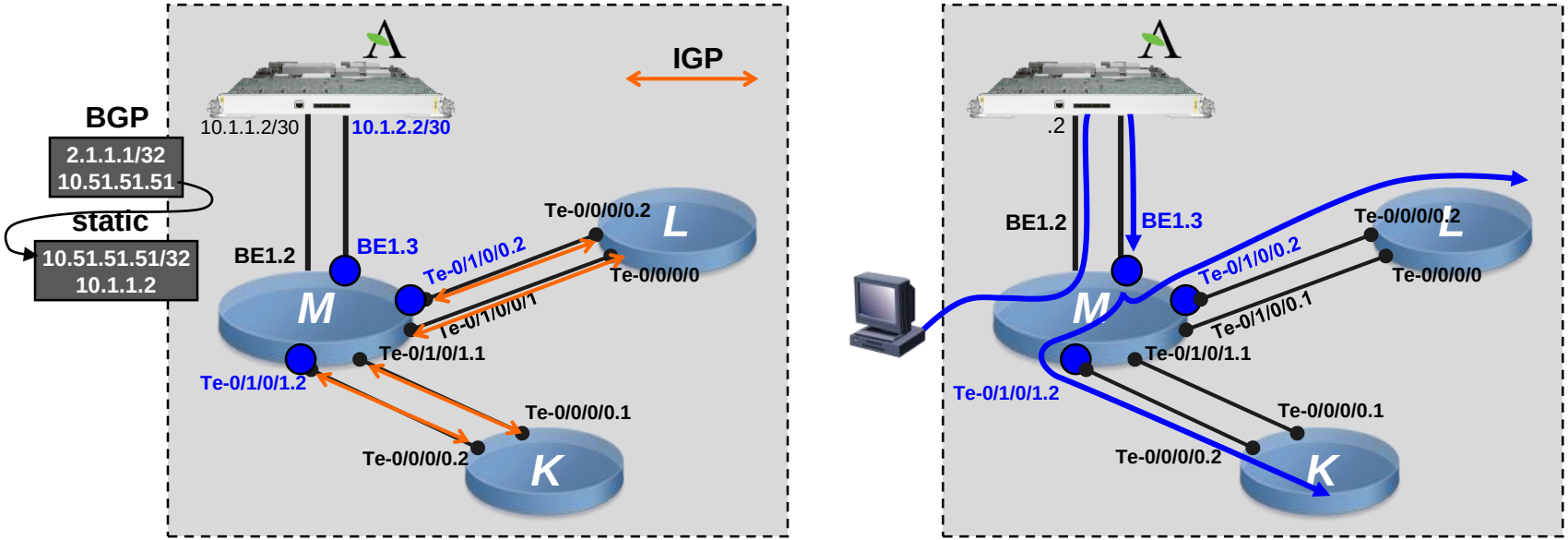
- With the specific route received we now have to deal with a routing loop for the legit traffic going out of the TMS device. We need solutions to prevent it



# IP-only Network w/ Distributed TMS

## Solution to Avoid the Routing Loop (without BGP FS)

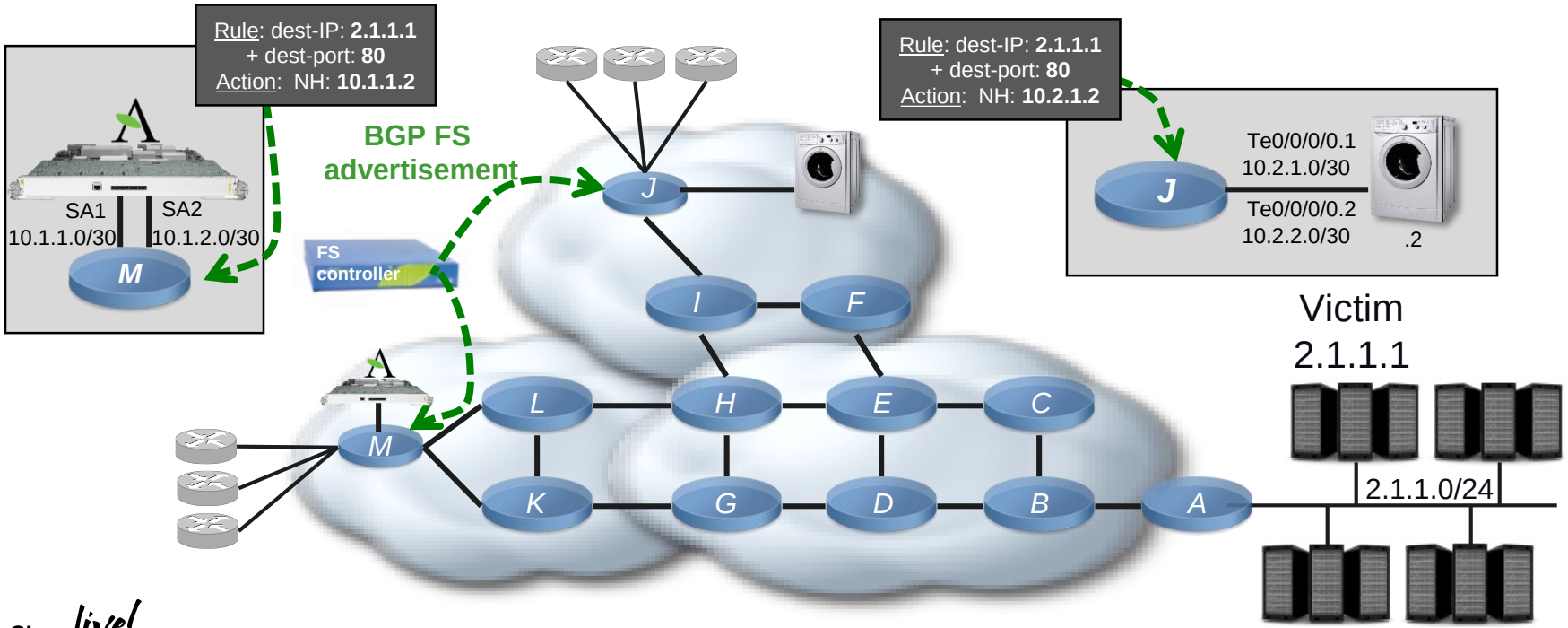
- Define an VRF-Lite Clean and assigned the egress TMS interfaces to it
  - We need two sub-interfaces to the core, one in GRT, one in the clean VRF
  - In the clean VRF, to pick the best path to the destination, we need the full IGP table



# IP-only Network w/ Distributed TMS

## BGP FlowSpec Improvement: Granularity

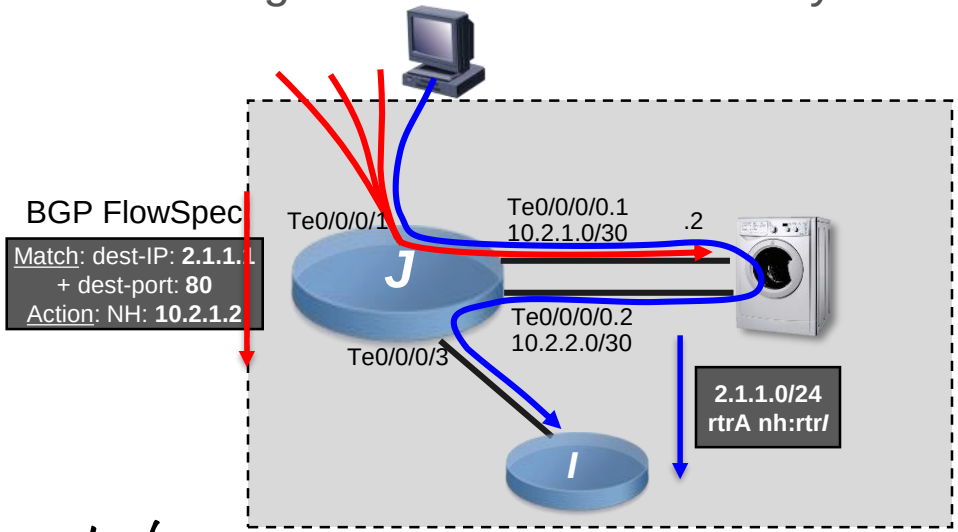
- BGP FS defines precisely the flow to divert to the local scrubbing device

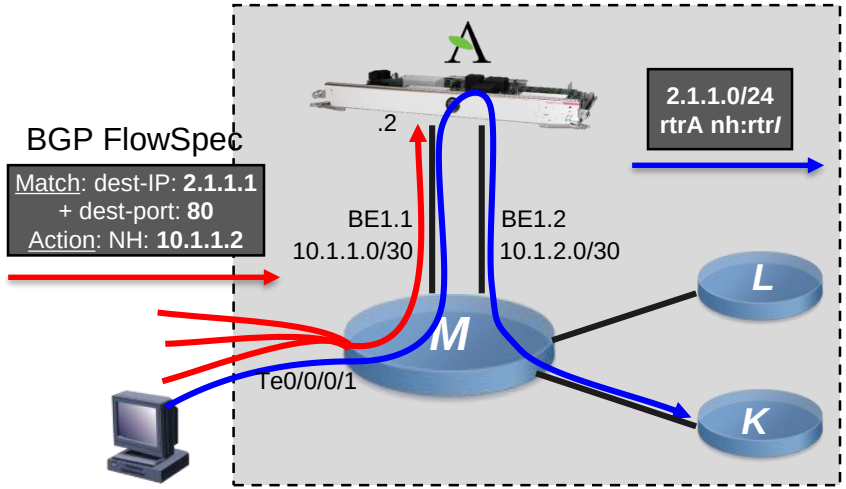


# IP-only Network w/ Distributed TMS

## BGP FlowSpec Improvement: No VRF-Lite needed

- BGP FlowSpec is activated on Te0/0/0/1, dirty traffic targeted to 2.1.1.1:80 is forwarded to the scrubbing device address 10.2.1.2
- BGP FlowSpec is deactivated on port te0/0/0/0.2, clean traffic from the scrubbing device is routed naturally via IGP route 2.1.1.0/24 to router I



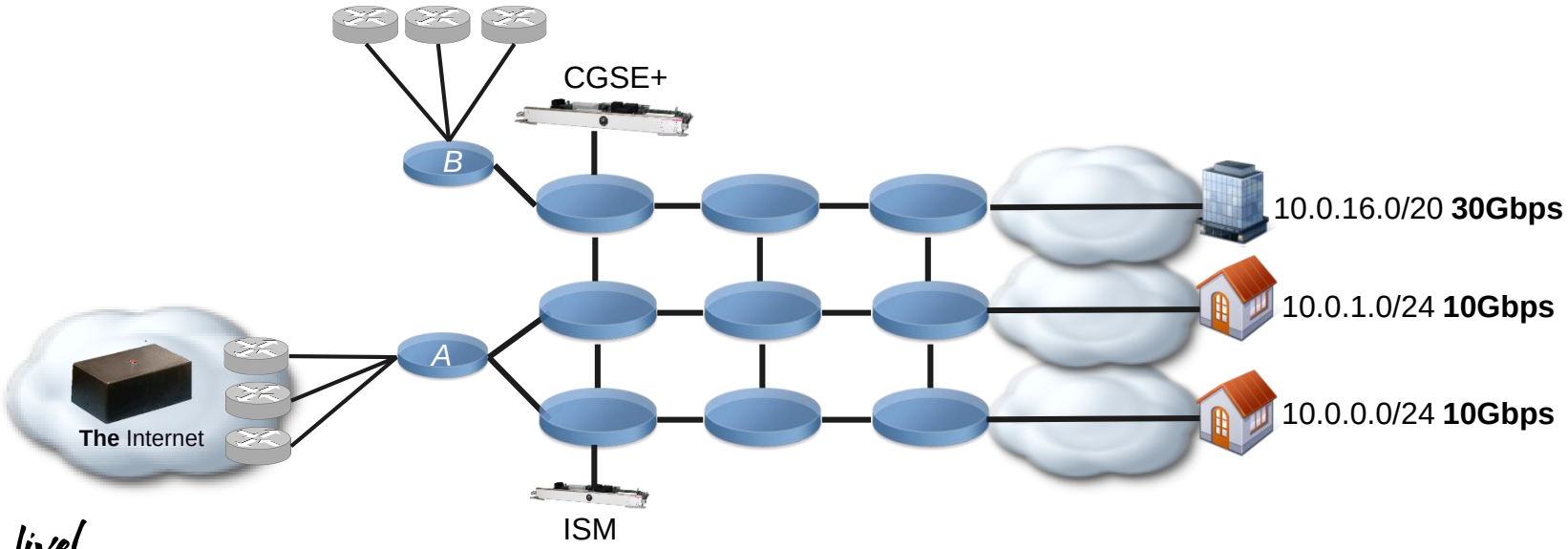



# Other Use-Cases

# Other BGP FS Use-Cases

## Unequal Load-Balancing

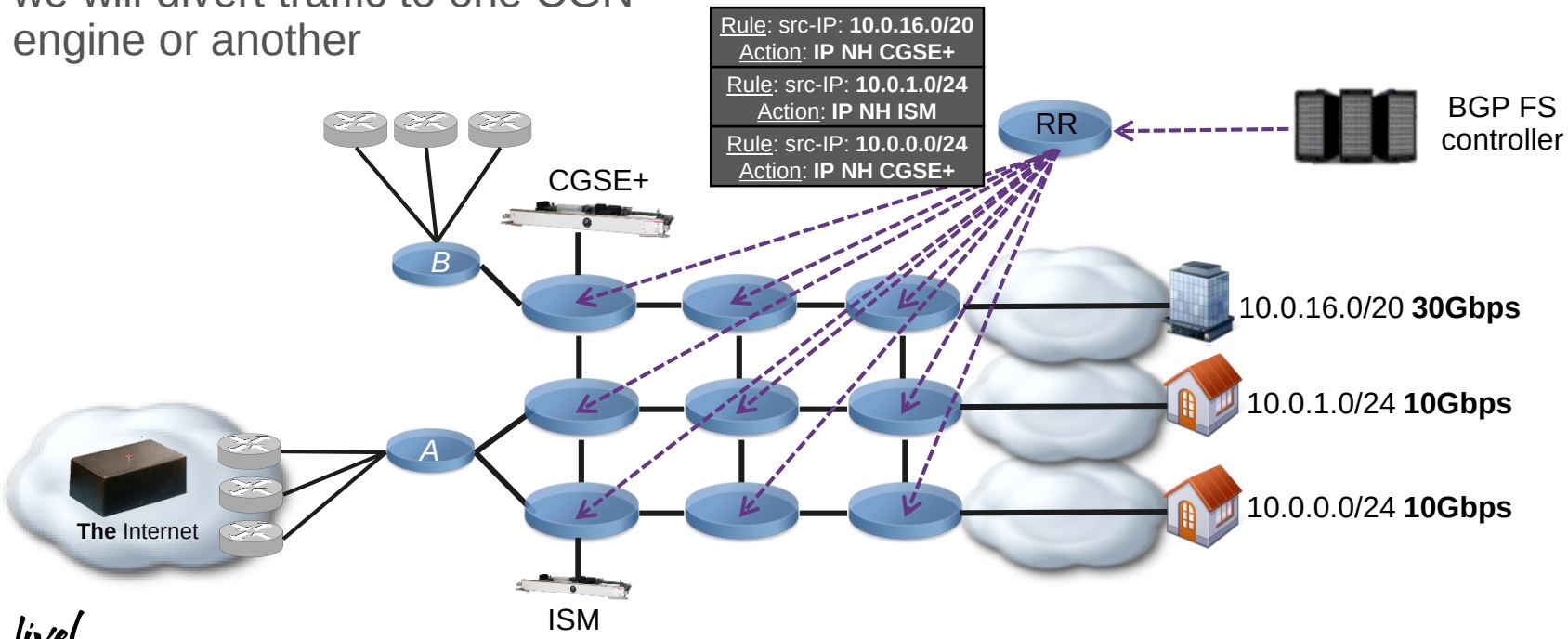
- Different peering / transit points
- Different NATing points with different performances / capabilities



# Other BGP FS Use-Cases

## Unequal Load-Balancing

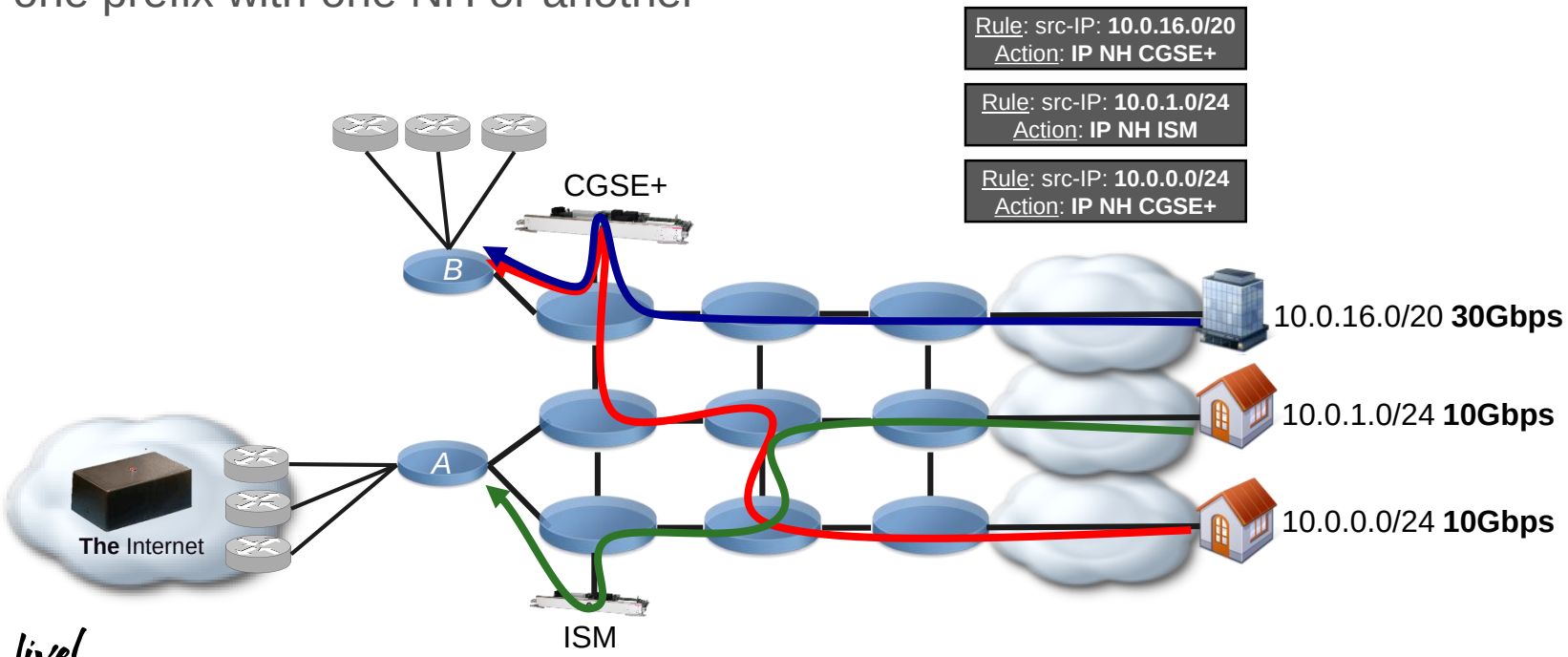
- Based on the source ranges, we will divert traffic to one CGN engine or another



# Other BGP FS Use-Cases

## Unequal Load-Balancing

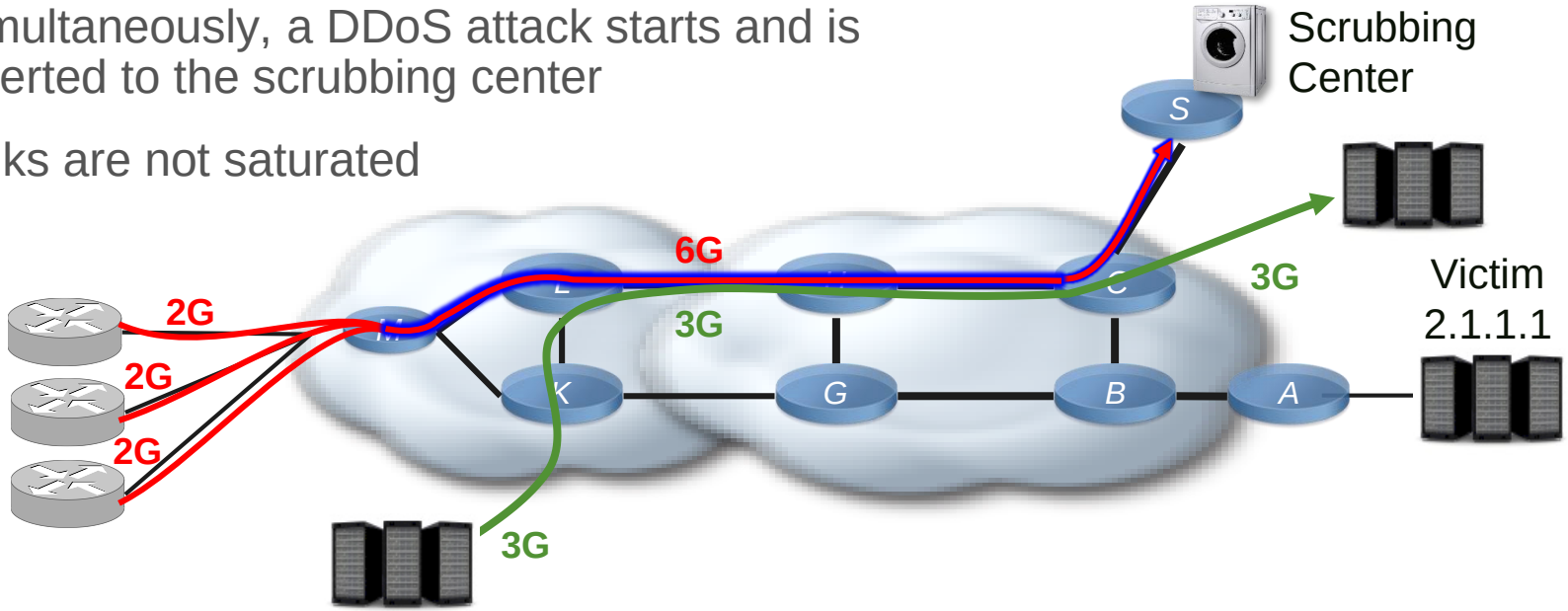
- This approach allows fine tuning of the traffic in the NAT engines, advertising one prefix with one NH or another



# Other BGP FS Use-Cases

## Low QoS Priority Traffic for DDoS Attacks

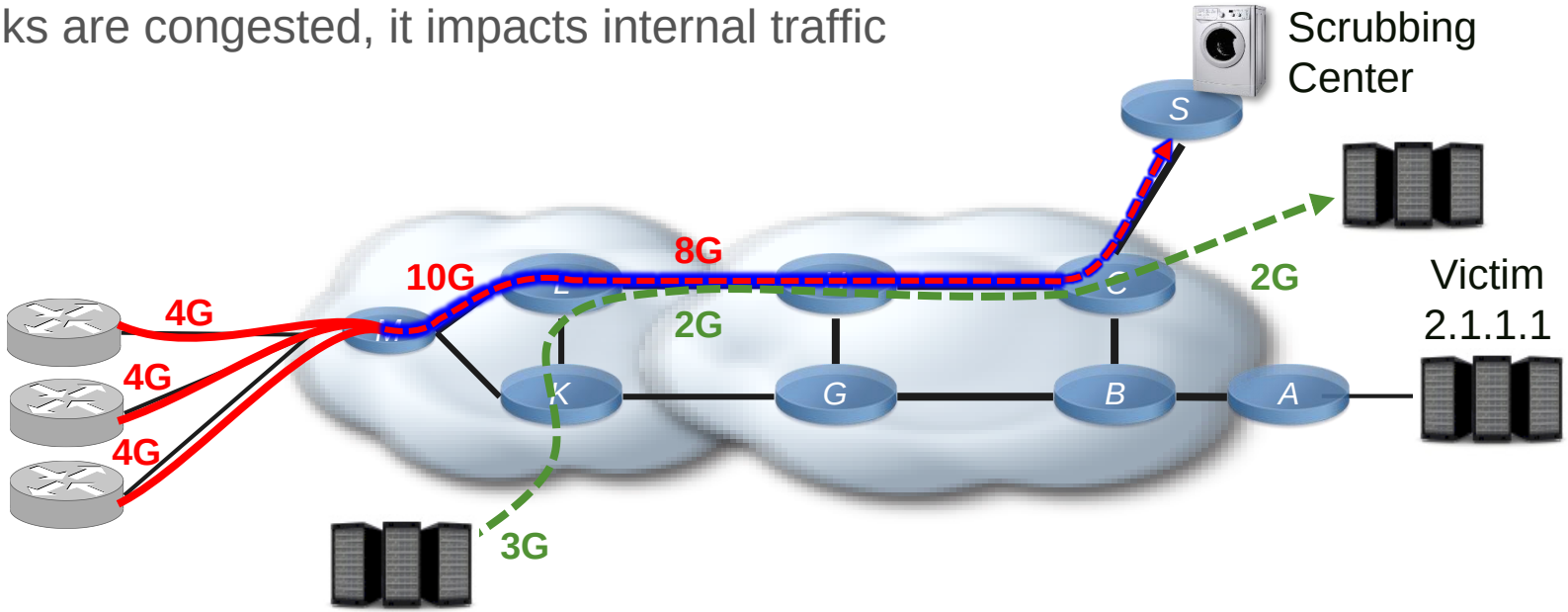
- Important back-up is using 3 Gbps of traffic
- Simultaneously, a DDoS attack starts and is diverted to the scrubbing center
- Links are not saturated



# Other BGP FS Use-Cases

## Low QoS Priority Traffic for DDoS Attacks

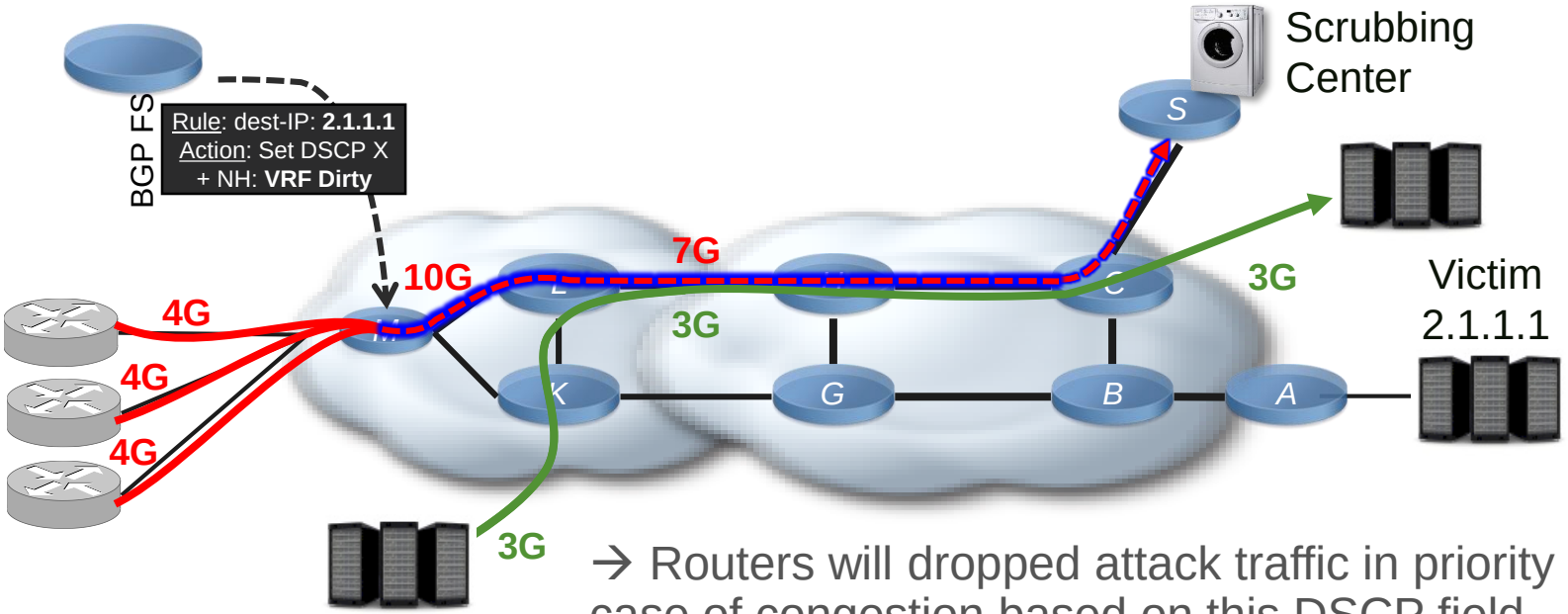
- The attack intensity increases
- Links are congested, it impacts internal traffic



# Other BGP FS Use-Cases

## Low QoS Priority Traffic for DDoS Attacks w/ FlowSpec

- BGP FS rule forces the route leaking in VRF-Dirty and positioning a DSCP field

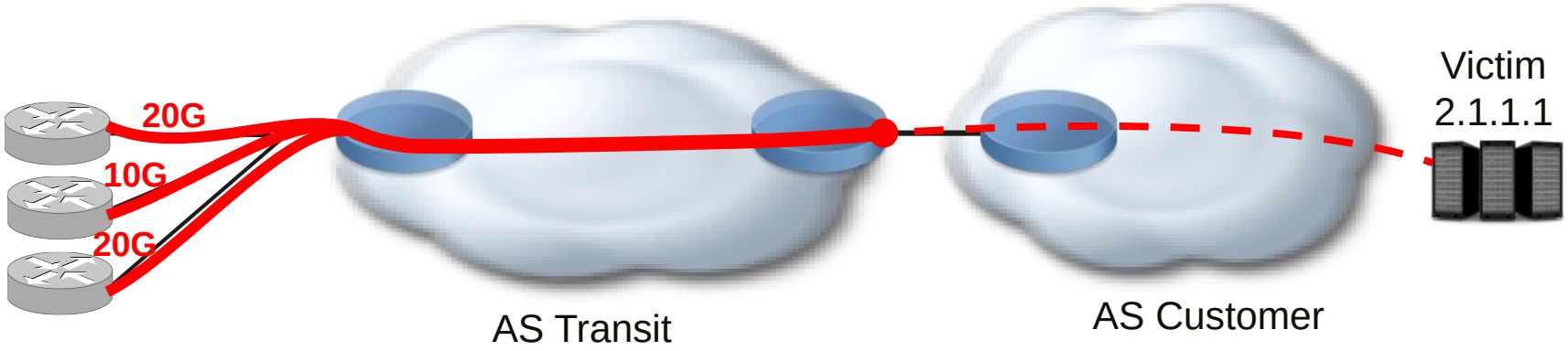



→ Routers will dropped attack traffic in priority in case of congestion based on this DSCP field

# Other BGP FS Use-Cases

## Transit AS Policing

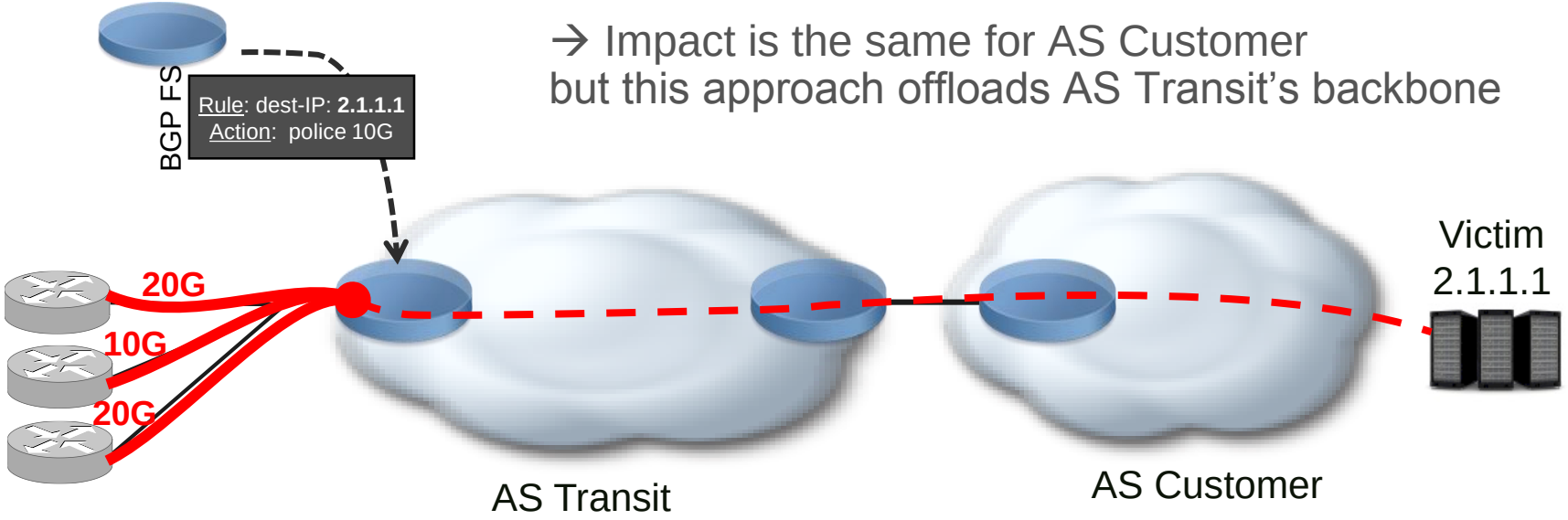
- A transit provider is offering to AS Customer a 10GE connectivity to the Internet
- An asset in AS Customer is under a heavy DDoS attack of 50Gbps
- It's pointless for AS Transit to transport the 50Gbps in it's infra to drop it on the last router connecting to AS Customer



# Other BGP FS Use-Cases

## Transit AS Policing

- AS Transit programs a BGP FS rule to rate-limit traffic targeted to the victim IP address at the level of the committed bandwidth (10Gbps here)

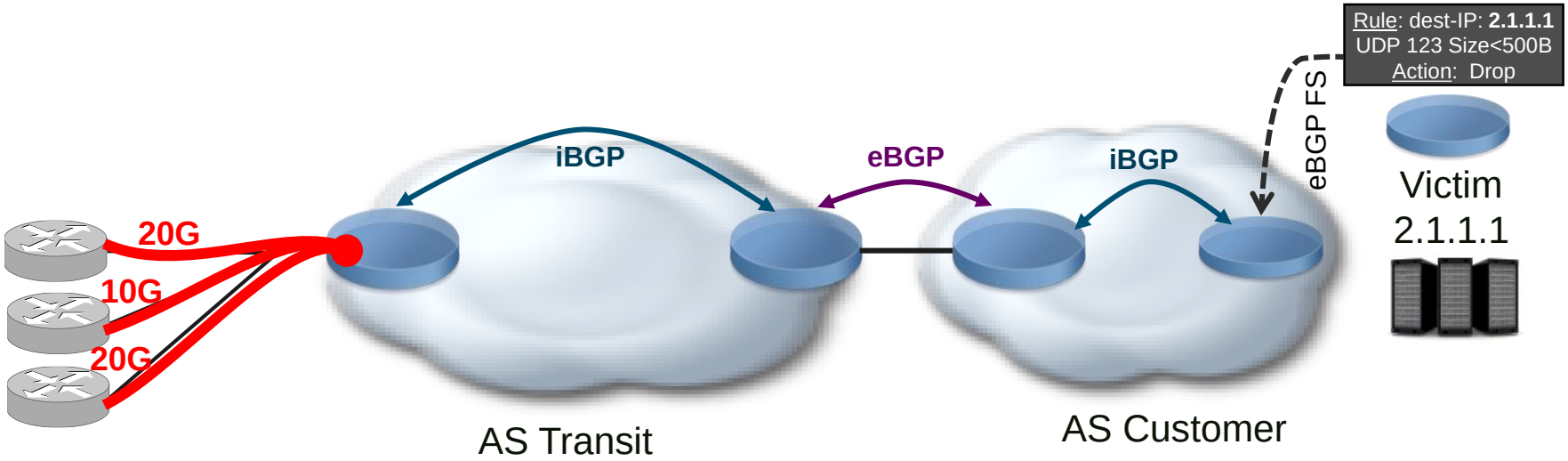



→ Impact is the same for AS Customer
but this approach offloads AS Transit's backbone

# Other BGP FS Use-Cases

## Give the Power to the Victim

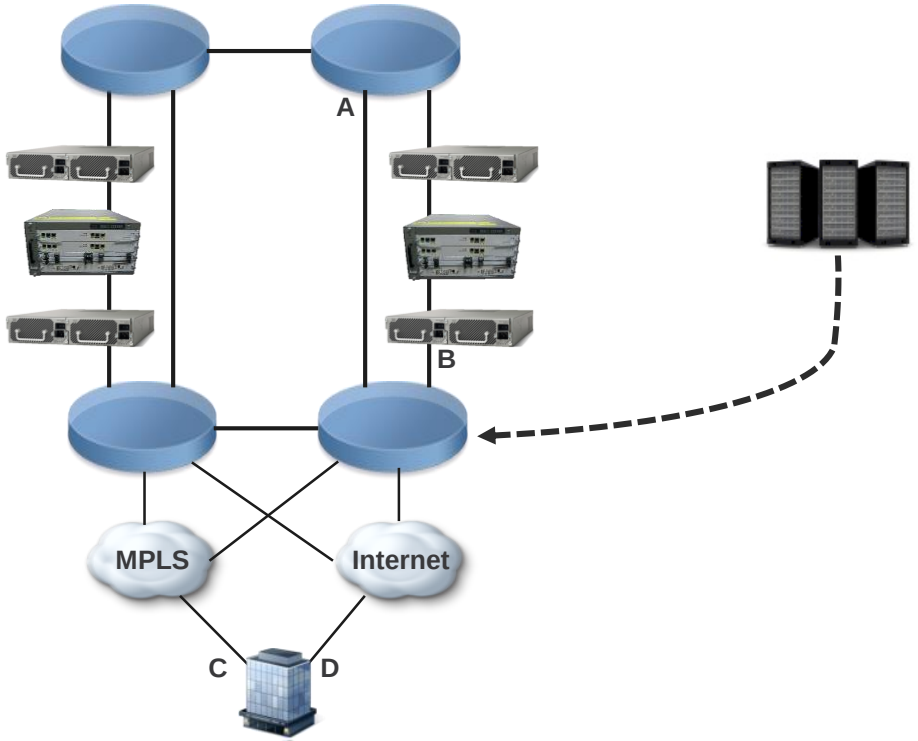
- Rule disseminated upstream to tackle the attack as early as possible
- Not popular today, but may change in the future



# Other BGP FS Use-Cases

## Enterprise PBR

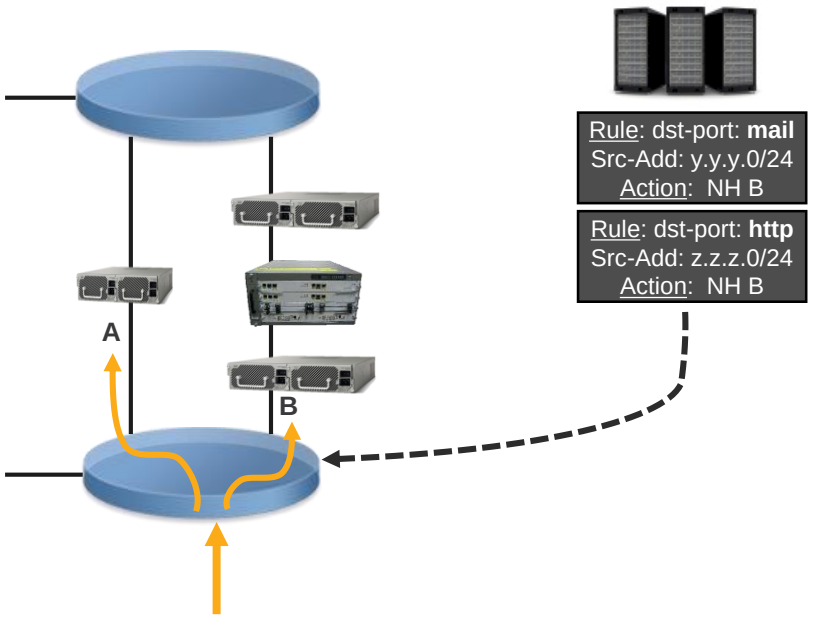
- Used to redirect traffic through security devices
- Used to select a transport (MPLS or Internet)



# Other BGP FS Use-Cases

## Enterprise PBR: Security Classification

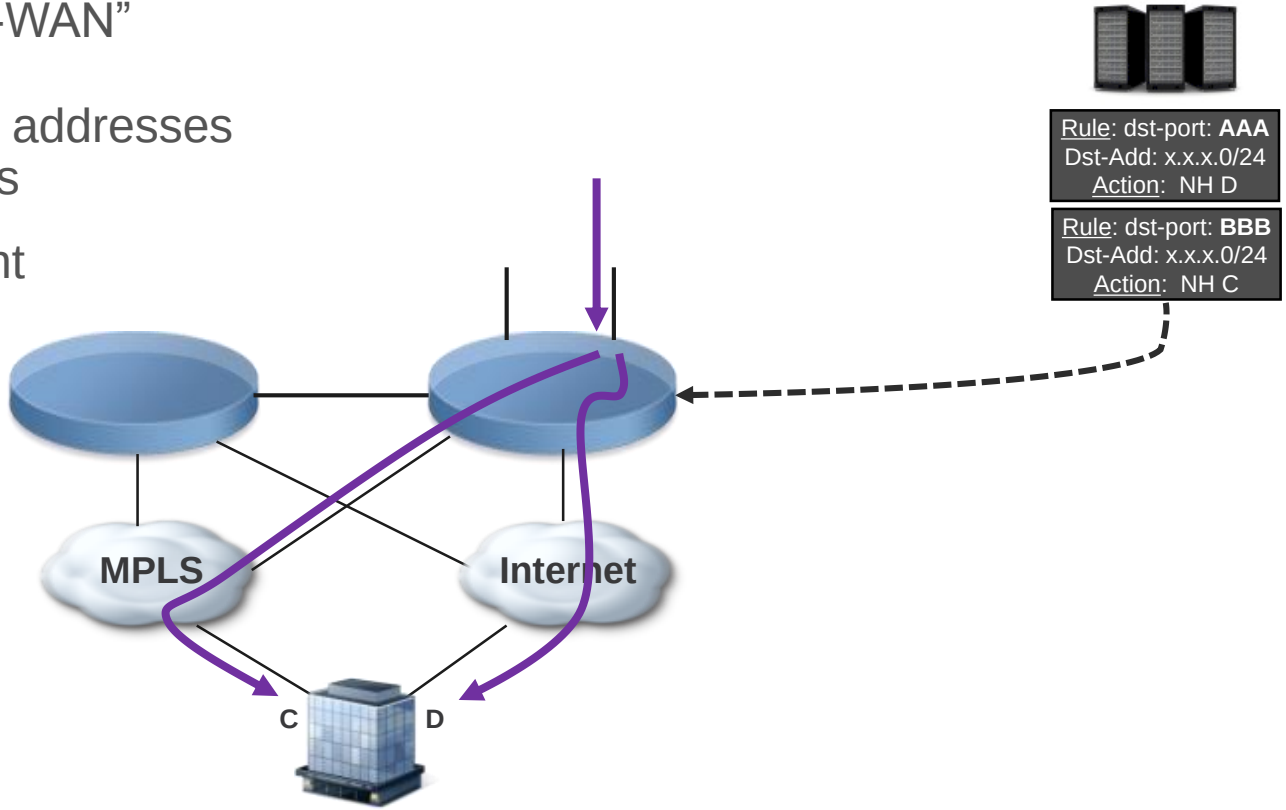
- Based on source addresses and application ports
- Packets diverted to specific security appliances (proxy, antispam, waf, fw, …)



# Other BGP FS Use-Cases

## Enterprise PBR: “SD-WAN”

- Based on dstination addresses and application ports
- Packets use different transport



# BGP FlowSpec Configuration

# Configuring BGP FlowSpec on IOS XR Routers

## Overview of the Configuration Steps

- On both Client and Controller


- On Client


- On Controller


C3PL model

Note: all examples in following slides are equally valid for IPv4 and IPv6

# Configuring BGP FlowSpec on IOS XR Routers

## Signalization: Use of a new Address-Family flowspec

### Controller

```
router bgp 1
 bgp router-id 6.6.6.6
 address-family ipv4 flowspec
 !
 neighbor-group ibgp-flowspec
  remote-as 1
  update-source loopbook0
  address-family ipv4 flowspec
  !
 !
 neighbor 25.2.1.3
  use neighbor-group ibgp-flowspec
 !
 neighbor 25.2.1.4
  use neighbor-group ibgp-flowspec
 !
!
flowspec
 address-family ipv4
  service-policy type pbr FS
```

### Client

```
router bgp 1
 bgp router-id 3.3.3.3
 address-family ipv4 flowspec
 !
 neighbor-group ibgp-flowspec
  remote-as 1
  update-source loopback0
  address-family ipv4 flowspec
  !
 neighbor 25.2.1.11
  use neighbor-group ibgp-flowspec
 !
!
flowspec
 local-install interface-all
!
```

Advertise policy FS

Install all rules on all interfaces

# Configuring BGP FlowSpec on IOS XR Routers

## Configuring Rules on the Controller

```
class-map type traffic match-all match-UDP53
 match destination-port 53
 match protocol udp
 end-class-map
!
class-map type traffic match-all match-src-ipv4-addr
 match destination-address ipv4 25.1.104.0 255.255.255.0
 end-class-map
!
```

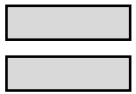
```
policy-map type pbr FS
 class type traffic match-src-ipv4-addr
  police rate 100000 bps
  !
 !
 class type traffic match-UDP53
  redirect next 192.42.52.125
  !
 !
 class type traffic class-default
 !
 end-policy-map
```

```
flowspec
 address-family ipv4
  service-policy type pbr FS
```

# Configuring BGP FlowSpec on IOS XR Routers

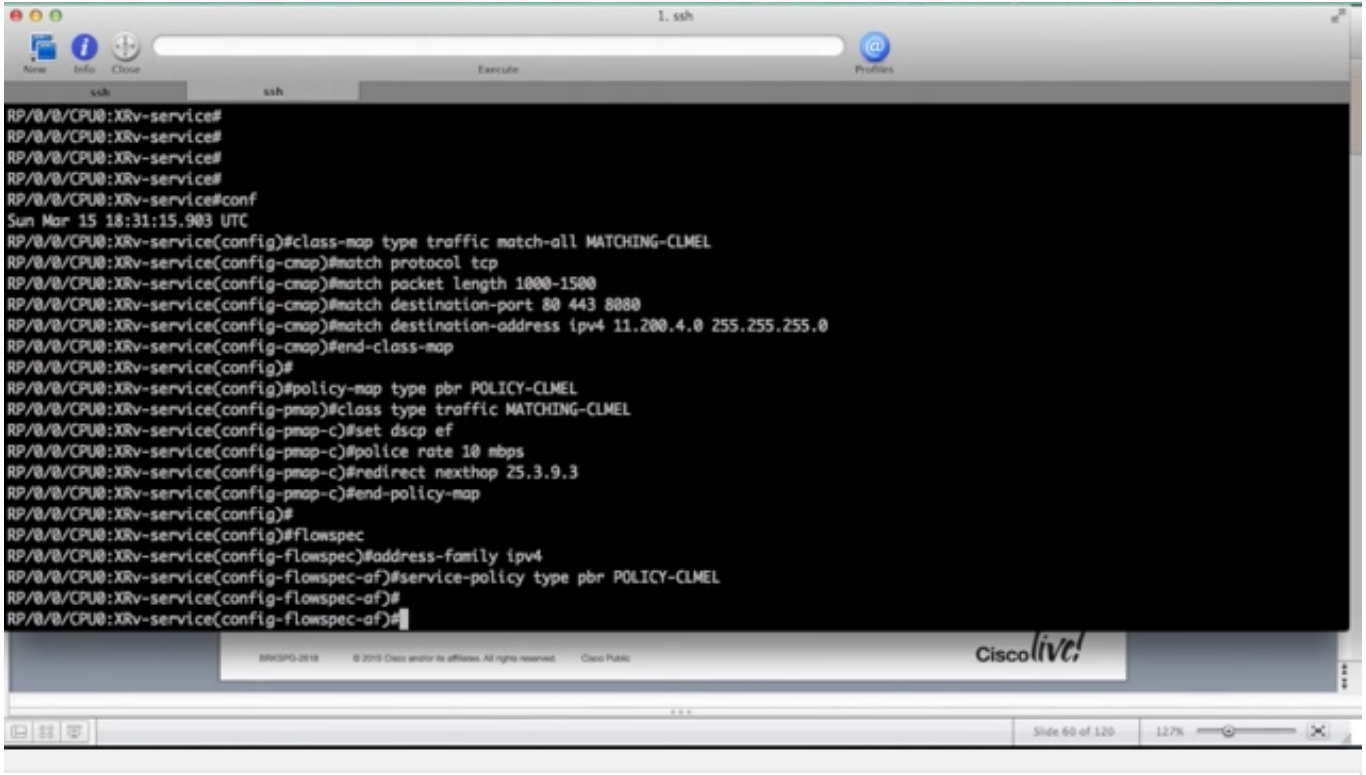
## Configuring Rules on the Controller

```
class-map type traffic match-all MATCH-UDP123
 match destination-port 123
 match protocol udp
 end-class-map
!
class-map type traffic match-all MATCH-SRCv4
 match destination-address ipv4 2.1.1.0/24
 end-class-map
!
policy-map type pbr FS1
 class type traffic MATCH-SRCv4
  police rate 100000 bps
  !
 end-policy-map
!
policy-map type pbr FS2
 class type traffic MATCH-UDP123
  redirect nexthop 192.168.2.5
  !
end-policy-map

flowspec
 address-family ipv4
  service-policy type pbr FS1
  service-policy type pbr FS2
```



```
class-map type traffic match-all MATCH-UDP123
 match destination-port 123
 match protocol udp
 end-class-map
!
class-map type traffic match-all MATCH-SRCv4
 match destination-address ipv4 2.1.1.0/24
 end-class-map
!
policy-map type pbr FS
 class type traffic MATCH-SRCv4
  police rate 100000 bps
  !
 class type traffic MATCH-UDP123
  redirect nexthop 192.168.2.5
  !
end-policy-map

flowspec
 address-family ipv4
  service-policy type pbr FS
```

# Configuration Demo




http://bit.ly/bgpfs-config

# Configuring BGP FlowSpec on IOS XR Routers

## Configuring a Type 1 Match “Destination Address”

```
RP/0/0/CPU0:Ctrl(config)#class-map type traffic match-all MATCHING-RULE
RP/0/0/CPU0:Ctrl(config-cmap)#match destination-address ipv4 81.253.193.0/24
RP/0/0/CPU0:Ctrl(config-cmap)#
```

```
RP/0/RP0/CPU0:Client#sh flowspec ipv4 detail

AFI: IPv4
 Flow            :Dest:81.253.193.0/24
  Actions        :Traffic-rate: 100000 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :                         0/0
   Transmitted          :                         0/0
   Dropped              :                         0/0
RP/0/RP0/CPU0:Client#sh flowspec ipv4 nlri

AFI: IPv4
 NLRI (Hex dump) :        0x011851fdc1
  Actions        :Traffic-rate: 100000 bps  (bgp.1)
RP/0/RP0/CPU0:Client#
```

| Type | Prefix length | Prefix |
|---|---|---|
| 1 byte | 1 byte | Variable |
| 1 | /24 | 81.253.193 |
| 0 x01 | 0x18 | 0x 51 fd c1 |

**0x011851fdc1**

# Configuring BGP FlowSpec on IOS XR Routers

## Mixing Several Matching Statements

```
class-map type traffic match-all MATCHING-RULE1
 match source-port 10 20 30-40 50-52 60-70
 match protocol udp
 match dscp ef
 match packet length 10-100 102-200 202-400 402-1500
 match destination-port 80
 match destination-address ipv4 11.200.4.0 255.255.255.0
 end-class-map
```

```
RP/0/RSP0/CPU0:Client#sh flowspec afi-all detail
AFI: IPv4
 Flow
:Dest:11.200.4.0/24,Proto:=17,DPort:=80,SPort:=10|=20|>=30&<=40|>=50&<=52|>=60&<=70,Length:>=10&<
=100|>=102&<=200|>=202&<=400|>=402&<=1500,DSCP:=46
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :                         0/0
   Dropped              :                         0/0
RP/0/RSP0/CPU0:Client#sh flowspec afi-all nlri
AFI: IPv4
 NLRI (Hex dump) :
0x01180bc804038111058150060l0a0114031e452803324534033cc5460a030a4564036645c803ca550190130192d505d
c0b812e
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
RP/0/RSP0/CPU0:Client#
```

# Configuring BGP FlowSpec on IOS XR Routers

## Configuring an Action: Police

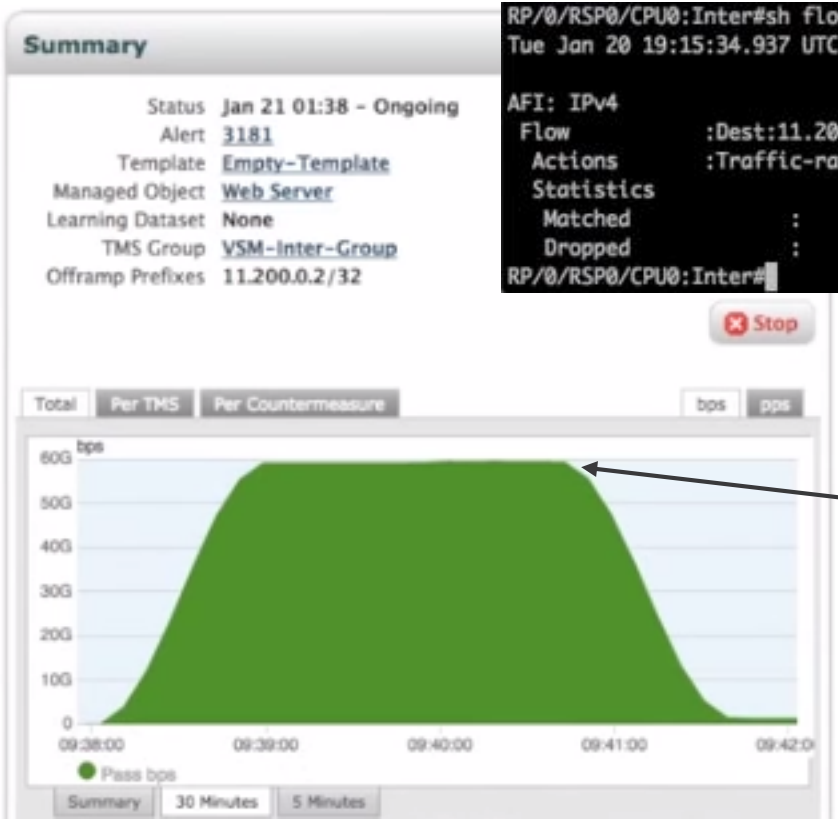
```
RP/0/0/CPU0:Ctrl(config)#policy-map type pbr FS
RP/0/0/CPU0:Ctrl(config-pmap)# class type traffic MATCHING-RULE1
RP/0/0/CPU0:Ctrl(config-pmap-c)#police ?
  rate  Committed Information Rate
RP/0/0/CPU0:Ctrl(config-pmap-c)#police rate ?
  <1-4294967295>  Committed Information Rate
RP/0/0/CPU0:Ctrl(config-pmap-c)#police rate 1000 ?
  bps      Bits per second (default)
  cellsps  Cells per second
  gbps     Gigabits per second
  kbps     Kilobits per second
  mbps     Megabits per second
  <cr>
RP/0/0/CPU0:Ctrl(config-pmap-c)#police rate 1000
RP/0/0/CPU0:Ctrl(config-pmap-c)#
```

| | TYPE (2 bytes) | ASN (only the last 2 bytes) (2 bytes) | Rate (bytes/s) (4 bytes) |
|---|---|---|---|
| RFC | | | |
| EX | 0x8006 | 0x1234 | 0x4a3ebc20 |

→ Hex 4a3ebc20 = 31,125,000 Bytes/sec
= 25 Mbps

# Configuring BGP FlowSpec on IOS XR Routers

## Configuring an Action: Police




```
RP/0/RSP0/CPU0:Inter#sh flowspec ipv4 detail
Tue Jan 20 19:15:34.937 UTC

AFI: IPv4
 Flow           :Dest:11.200.0.0/24,Proto:=17,DPort:=123,SPort:=123,Length:>=900&<=1500
  Actions       :Traffic-rate: 0 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :             92207278/115259100220
   Dropped              :             92207278/115259100220
RP/0/RSP0/CPU0:Inter#
```

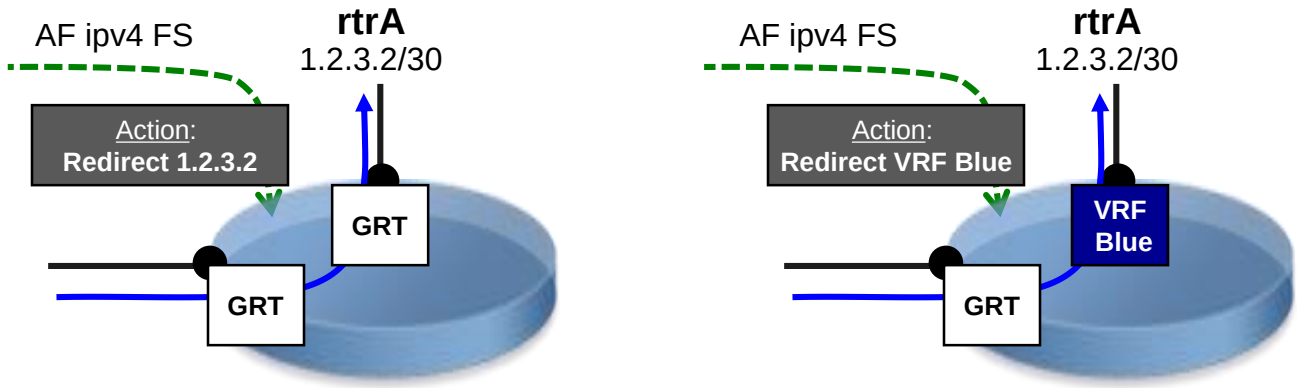
Application FS Rule

Drop = Police at 0bps

# Configuring BGP FlowSpec on IOS XR Routers

## Configuring an Action: Redirection

- If the ingress interface is in the Global Routing Table, the flowspec rule should be advertised via an “address-family IPv4 flowspec”
- Redirection to an NH address implies the egress interface is in the GRT too
- Redirection to a different VRF can not specify the destination address, a second lookup in this target VRF will happen to the destination address of the packet



# Configuring BGP FlowSpec on IOS XR Routers

## Configuring an Action: Example of a Redirection to an IP Address

### Controller Configuration

```
policy-map type pbr TEST
 class type traffic MATCHING-RULE1
  redirect nexthop 25.3.9.3
 !
 class type traffic class-default
 !
 end-policy-map
!
traffic MATCHING-RULE1
class-map type traffic match-all MATCHING-RULE1
 match protocol udp
 match packet length 500-1550
 match destination-address ipv4 25.1.102.1
255.255.255.255
 end-class-map
!
```

### Client View

```
RP/0/RSP0/CPU0:Client#show bgp ipv4 flowspec
<SNIP>
Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale, N Nexthop-
discard
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop            Metric LocPrf Weight
Path
*>iDest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550/128
                      25.3.9.3                      100      0 i

Processed 1 prefixes, 1 paths

RP/0/RSP0/CPU0:Client#show flowspec afi-all detail

AFI: IPv4
 Flow           :Dest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550
  Actions       :Nexthop: 25.3.9.3  (bgp.1)
  Statistics                        (packets/bytes)
   Matched              :                  0/0
   Dropped              :                  0/0
RP/0/RSP0/CPU0:Client#
```

For Your Reference

# Action Redirect: Digging Deeper

## Controller Configuration

```
policy-map type pbr test
 class type traffic test
  redirect ipv4 nexthop 25.3.9.3
 !
end-policy-map
```

## Client View (Debug all Flowspec Events)

```
bgp[1052]: FlowSpec: Updating NLRI Proto:=6,DPort:=80 for TBL default:IPv4.
flowspec_mgr[1094]: FlowSpec: Client bgp.1 NLRI Proto:=6,DPort:=80 Update for TBL default:IPv4.
flowspec_mgr[1094]: FlowSpec: Added client bgp.1 flow active Proto:=6,DPort:=80 with actions IP-25.3.9.3 from TBL
default:IPv4.
flowspec_mgr[1094]: FlowSpec: Finished receving 1 IPC msgs for conn 0x20000099, 0:No error.
```

In this case, we used an IPv4 address for the Next-Hop.

It’s transported as a BGP attribute and no longer as an Extended Community

# Configuring BGP FlowSpec on IOS XR Routers

## Gotchas with Redirect Action

- A rule is advertised from controller only if the configured NH is reachable
- Not necessary reachable on the client side but mandatory on the controller side

```
RP/0/0/CPU0:Ctrl#sh route 25.1.102.1

% Network not in table

RP/0/0/CPU0:Ctrl#
```

```
RP/0/RSP0/CPU0:Client#sh bgp ipv4 flowspec

RP/0/RSP0/CPU0:Client#sh bgp ipv4 flowspec sum
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
25.2.1.11         0     1   16488   16457      596    0    0 00:32:57           0

RP/0/RSP0/CPU0:Client#
```

```
RP/0/0/CPU0:Ctrl#sh run router static
router static
 address-family ipv4 unicast
  25.3.9.3/32 GigabitEthernet0/0/0/0
!
!
```

```
RP/0/RSP0/CPU0:Client#show bgp ipv4 flowspec

Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale, N Nexthop-discard
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop            Metric LocPrf Weight Path
*>iDest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550/128
                      25.3.9.3                 100      0 i

Processed 1 prefixes, 1 paths
RP/0/RSP0/CPU0:Client#
```

# Configuring BGP FlowSpec on IOS XR Routers

## Gotchas with Redirect Action

- If the NH is not reachable in the Client, the rule will be ignored

```
RP/0/RSP0/CPU0:Client#sh route 11.22.33.44

% Network not in table

RP/0/RSP0/CPU0:Client#
```

```
RP/0/0/CPU0:Ctrl#sh run policy-map type pbr TEST
policy-map type pbr TEST
 class type traffic MATCHING-RULE1
  redirect nexthop 11.22.33.44
 !
 class type traffic class-default
 !
 end-policy-map
!
RP/0/0/CPU0:XRv-service#sh run router static
router static
 address-family ipv4 unicast
  11.22.33.44/32 GigabitEthernet0/0/0/0
 !
!
RP/0/0/CPU0:Ctrl#
```

```
RP/0/RSP0/CPU0:Client#show bgp ipv4 flowspec
Dest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550/128 detail
BGP routing table entry for
Dest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550/128
<SNIP>
Last Modified: Feb  8 12:55:45.095 for 00:01:19
Paths: (1 available, no best path)
  Not advertised to any peer
  Path #1: Received by speaker 0
  Flags: 0x4000000000020005, import: 0x20
  Not advertised to any peer
  Local
    11.22.33.44 (inaccessible) from 25.2.1.11 (6.6.6.6)
      Origin IGP, localpref 100, valid, internal
      Received Path ID 0, Local Path ID 0, version 0
      Extended community: FLOWSPEC Redirect-IP:0

RP/0/RSP0/CPU0:Client#show flowspec afi-all detail

RP/0/RSP0/CPU0:Client#
```

→ No blackhole

# Configuring BGP FlowSpec on IOS XR Routers

## Mixing Multiple Actions

- We can mix several Actions:
  - Rate-limit + Redirect VRF/IP
  - Rate-limit + DSCP Marking
  - Redirect VRF/IP + DSCP Marking
  - Rate-limit + Redirect VRF/IP + DSCP Marking
- It's not possible to mix:
  - Redirect VRF + Redirect NH IP
  - Redirect NH IP@A + Redirect NH IP@B

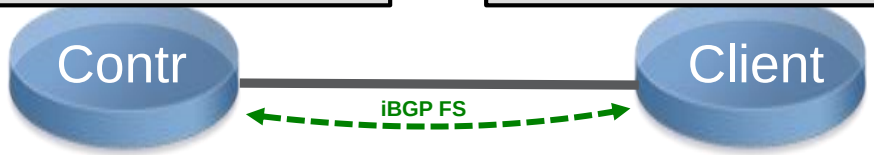

```
RP/0/RP0/CPU0:Client#sh flowspec ipv4 detail
AFI: IPv4
 Flow           :Dest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550
  Actions       :Traffic-rate: 100000 bps DSCP: ef Nexthop: 25.3.9.3  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :             75899782/106259694800
   Dropped              :             75686514/105961119600
RP/0/RP0/CPU0:Client#
```

# BGP Persistence

- In IOS XR 5.2.2 we introduced the support of the LLGR draft *draft-uttaro-idr-bgp-persistence-02*
- Both sides need to negotiate this capability when establishing the session

```
 neighbor-group ibgp-flowspec
  remote-as 1
  update-source GigabitEthernet0/0/0/0
  address-family ipv4 flowspec
   long-lived-graceful-restart capable
   long-lived-graceful-restart stale-time send 360 accept 360
!
 neighbor 10.0.0.2
  use neighbor-group ibgp-flowspec
```

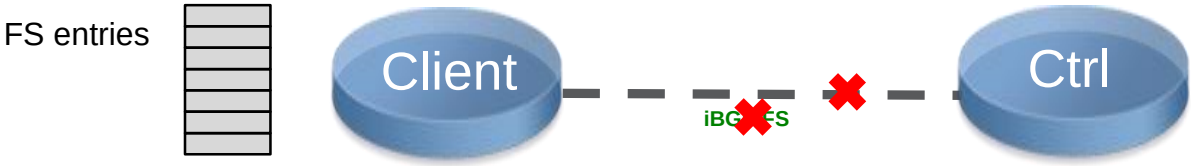
```
 neighbor-group ibgp-flowspec
  remote-as 1
  update-source GigabitEthernet0/0/0/0
  address-family ipv4 flowspec
   long-lived-graceful-restart capable
   long-lived-graceful-restart stale-time send 360 accept 360
!
 neighbor 10.0.0.1
  use neighbor-group ibgp-flowspec
```




```
RP/0/0/CPU0:Client#sh bgp ipv4 flowspec neighbors 10.0.0.1 detail | i Long-lived
Long-lived Graceful Restart Capability advertised
      Advertised Long-lived Stale time 360 seconds
    Long-lived Graceful Restart Capability received
RP/0/0/CPU0:Client#
```

# BGP Persistence

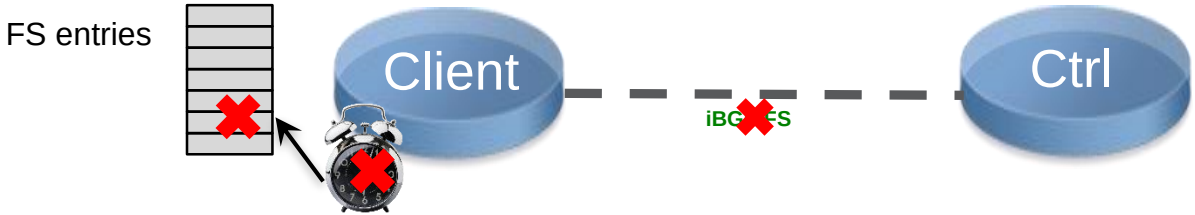
- We cut the link between the Client and Controller




```
RP/0/0/CPU0:May 11 16:01:53.980 : bgp[1052]: %ROUTING-BGP-5-ADJCHANGE : neighbor 10.0.0.1 Down - BGP
Notification sent, hold time expired (VRF: default) (AS: 1)
RP/0/0/CPU0:May 11 16:01:53.980 : bgp[1052]: %ROUTING-BGP-5-NSR_STATE_CHANGE : Changed state to NSR-Ready
RP/0/0/CPU0:Client#sh bgp ipv4 flowspec sum
BGP router identifier 2.2.2.2, local AS number 1
<SNIP>
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.0.0.1          0     1    7508    7513        0    0    0 00:00:29 Active

RP/0/0/CPU0:Client#sh flowspec ipv4 detail
AFI: IPv4
  Flow           :Dest:11.200.4.0/24,Proto:=6,DPort:=80|=443|=8080,Length:>=1000&<=1500
    Actions      :Traffic-rate: 10000000 bps DSCP: ef Nexthop: 25.3.9.3  (bgp.1)
RP/0/0/CPU0:Client#sh bgp ipv4 flowspec neighbors 10.0.0.1 detail | i "(Long|LLGR)"
Long-lived Graceful Restart Capability advertised
      Advertised Long-lived Stale time 360 seconds
    Remaining LLGR stalepath time 320
RP/0/0/CPU0:Client#
```

# BGP Persistence

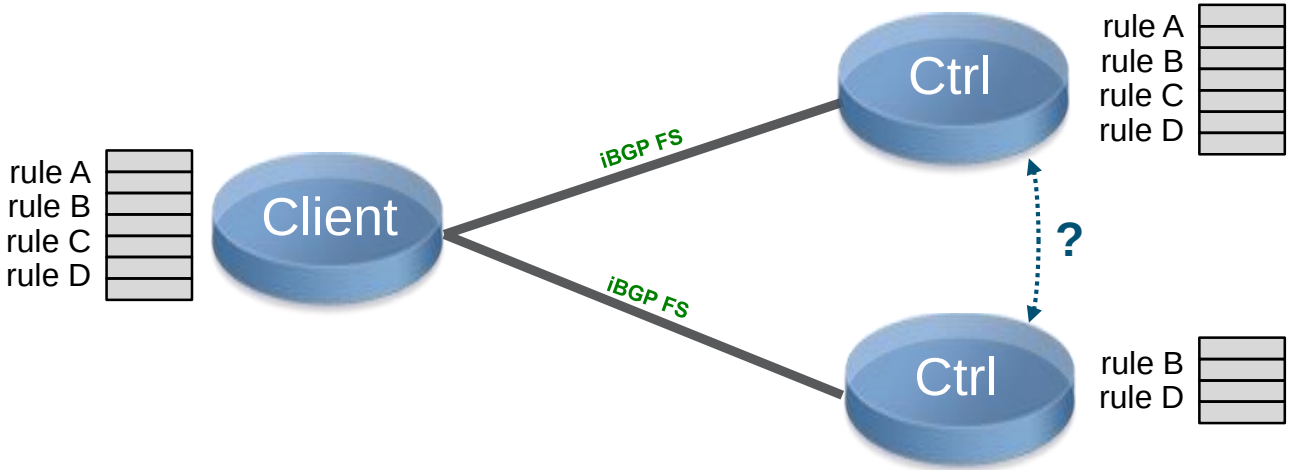
- When the timer expires, the associated BGP FS entries are removed




```
RP/0/0/CPU0:XRv2-demo#sh bgp ipv4 flowspec neighbors 10.0.0.1 detail | i "(Long|LLGR)"
Mon May 11 16:07:53.845 UTC
    Long-lived Graceful Restart Capability advertised
      Advertised Long-lived Stale time 360 seconds
    Remaining LLGR stalepath time 2
RP/0/0/CPU0:XRv2-demo#sh bgp ipv4 flowspec neighbors 10.0.0.1 detail | i "(Long|LLGR)"
Mon May 11 16:08:01.285 UTC
    Long-lived Graceful Restart Capability advertised
      Advertised Long-lived Stale time 360 seconds
    Long-lived Graceful Restart not in effect as Graceful Restart capability not received
RP/0/0/CPU0:XRv2-demo#sh flowspec ipv4 detail
Mon May 11 16:08:04.615 UTC
RP/0/0/CPU0:XRv2-demo#
```

# BGP FS Controller Redundancy

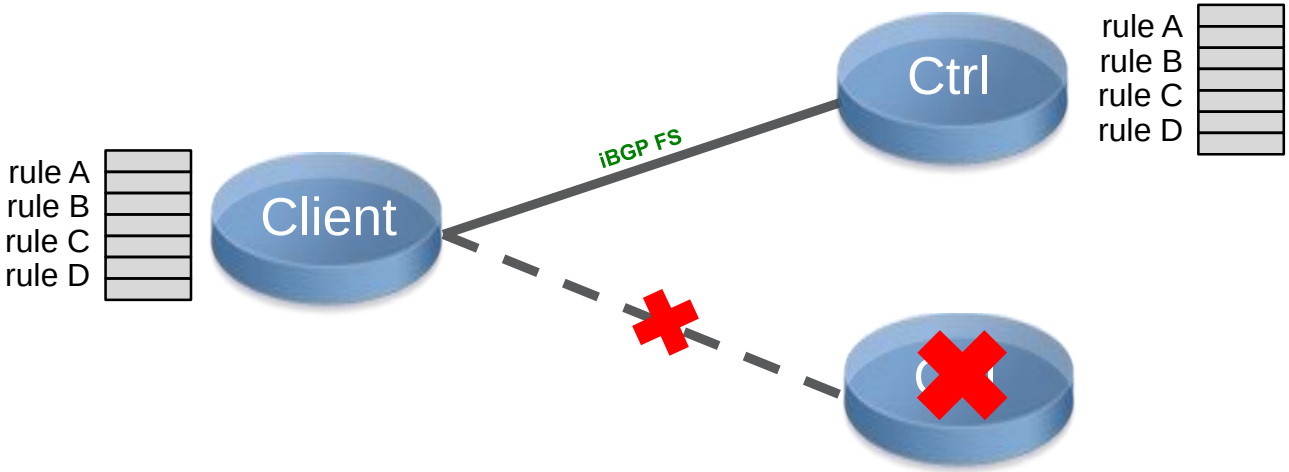
- No Controller to Controller protocol to sync the rules advertisement




- You need manual config or scripting to align config on each Controller

# BGP FS Controller Redundancy

- If a controller is lost, the rules are not temporarily removed and re-installed



# Configuring BGP FlowSpec

## Order of Matching Types

- Not dependent on the arrival order of the flow specification's rules
- The algorithm starts by comparing the left-most components of the rules.
- If the types differ, the rule with lowest numeric type value has higher precedence (and thus will match before) than the rule that doesn't contain that component type.

Order of preference

| NLRI type | Match fields |
|---|---|
| Type 1 | IPv4 Destination address |
| Type 2 | IPv4 Source address |
| Type 3 | IPv4 protocol |
| Type 4 | IPv4 source or destination port |
| Type 5 | IPv4 destination port |
| Type 6 | IPv4 Source port |
| Type 7 | IPv4 ICMP type |
| Type 8 | IPv4 ICMP code |
| Type 9 | IPv4 TCP flags (2 bytes include reserved bits) |
| Type 10 | IPv4 Packet length |
| Type 11 | IPv4 DSCP |
| Type 12 | IPv4 fragmentation bits |

For Your Reference

# Configuring BGP FlowSpec

## Order of Matching Types

- If the component types are the same, then a type-specific comparison is performed.
- For IP prefix values (IP destination and source prefix) precedence is given to the lowest IP value of the common prefix length; if the common prefix is equal, then the most specific prefix has precedence.
- For all other component types, unless otherwise specified, the comparison is performed by comparing the component data as a binary string using the memcmp() function as defined by the ISO C standard.
- For strings of different lengths, the common prefix is compared. If equal, the longest string is considered to have higher precedence than the shorter one.

# Configuring BGP FlowSpec

For Your Reference

```
class-map type traffic match-all MATCHING-RULE1
 match protocol udp
 match packet length 500-1550
 match destination-address ipv4 25.1.102.1 255.255.255.255
 end-class-map
!
class-map type traffic match-all MATCHING-RULE2
 match protocol udp
 match packet length 500-1550
 match destination-address ipv4 25.1.102.0 255.255.255.0
 end-class-map
!
policy-map type pbr TEST1
 class type traffic MATCHING-RULE1
  redirect nexthop 25.4.9.3
 class type traffic class-default
 !
 end-policy-map
!
policy-map type pbr TEST2
 class type traffic MATCHING-RULE2
  redirect nexthop 25.3.9.3
 class type traffic class-default
 !
 end-policy-map
flowspec
 address-family ipv4
  service-policy type pbr TEST1
  service-policy type pbr TEST2
 !
```

Controller

```
RP/0/RSP0/CPU0:Client#show flowspec afi-all detail

AFI: IPv4
 Flow
:Dest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550
  Actions       :Nexthop: 25.4.9.3   (bgp.1)
  Statistics                         (packets/bytes)
   Matched              :            304006799/425609518600
   Dropped              :                    0/0
 Flow
:Dest:25.1.102.0/24,Proto:=17,Length:>=500&<=1550
  Actions       :Nexthop: 25.3.9.3   (bgp.1)
  Statistics                         (packets/bytes)
   Matched              :                    0/0
   Dropped              :                    0/0

RP/0/RSP0/CPU0:Client#
```

Client

25.1.102.1/32 more specific than 25.1.102.0/24

# NLRI Filtering

"Safety Net"

- We don't want any user or operator to accidentally blackhole important traffic
  - DNS servers (8.8.8.8)
  - Infrastructure addresses (routers, tacacs/radius, netflow collectors, snmp, …)
  - Addresses of other customers
- Local definition of prefixes / protocols which can NOT be overruled by BGP FlowSpec

# NLRI Filtering

## Configuration

```
prefix-set ALLOW-FLOW
 1.1.1.0/24 ge 32
end-set
!
route-policy ALLOW-FLOW-POLICY
  if destination-prefix in ALLOW-FLOW then
    pass
  endif
end-policy
!
router bgp 65117
 neighbor 25.2.1.14
  remote-as 65117
  update-source GigabitEthernet0/0/0/0
  address-family ipv4 flowspec
   route-policy ALLOW-FLOW-POLICY in
  !
```

- Server advertises two BGP FS rules:
  - Destination 1.1.1.1/32
  - Destination 1.1.2.1/32

```
RP/0/RP0/CPU0:Client#sh flowspec ipv4 detail
AFI: IPv4
  Flow           :Dest:1.1.1.1/32
    Actions      :Traffic-rate: 0 bps  (bgp.1)
RP/0/RP0/CPU0:Client#
```

→ Only the 1.1.1.1/32 rule is accepted and configured.

# Consistency Checking

## Example: TCP with ICMP Code

```
class-map type traffic match-all c21
 match protocol tcp
 match ipv4 icmp-type 10
 end-class-map
```

```
RP/0/0/CPU0:CONTROLLER#show flowspec vrf foo1 ipv4 internal
 VRF: foo1     AFI: IPv4
  Flow           :Proto:=6,ICMPType:=10
    Actions      :DSCP: af11  (policy.1.p21.c21)
<... SNIP ...>
   Sequence:                  1024
   Match Unsupported:         ICMP type/code with non-ICMP protocol
   Synced:                    FALSE
<... SNIP ...>
   Statistics                          (packets/bytes)
     Matched             :                    0/0
     Transmitted         :                    0/0
     Dropped             :                    0/0
RP/0/0/CPU0:CONTROLLER#
```

For Your Reference

# Consistency Checking

## Other Examples

```
class-map type traffic match-all c22
 match protocol icmp
 match tcp-flag 16
 end-class-map
```

```
RP/0/0/CPU0:CONTROLLER#show flowspec vrf foo2 ipv4 internal
VRF: foo2     AFI: IPv4
  Flow           :Proto:=1,TCPFlags:=0x10
    Actions      :DSCP: af11  (policy.1.p22.c22)
<... SNIP ...>
   Sequence:                 1024
   Match Unsupported:        TCP flags with non-TCP protocol
   Synced:                   FALSE
<... SNIP ...>
   Statistics                          (packets/bytes)
     Matched             :                   0/0
     Transmitted         :                   0/0
     Dropped             :                   0/0
RP/0/0/CPU0:CONTROLLER#
```

```
class-map type traffic match-all c23
 match protocol icmp
 match destination-port 10
 end-class-map
```

```
RP/0/0/CPU0:CONTROLLER#show flowspec vrf foo3 ipv4 internal
VRF: foo3     AFI: IPv4
  Flow           :Proto:=1,DPort:=10
    Actions      :DSCP: af11  (policy.1.p23.c23)
<... SNIP ...>
   Sequence:                 1024
   Match Unsupported:        Port with non-TCP/UDP protocol
   Synced:                   FALSE
<... SNIP ...>
   Statistics                          (packets/bytes)
     Matched             :                   0/0
     Transmitted         :                   0/0
     Dropped             :                   0/0
RP/0/0/CPU0:CONTROLLER#
```

# Checking Counters with Netconf/XML

- Proprietary models are available for configuration and monitoring

```
<rpc message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
 <get>
  <filter>
   <Operational>
<FlowSpec></FlowSpec>
</Operational>
</filter>
</get>
</rpc>]]>]]>
```

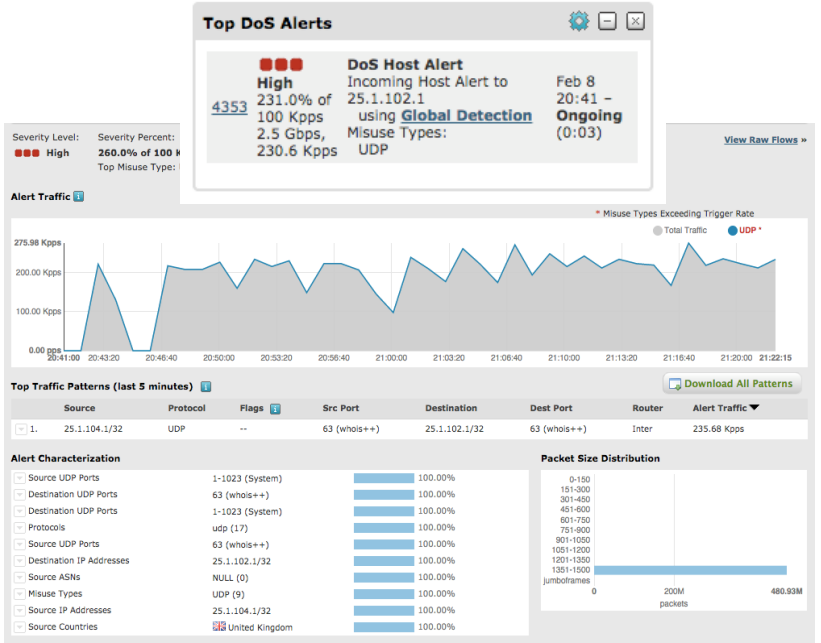
```
<<<SNIP>>>
          <FlowTable>
            <Flow>
              <Naming>
                <FlowNotation>
                  Dest:25.1.104.0/24
                </FlowNotation>
              </Naming>
              <FlowStatistics>
                <Classified>
                  <Packets>
                    21946725652
                  </Packets>
                  <Bytes>
                    13958117514672
                  </Bytes>
                </Classified>
                <Dropped>
                  <Packets>
                    21946488774
                  </Packets>
                  <Bytes>
                    13957966860264
                  </Bytes>
                </Dropped>
              </FlowStatistics>
            </Flow>
<<</SNIP>>>
```

# Netflow Sampling vs BGP flowspec

- Even if a BGP flowspec rule drops the packets, they are sampled and handled by the linecard CPU.

```
RP/0/RSP0/CPU0:Client#sh run int hundredGigE 0/0/0/0
interface HundredGigE0/0/0/0
 description *** to Boca ***
 cdp
 ipv4 address 25.1.9.4 255.255.255.0
 load-interval 30
 flow ipv4 monitor MON-MAP-IP sampler SAM-MAP ingress
!

RP/0/RSP0/CPU0:Client#sh flowspec ipv4 detail
AFI: IPv4
 Flow            :Proto:=17,Length:>=500&<=1550
  Actions        :Traffic-rate: 0 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched                :          146077011/182594343700
   Dropped                :          146077011/182594343700
RP/0/RSP0/CPU0:Client#
```

Attack still detected



# Netflow Sampling vs BGP flowspec

- Before applying the BGP FlowSpec rules, we check the NF cache:

```
RP/0/RSP0/CPU0:Client#sh flow monitor MON-MAP-IP cache location 0/0/CPU0
Cache summary for Flow Monitor MON-MAP-IP:
Cache size:                             1000000
Current entries:                         164916
Flows added:                            2043769
<SNIP>
Flows exported                          1878853

IPV4SrcAddr      IPV4DstAddr      L4SrcPort  L4DestPort BGPDstOrigAS BGPSrcOrigAS BGPNextHopV4     IPV4DstPrfxLen
IPV4SrcPrfxLen  IPV4Prot IPV4TOS  InputInterface  OutputInterface L4TCPFlags   ForwardStatus        FirstSwitched
LastSwitched    ByteCount    PacketCount  Dir SamplerID  InputVRFID                        OutputVRFID
100.102.8.178    11.200.0.2       123        123        0            0            0.0.0.0          24
0               udp      0       Hu0/0/0/0       Te0/2/0/1       0            Fwd                   12 15:47:40:093
12 15:47:40:093 1402         1            Ing 1          default                           default
100.2.42.67      11.200.0.2       123        123        0            0            0.0.0.0          24
0               udp      0       Hu0/0/0/0       Te0/2/0/1       0            Fwd                   12 15:47:51:618
12 15:47:51:618 1182         1            Ing 1          default                           default
100.77.86.28     11.200.0.2       123        123        0            0            0.0.0.0          24
0               udp      0       Hu0/0/0/0       Te0/2/0/1       0            Fwd                   12 15:48:31:530
12 15:48:31:530 1082         1            Ing 1          default                           default

RP/0/RSP0/CPU0:Client#
```

# Netflow Sampling vs BGP flowspec

- After applying the BGP FlowSpec rules, we check the NF cache:

```
RP/0/RSP0/CPU0:Client#sh flow monitor MON-MAP-IP cache location 0/0/CPU0
Cache summary for Flow Monitor MON-MAP-IP:
Cache size:                             1000000
Current entries:                          12706
Flows added:                            1467559
<SNIP>
Flows exported                          1454853

IPV4SrcAddr      IPV4DstAddr      L4SrcPort  L4DestPort BGPDstOrigAS BGPSrcOrigAS BGPNextHopV4     IPV4DstPrfxLen
IPV4SrcPrfxLen  IPV4Prot IPV4TOS  InputInterface  OutputInterface L4TCPFlags   ForwardStatus        FirstSwitched
LastSwitched    ByteCount    PacketCount  Dir SamplerID  InputVRFID                        OutputVRFID
100.37.17.132    11.200.0.2       123        123        0            0            0.0.0.0          24
0               udp      0        Hu0/0/0/0       0               0            DropACLDeny          12 15:45:00:310
12 15:45:00:310 1362         1            Ing 1          default                          0
100.47.47.62     11.200.0.2       123        123        0            0            0.0.0.0          24
0               udp      0        Hu0/0/0/0       0               0            DropACLDeny          12 15:45:01:850
12 15:45:01:850 1122         1            Ing 1          default                          0
100.11.100.55    11.200.0.2       123        123        0            0            0.0.0.0          24
0               udp      0        Hu0/0/0/0       0               0            DropACLDeny          12 15:45:00:947
12 15:45:00:947 1462         1            Ing 1          default                          0

RP/0/RSP0/CPU0:Client#
```

# ACL vs BGP flowspec

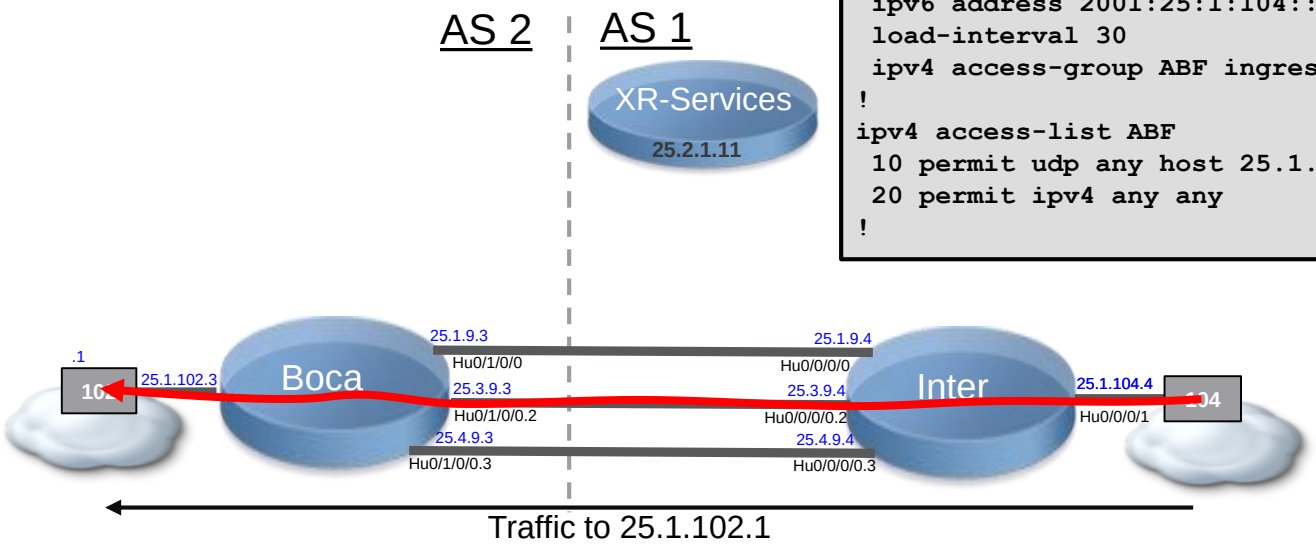
- It's important that ACL is applied before the BGP FlowSpec action.

```
RP/0/RSP0/CPU0:Client#sh int hundredGigE 0/0/0/1 accounting rates
HundredGigE0/0/0/1
                                Ingress                          Egress
  Protocol              Bits/sec         Pkts/sec     Bits/sec         Pkts/sec
  IPV4_UNICAST       5065311000           458150         1000                2

RP/0/RSP0/CPU0:Client#sh flowspec ipv4 detail
AFI: IPv4
 Flow           :Dest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550
  Actions       :Nexthop: 25.3.9.3  (bgp.1)
  Statistics                        (packets/bytes)
   Matched              :                   0/0
   Dropped              :                   0/0

RP/0/RSP0/CPU0:Client#sh access-lists ipv4 INFRA-ACL hardware ingress location 0/0/CPU0

ipv4 access-list INFRA-ACL
 10 deny udp any host 25.1.102.1 counter INFRA-ACL-COUNT (230292976 hw matches)
 20 permit ipv4 any any
RP/0/RSP0/CPU0:Client#
```

# ACL-Based Fwd (PBR) vs BGP flowspec

- Which one will take precedence ?
  Before applying the BGP FS rule, on the Client side:

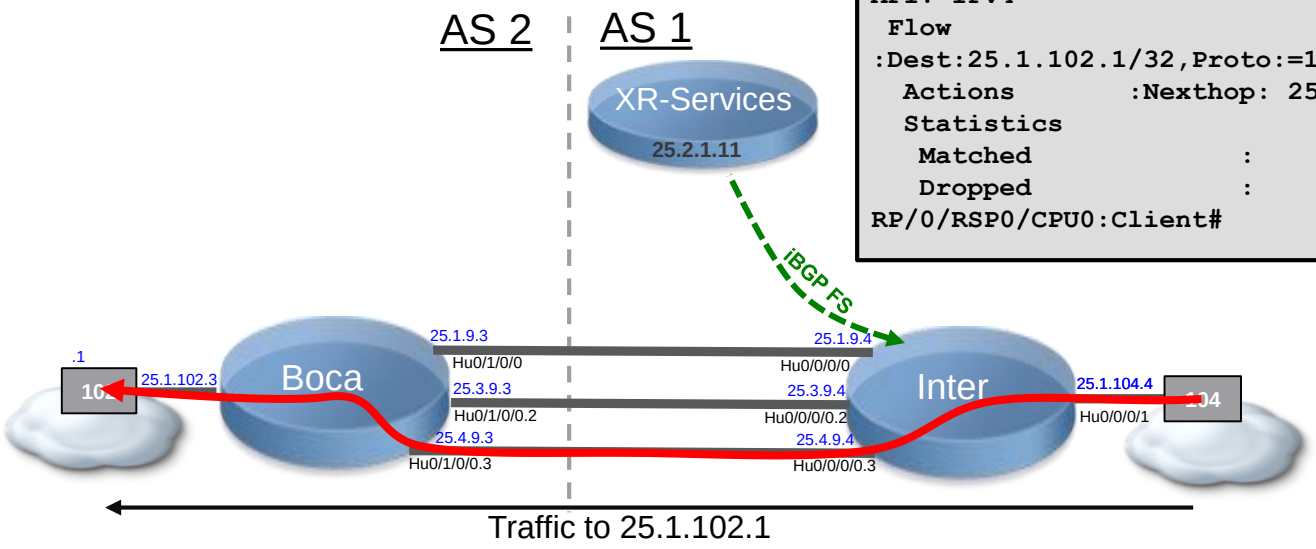
```
interface HundredGigE0/0/0/1
 ipv4 address 25.1.104.4 255.255.255.0
 ipv6 address 2001:25:1:104::4/64
 load-interval 30
 ipv4 access-group ABF ingress
!
ipv4 access-list ABF
 10 permit udp any host 25.1.102.1 nexthop1 ipv4 25.3.9.3
 20 permit ipv4 any any
!
```



# ACL-Based Fwd (PBR) vs BGP flowspec

- BGP FlowSpec action takes precedence over ABF/PBR
  After applying the rule, traffic follows the BGP FlowSpec Redirect action.

```
RP/0/RSP0/CPU0:Client#sh flowspec ipv4 detail

AFI: IPv4
 Flow
:Dest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550
  Actions        :Nexthop: 25.4.9.3   (bgp.1)
  Statistics                          (packets/bytes)
   Matched                :            2217686/3104760400
   Dropped                :                  0/0
RP/0/RSP0/CPU0:Client#
```

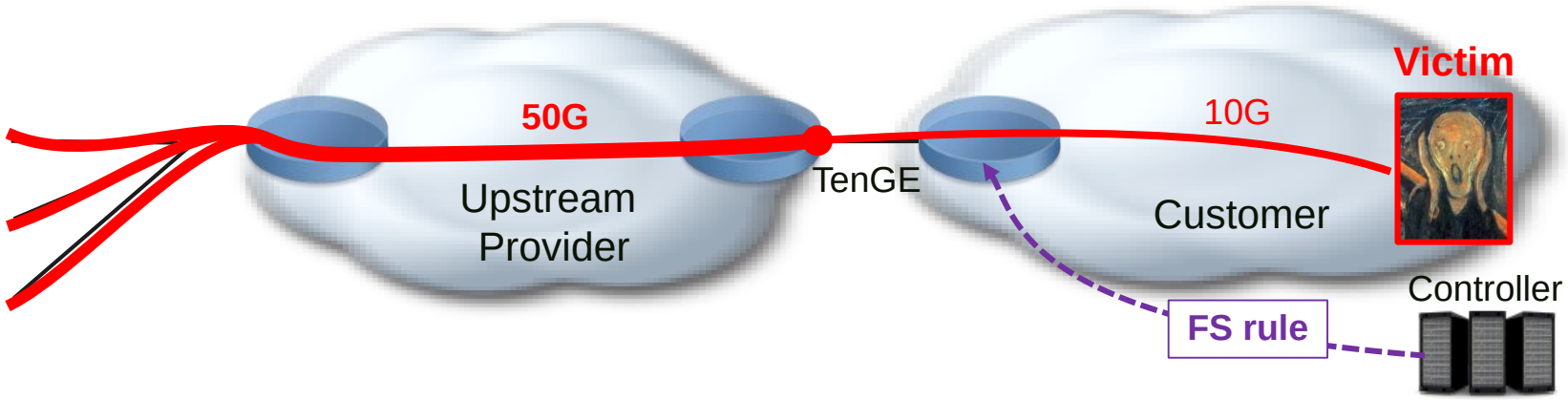


# Caveats and Limitations

# Too Late ?

## Upstream Link Saturated

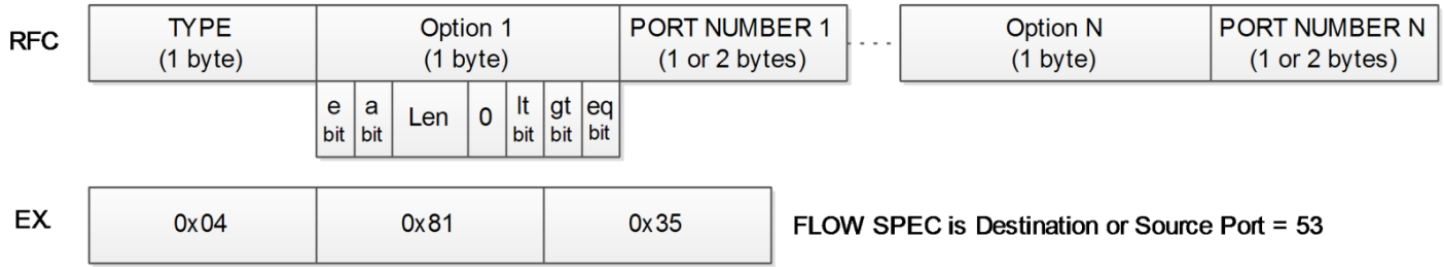
- Using BGP FS in the limit of your AS only can be too late



# Configuring a Type 4 Match “Source or Dest Ports”

- We can receive Type4 messages on client but can not generate it on the controller due to C3PL limitation

```
RP/0/0/CPU0:Ctrl(config)#show config failed
<SNIP>
class-map type traffic match-any MATCH-TYPE-4
 match source-port 123
 match destination-port 123
 end-class-map
!
!!% Policy manager does not support this feature: Match all is the only mode supported
for match type "source-port" in class-map type "traffic"
End
```



# Rate-limiter Shared per NPU

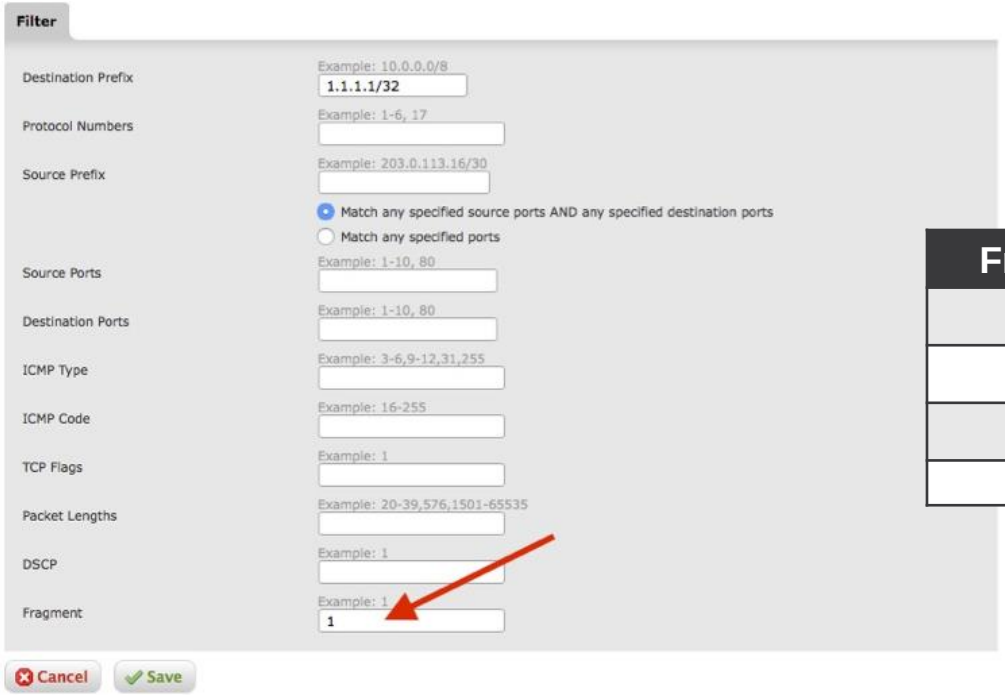
- A policer action will be applied at the NPU level and not at the port level
- Ex: you receive a 50Mbps police action, and FS is activated on three ports
  - Te0/1/0/18 is assigned to one NPU
  - Te0/1/0/10 and Te0/1/0/11 are assigned to a different NPU

| Traffic Item | Tx Frames | Rx Frames | Frames Delta | Loss % | Tx Rate (Mbps) | Rx Rate (Mbps) |
|---|---|---|---|---|---|---|
| ASR9K 10G te0/1/0/18 | 13,707,858 | 10,038,860 | 3,668,998 | 26.766 | 2,000.000 | 50.061 |
| ASR9K 10G te0/1/0/11 | 13,707,858 | 9,991,789 | 3,716,069 | 27.109 | 2,000.000 | 24.549 |
| ASR9K 10G te0/1/0/10 | 13,707,858 | 9,998,118 | 3,709,740 | 27.063 | 2,000.000 | 25.748 |

- We apply the policer per NPU
  - Traffic on Te0/1/0/18 is rate limited to 50Mbps
  - Total traffic on Te0/1/0/10+Te0/1/0/11 is rate-limit to 50Mbps, hence 25Mbps each
- Not relevant if the action is drop

# Description of Fragmentation

- ASR9000 only matches traffic on the indication of the fragmentation:
  - With first-fragment and is-fragment
  - Not with last-fragment nor do-not-fragment




| Frag | Description | Supported? |
|---|---|---|
| 1 | Don't Fragment | No |
| 2 | Is a Fragment | Yes |
| 4 | First Fragment | Yes |
| 8 | Last Fragment | No |

# ICMP Lists and Ranges

- FlowSpec rules for ICMP can only support one type and code
- No support for lists or ranges
  - Decoded but not programmed in hardware

```
RP/0/RSP0/CPU0:Client#sh bgp ipv4 flowspec
Network            Next Hop            Metric LocPrf Weight Path
*>iICMPType:=1|=2|=3|=4|=5,ICMPCode:=1/112
                     0.0.0.0                        100      0 i

Processed 1 prefixes, 1 paths
RP/0/RSP0/CPU0:Client#show policy-map transient type pbr pmap-name __bgpfs_default_IPv4
policy-map type pbr __bgpfs_default_IPv4
 handle:0x36000002
 table description: L3 IPv4 and IPv6
 class handle:0xf6000002  sequence 4294967295 (class-default)
 !
 end-policy-map
RP/0/RSP0/CPU0:Client#sh flowspec ipv4 internal | i Match Unsupported
  Match Unsupported:        ICMP type count exceeded
RP/0/RSP0/CPU0:Client#
```

Workaround: configure multiple rules

# Filter with Inter-AS BGP FlowSpec

- We support NLRI filtering on source and destination
- But we don’t filter on action type
- Customer could potentially
  - Redirect their traffic to a NH address or force the leaking into a different VRF
  - Remark all their traffic to an EF class, potentially giving them higher priority in case of congestion

# Per Interface Selection

- Today implementation is binary, BGP FS applied or not applied on an interface
- XR: No current way to decide which FS rule should be applied on which interface
- XE: interface-set draft is supported

# Conclusion

# BGP FlowSpec in SP Security

- Very powerful addition to your countermeasure tools
- Large adoption now in the industry
- Interoperable, Standard-based solution to remotely program actions on precisely identified flows
- Particularly useful in DDoS mitigation architectures
  - Filtering the stateless attacks on the Edge router, it offloads the scrubbing devices
  - Allow redirection of only the attack traffic into the scrubbing device
- But new use-cases are emerging
- You can start learning right now in a virtual environment:
  - XRv 9000 can be used as a controller, CSR1000v can be used as a client

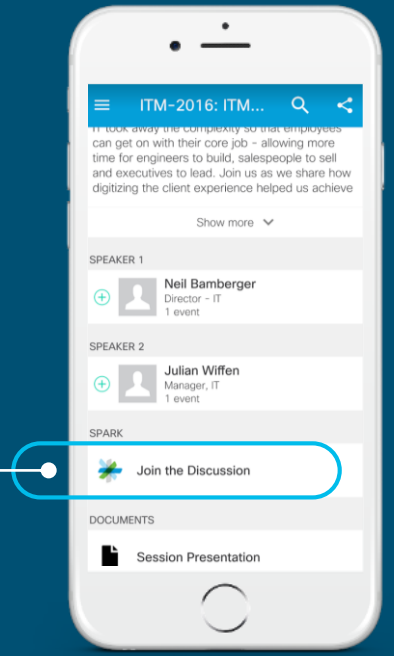
# Cisco Spark

## Questions?

Use Cisco Spark to communicate with the speaker after the session

## How

1. Find this session in the Cisco Live Mobile App
2. Click “Join the Discussion”
3. Install Spark or go directly to the space
4. Enter messages/questions in the space

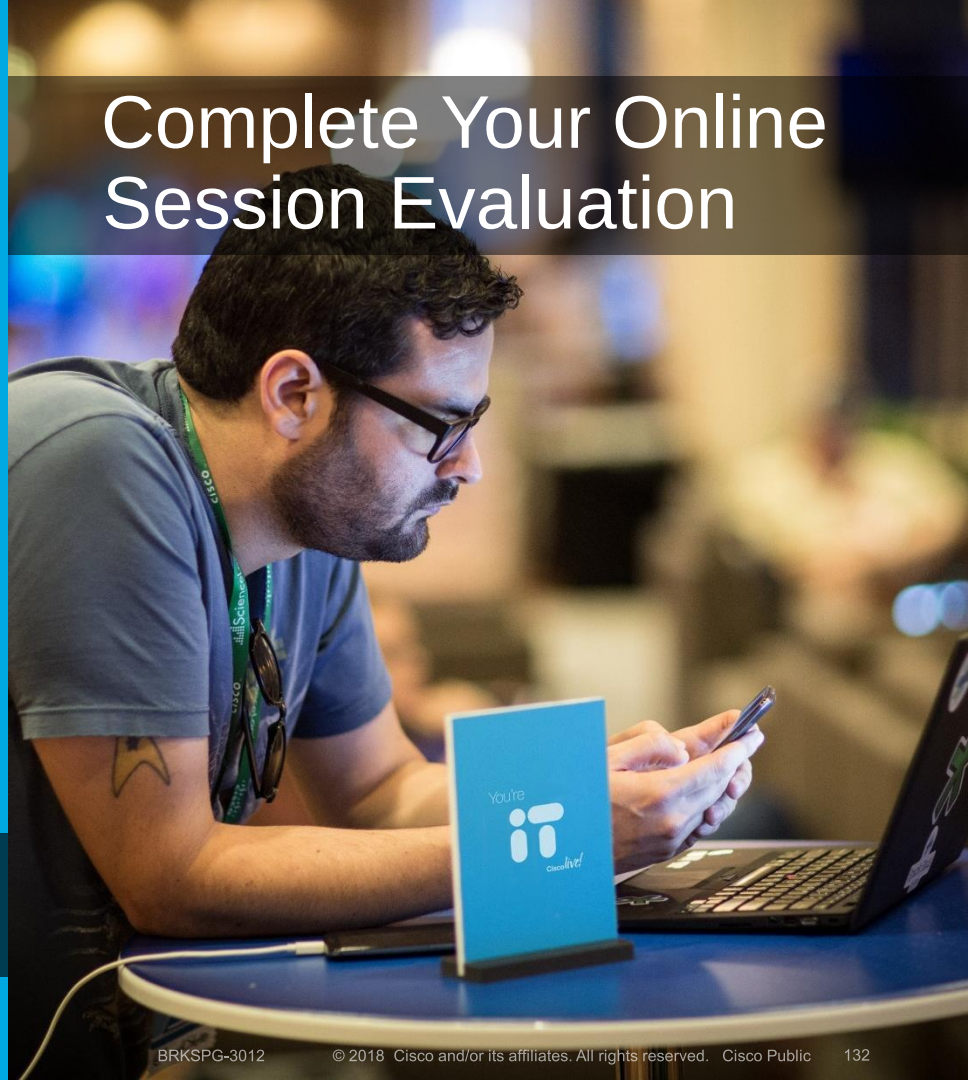



cs.co/ciscolivebot#BRKSPG-3012

# Complete Your Online Session Evaluation

- Please complete your Online Session Evaluations after each session
- Complete 4 Session Evaluations & the Overall Conference Evaluation (available from Thursday) to receive your Cisco Live T-shirt
- All surveys can be completed via the Cisco Live Mobile App or the Communication Stations

Don’t forget: Cisco Live sessions will be available for viewing on-demand after the event at www.ciscolive.com/global/on-demand-library/.



# Continue Your Education

- Demos in the Cisco campus
- Walk-in Self-Paced Labs
- Lunch & Learn
- Meet the Engineer 1:1 meetings
- Related sessions

# Thank you

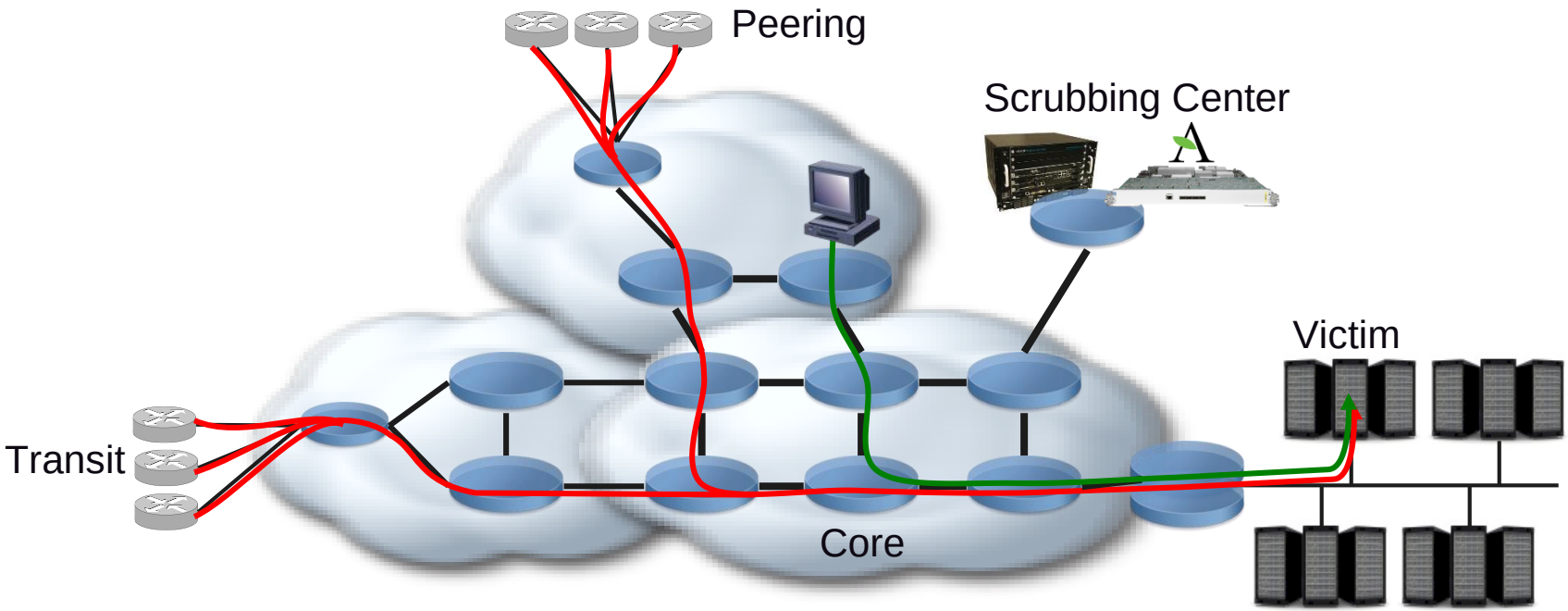

You're
Cisco live!

# Back-Up Slides
# Other Use-Cases

# DDoS Mitigation Models

## Centralized

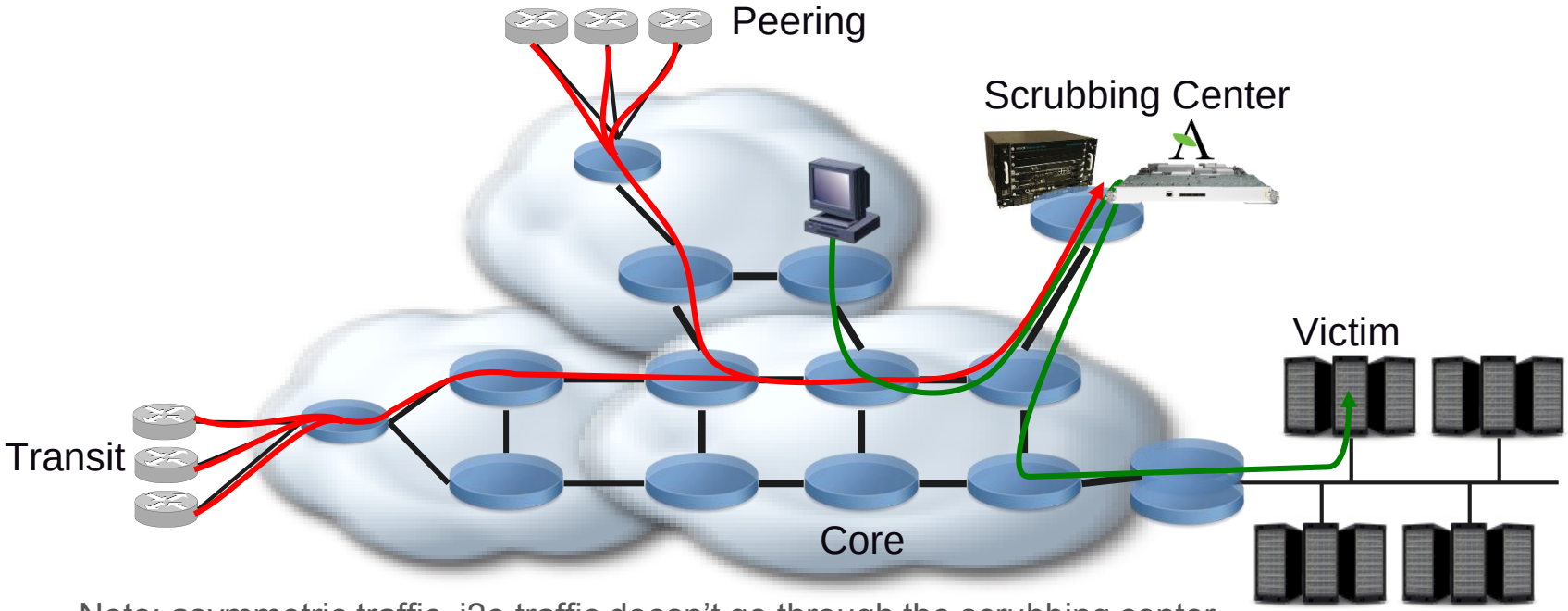
- A central point in the network is dedicated for hosting scrubbing devices



# DDoS Mitigation Models

## Centralized

- Traffic target to the victim is diverted to this place for analysis

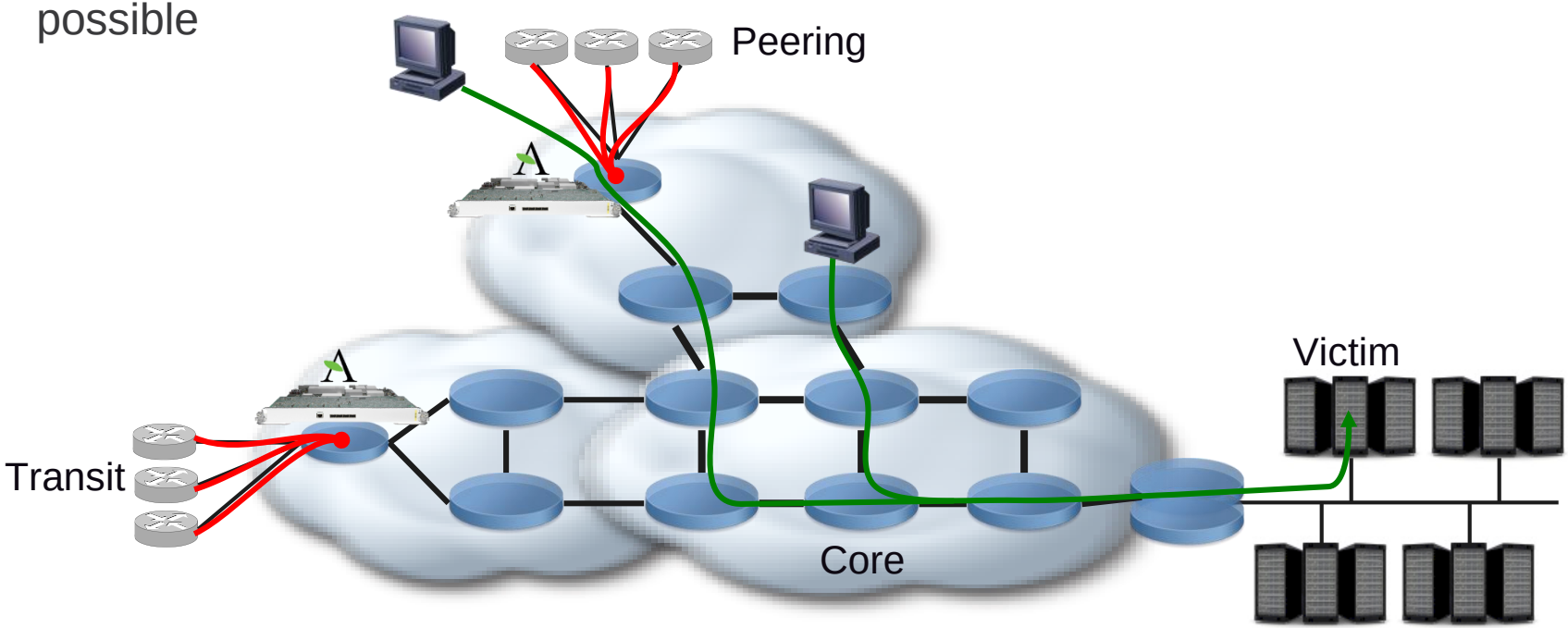



Note: asymmetric traffic, i2o traffic doesn't go through the scrubbing center

# DDoS Mitigation Models

## Distributed

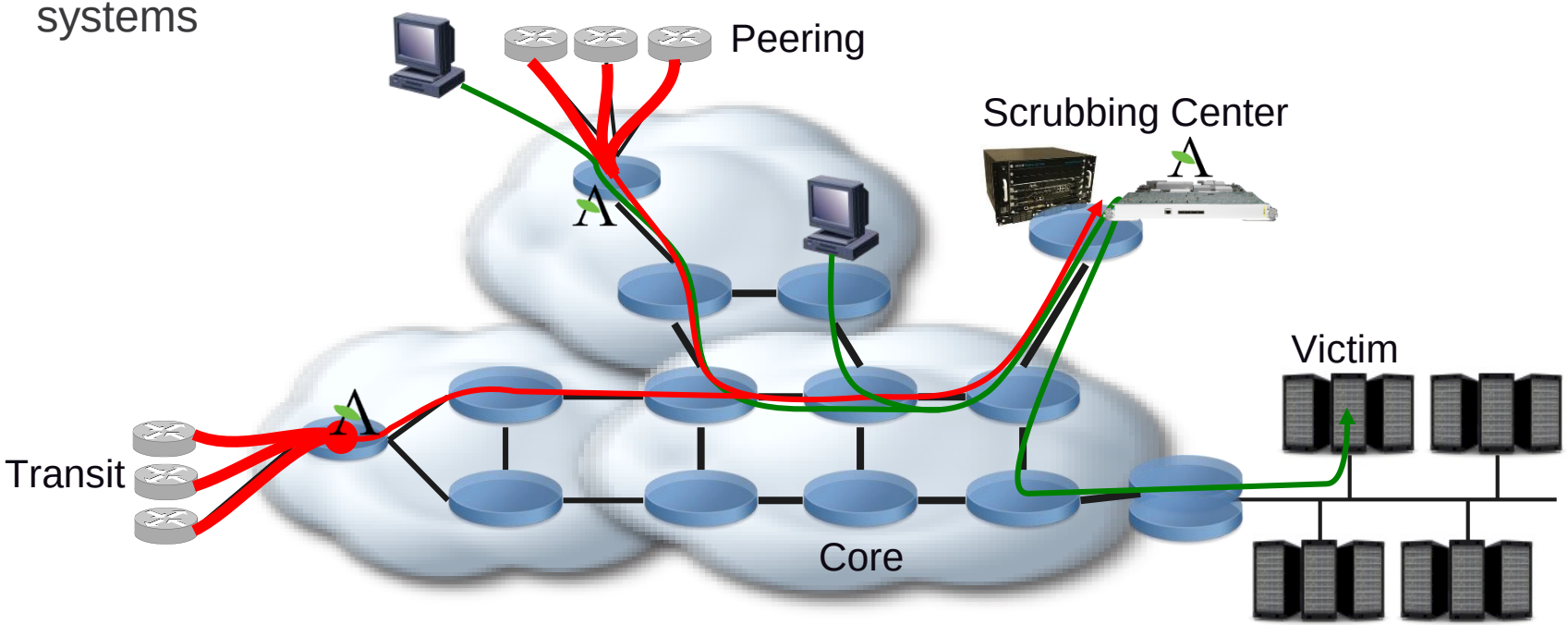
- We install scrubbers at the edge of the backbone to tackle the attack as early as possible



# DDoS Mitigation Models

## Mixed

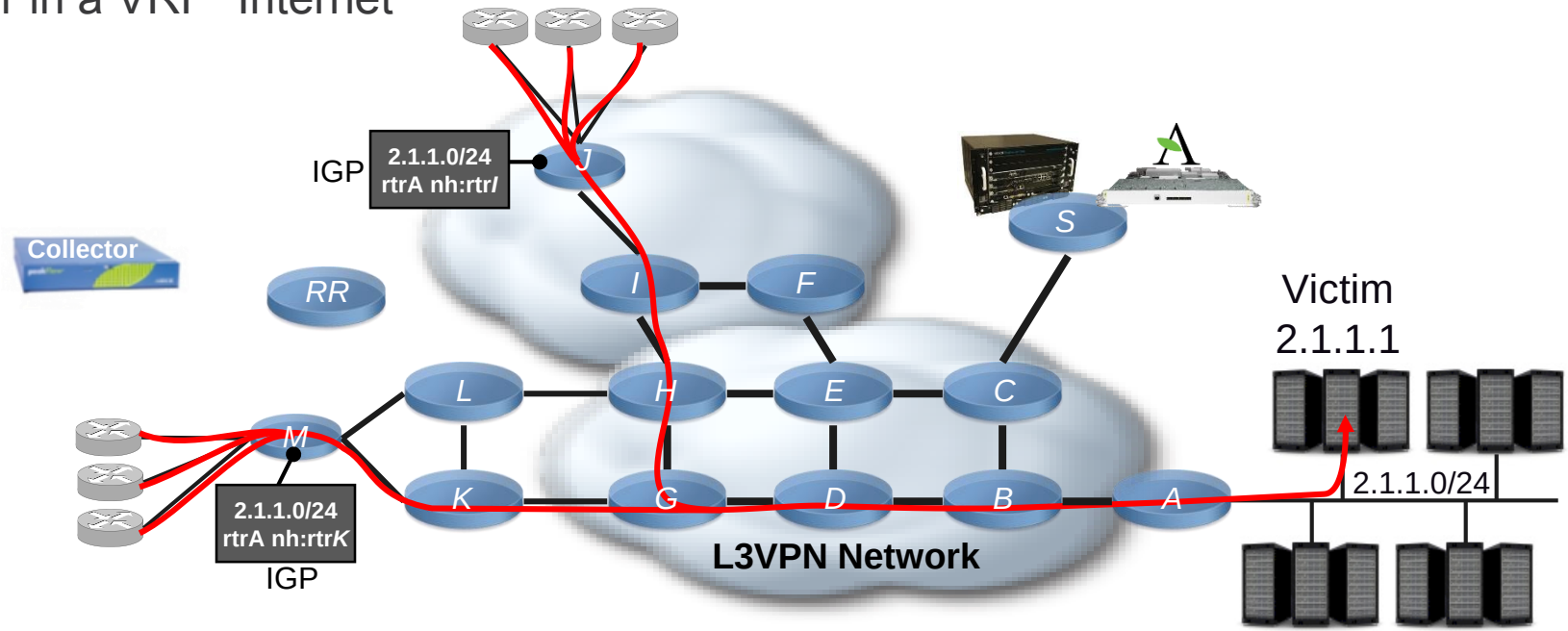
- Specific attacks can be handled in the central point or to off-load the edge systems



# L3VPN Network w/ Scrubbing Center

## Currently deployed

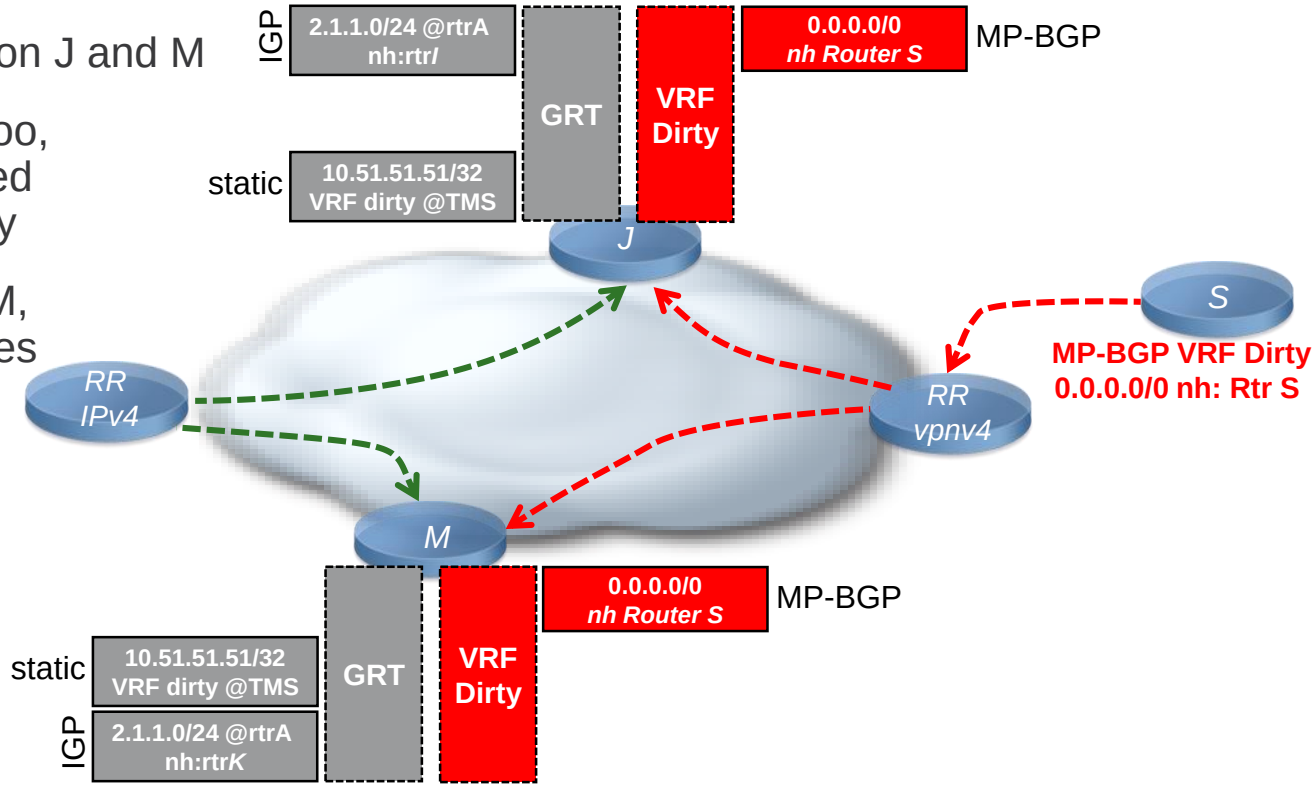
- 2.1.1.1 is victim of a  large size SYN attack. Traffic is transported in the GRT or in a VRF “Internet”



# L3VPN Network w/ Scrubbing Center

## Currently deployed

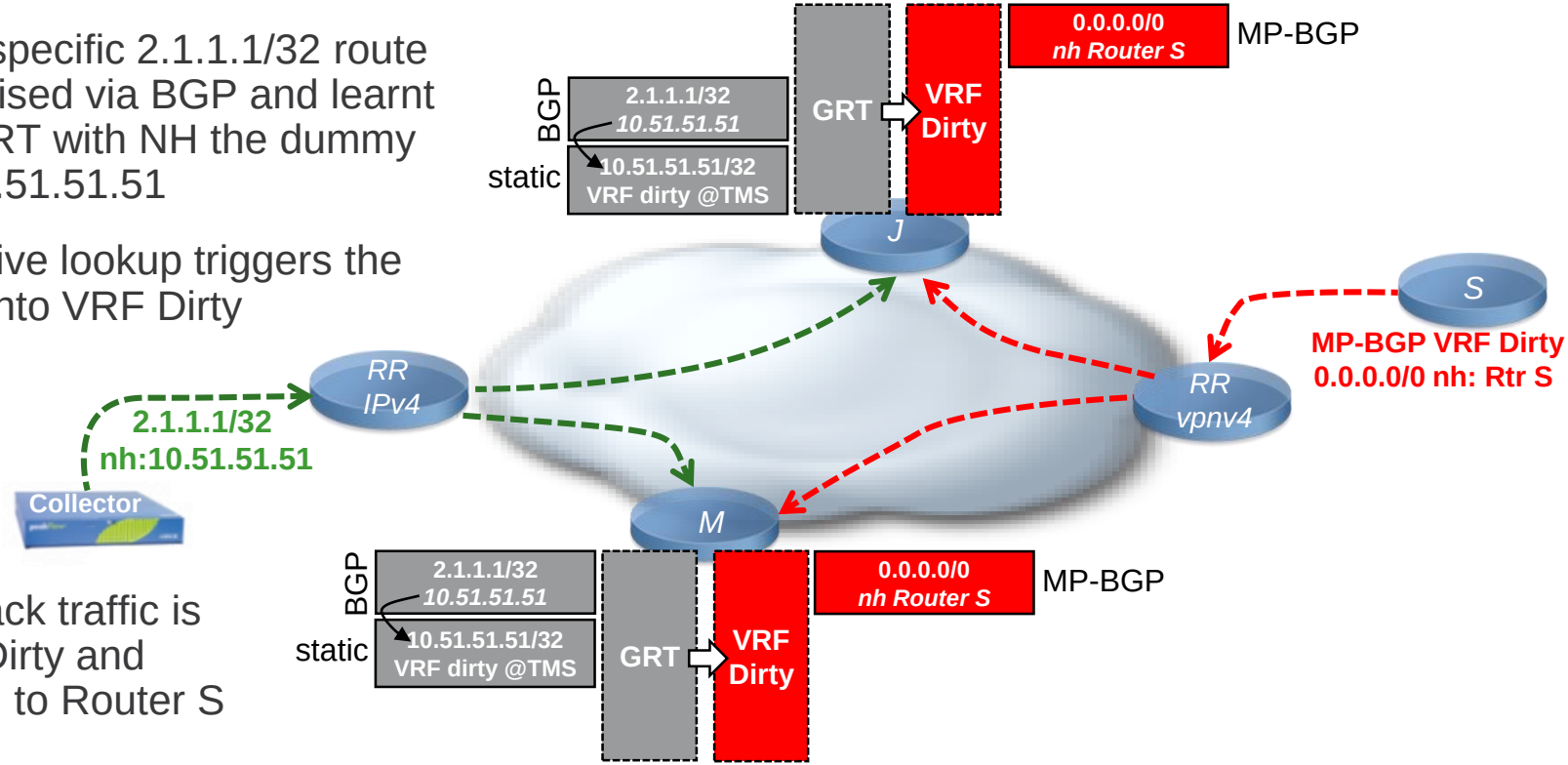
- VRF Dirty is configured on J and M
- MP-BGP is configured too, default route is advertised from @TMS in VRF Dirty
- On edge routers J and M, we configure static entries for a dummy host route (10.51.51.51/32) with a NH in VRF Dirty.
  If matched, traffic will leak into this VRF Dirty
- Now, traffic to 2.1.1.1 uses the IGP route 2.1.1.0/24



# L3VPN Network w/ Scrubbing Center

## Currently deployed

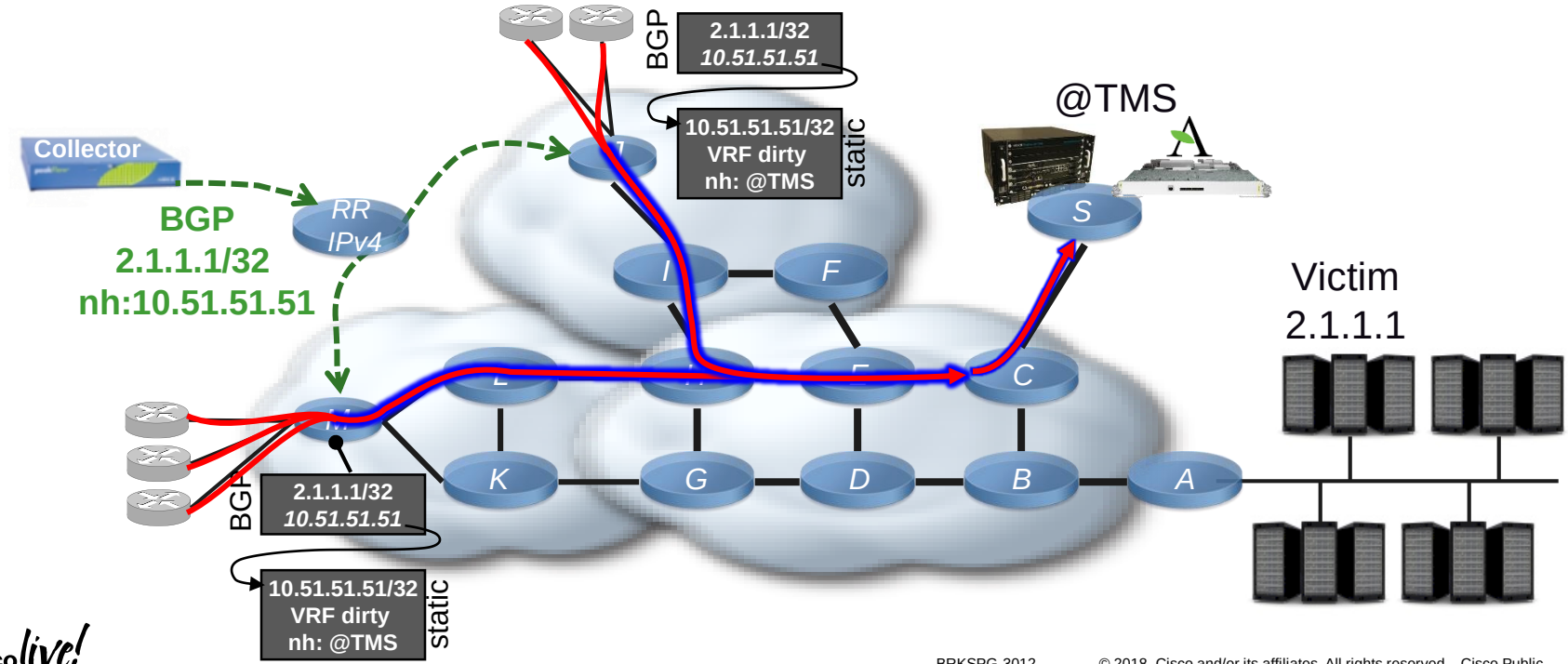
- A more specific 2.1.1.1/32 route is advertised via BGP and learnt in the GRT with NH the dummy route 10.51.51.51
- A recursive lookup triggers the leaking into VRF Dirty
- Now attack traffic is in VRF Dirty and attracted to Router S



# L3VPN Network w/ Scrubbing Center

## Currently deployed

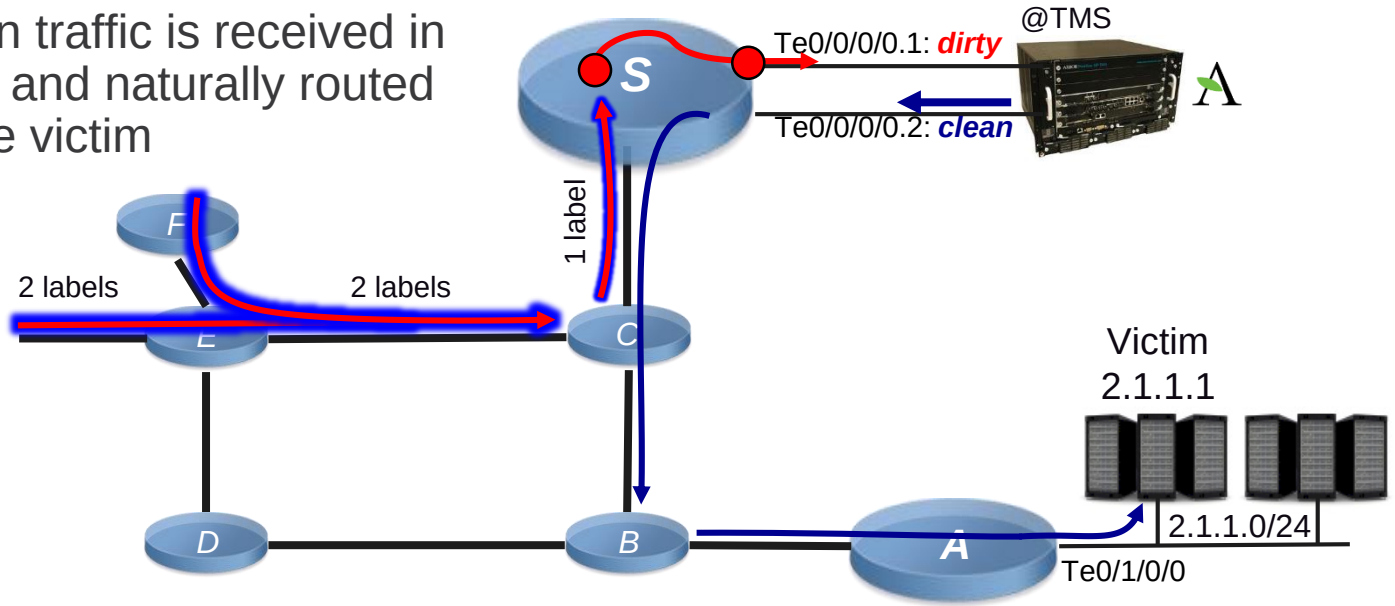
- CP advertises a BGP route for 2.1.1.1/32 with next-hop the dummy 10.51.51.51



# L3VPN Network w/ Scrubbing Center

## Currently deployed

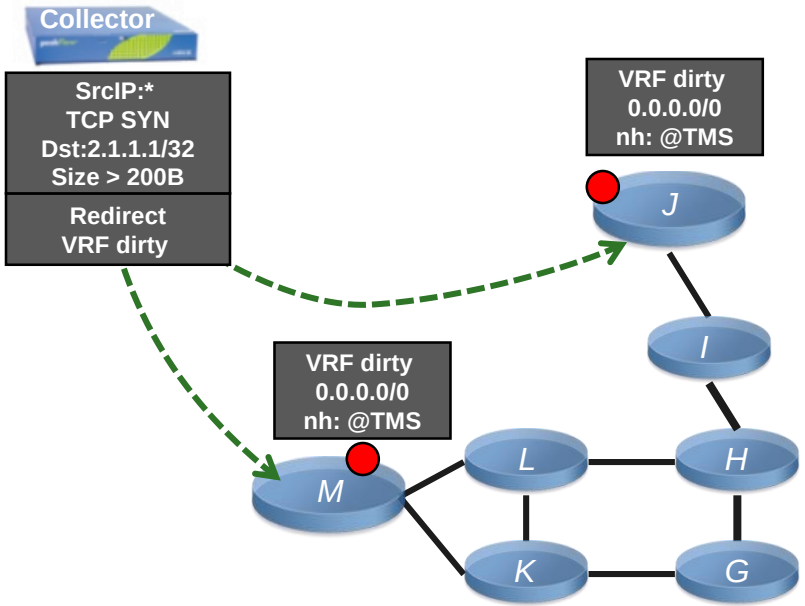
- Traffic with a VRF label Dirty is dragged to router S
- Router S is pushing unlabeled traffic to the TMS via an interface in VRF Dirty
- Clean traffic is received in GRT and naturally routed to the victim



# L3VPN Network w/ Scrubbing Center

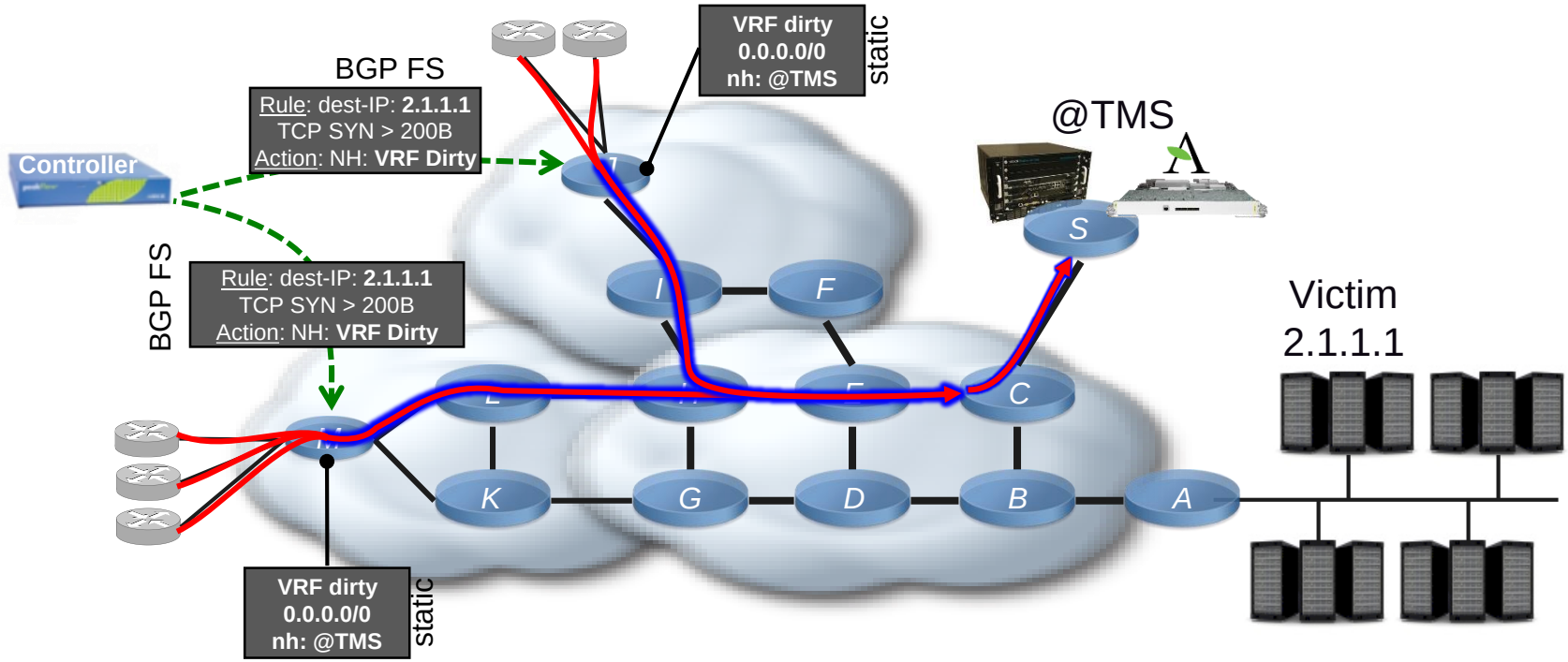
## Improved with BGP FlowSpec

- BGP FlowSpec inject rules to redirect attack traffic into VRF dirty
- No more dummy route needed
- Only a default route in dirty VRF is needed to reach the scrubber
- More granular “matching” parameters: only the packets with specific protocol/port/packet-size/etc are diverted in Dirty VRF



# L3VPN Network w/ Scrubbing Center

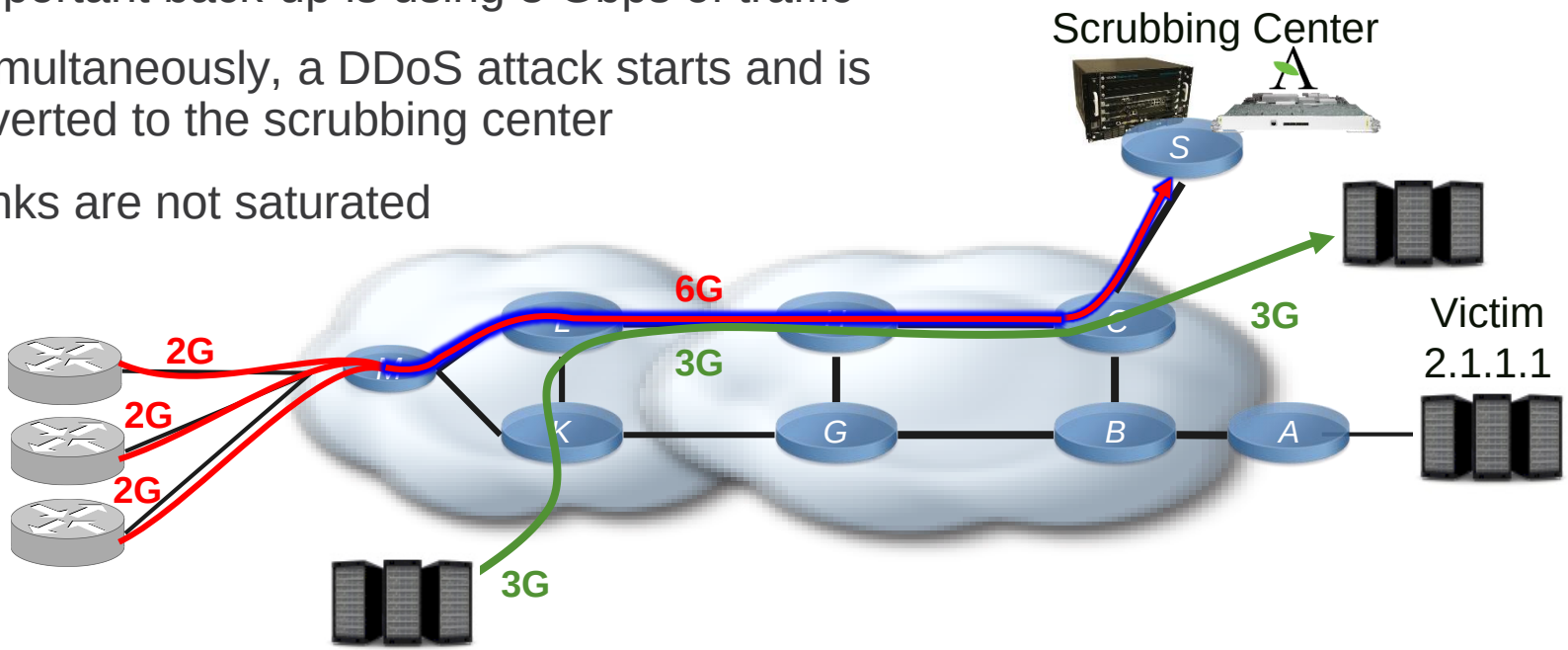
## Improved with BGP FlowSpec



# Other BGP FS Use-Cases

## Low QoS Priority Traffic for DDoS Attacks

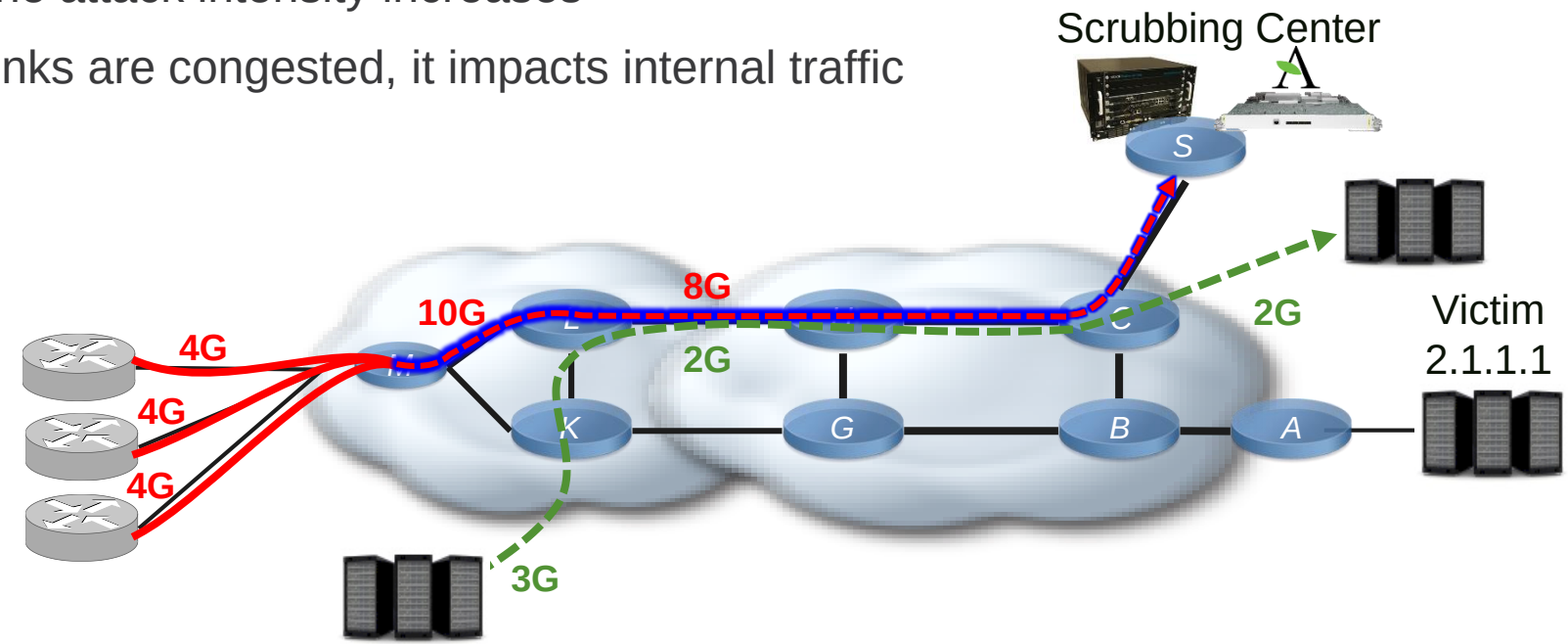
- Important back-up is using 3 Gbps of traffic
- Simultaneously, a DDoS attack starts and is diverted to the scrubbing center
- Links are not saturated



# Other BGP FS Use-Cases

## Low QoS Priority Traffic for DDoS Attacks

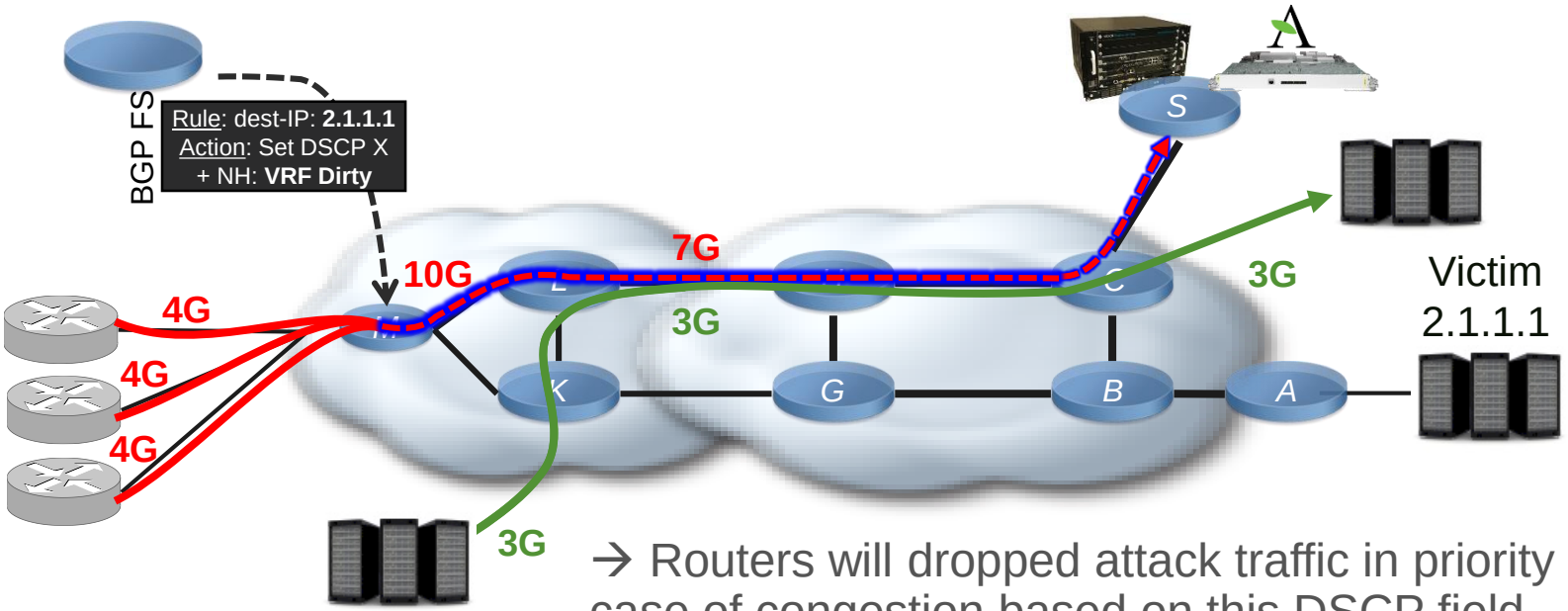
- The attack intensity increases
- Links are congested, it impacts internal traffic



# Other BGP FS Use-Cases

## Low QoS Priority Traffic for DDoS Attacks w/ Flowspec

- BGP FS rule forces the route leaking in VRF-Dirty and positioning a DSCP field




→ Routers will dropped attack traffic in priority in case of congestion based on this DSCP field

# Back-Up Slides Configuration

For Your Reference

# Configuring BGP FlowSpec

## Configuring a Type 1 Match “Destination Address”

```
RP/0/0/CPU0:Ctrl(config)#class-map type traffic match-all MATCHING-RULE
RP/0/0/CPU0:Ctrl(config-cmap)#match destination-address ipv4 81.253.193.0/24
RP/0/0/CPU0:Ctrl(config-cmap)#
```

```
RP/0/RP0/CPU0:Client#show contr pse tcam summary location 0/0/CPU0

<SNIP>

TCAM Device Information for Ingress PSE, CAM bank 1:
Device size: 20M (256K array entries of 80-bits), 261122 available
Current mode of operation: Turbo
<SNIP>
Feature specific information:
<SNIP>
        FlowSpec IPv4 (id 32):
                Owner client id: 20.   Limit 245760 cells
                Total 1 regions using 4 CAM cells
<SNIP>
```

# Configuring BGP FlowSpec

## Configuring a Type 2 Match “Source Address”

```
RP/0/0/CPU0:Ctrl(config)#class-map type traffic match-all MATCHING-RULE
RP/0/0/CPU0:Ctrl(config-cmap)#match source-address ipv4 2.2.0.0/16
RP/0/0/CPU0:Ctrl(config-cmap)#
```

```
RP/0/RP0/CPU0:Client#sh flowspec ipv4 detail

AFI: IPv4
 Flow           :Source:2.2.0.0/16
  Actions       :Traffic-rate: 100000 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :                       0/0
   Transmitted          :                       0/0
   Dropped              :                       0/0
RP/0/RP0/CPU0:Boca#sh flowspec ipv4 nlri

AFI: IPv4
 NLRI (Hex dump) :       0x02100202
  Actions       :Traffic-rate: 100000 bps  (bgp.1)
RP/0/RP0/CPU0:Boca#
```

| Type | Prefix length | Prefix |
|---|---|---|
| 1 byte | 1 byte | Variable |
| 2 | /16 | 2.2 |
| 0x **02** | 0x **10** | 0x **02 02** |

**0x02100202**

# Configuring BGP FlowSpec

## Configuring a Type 3 Match “IPv4 Protocol Type” / “IPv6 Next Header”

```
RP/0/0/CPU0:Ctrl(config)#class-map type traffic match-all MATCHING-RULE
RP/0/0/CPU0:Ctrl(config-cmap)#match protocol udp tcp
RP/0/0/CPU0:Ctrl(config-cmap)#
```

```
RP/0/RP0/CPU0:Client#sh flowspec ipv4 detail

AFI: IPv4
 Flow           :Proto:=0|=17|=6
  Actions       :Traffic-rate: 100000 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :                     0/0
   Transmitted          :                     0/0
   Dropped              :                     0/0
RP/0/RP0/CPU0:Client#sh flowspec ipv4 nlri

AFI: IPv4
 NLRI (Hex dump) :       0x03010001118106
  Actions       :Traffic-rate: 100000 bps  (bgp.1)
RP/0/RP0/CPU0:Client#
```

| Option Byte | | | | | | |
|---|---|---|---|---|---|---|
| End | And | Len | 0 | Lt “<“ | Gt “>” | Eq “=“ |
| 1b | 1b | 2b | 1 b | 1b | 1b | 1b |

**0x03010001118106**

| Type | Option1 | IP proto1 | Option2 | IP proto2 | Option3 | IP proto3 |
|---|---|---|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte | 1 byte | 1 byte | 1 byte |
| 1 | 0b00000001 | 0x00 | 0b00000001 | 17 = 0x11 | 0b10000001 | 0x06 |
| 0x **03** | **01** | **00** | **01** | **11** | **81** | **06** |

# Configuring BGP FlowSpec

## Configuring a Type 5 Match “Destination Port”

```
RP/0/0/CPU0:Ctrl(config)#class-map type traffic match-all MATCHING-TYPE5
RP/0/0/CPU0:Ctrl(config-cmap)#match destination-port 80 443 8080
RP/0/0/CPU0:Ctrl(config-cmap)#
```

```
RP/0/RP0/CPU0:Client#show flowspec afi-all detail
AFI: IPv4
 Flow           :DPort:=80|=443|=8080
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
  Statistics                        (packets/bytes)
   Matched              :                 0/0
   Transmitted          :                 0/0
   Dropped              :                 0/0
RP/0/RP0/CPU0:Client#show flowspec ipv4 nlri
AFI: IPv4
 NLRI (Hex dump) :       0x0501501101bb911f90
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
RP/0/RP0/CPU0:Client#
```

| Option Byte | | | | | | | |
|---|---|---|---|---|---|---|---|
| | End | And | Len | 0 | Lt “<“ | Gt “>” | Eq “=“ |
| 01 | 0 | 0 | 00 | 0 | 0 | 0 | 1 |
| 11 | 0 | 0 | 01 | 0 | 0 | 0 | 1 |
| 91 | 1 | 0 | 01 | 0 | 0 | 0 | 1 |

| Type (1B) | Option x (1B) | Dest Port (1B or 2B) |
|---|---|---|
| 5 | equal/length=0<br>Not last | d80 = x50 |
| 0 **x05** | **0x01** | **0x50** |
| - | equal/length=1<br>Not last | d443 = x1BB |
| - | **0x11** | **0x01BB** |
| - | equal/length=1<br>last | d8080 = x1F90 |
| - | **0x91** | **0x1F90** |

**0x05**01501101bb911f90

# Configuring BGP FlowSpec

## Configuring a Type 6 Match “Source Port”

```
RP/0/0/CPU0:Ctrl(config)#class-map type traffic match-all MATCHING-TYPE6
RP/0/0/CPU0:Ctrl(config-cmap)#match source-port 80-100
RP/0/0/CPU0:Ctrl(config-cmap)#
```

```
RP/0/RP0/CPU0:Client#sh flowspec ipv4 detail
AFI: IPv4
 Flow           :SPort:>=80&<=100
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :                     0/0
   Transmitted          :                     0/0
   Dropped              :                     0/0
RP/0/RP0/CPU0:Client#sh flowspec ipv4 nlri
AFI: IPv4
 NLRI (Hex dump) :       0x060350c564
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
RP/0/RP0/CPU0:Client#
```

| Type (1B) | Option 1 (1B) | Dest Port |
|---|---|---|
| 6 | 0000 0011<br>greater+equal/le=0/not last | 80 |
| 0 **x06** | **0x03** | **0x50** |
| - | 1100 0101<br>lower+equal/le=0/last | 100 |
| - | **0xc5** | **0x64** |

**0x060350c564**

| Option Byte | | | | | | | |
|---|---|---|---|---|---|---|---|
| | End | And | Len | 0 | Lt “<“ | Gt “>” | Eq “=“ |
| 03 | 0 | 0 | 00 | 0 | 0 | 1 | 1 |
| c5 | 1 | 1 | 00 | 0 | 1 | 0 | 1 |

# Configuring BGP FlowSpec

## Configuring a Type 7+8 Match “ICMP Type” + “ICMP Code”

```
RP/0/0/CPU0:Ctrl(config-cmap)# match ipv4 icmp-type 3
RP/0/0/CPU0:Ctrl(config-cmap)# match ipv4 icmp-code 13
RP/0/0/CPU0:Ctrl(config-cmap)#commit
```

```
RP/0/RSP0/CPU0:Client#show flowspec afi-all detail
AFI: IPv4
 Flow            :ICMPType:=3,ICMPCode:=13
  Actions        :Traffic-rate: 314152 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched                :                    0/0
   Dropped                :                    0/0
RP/0/RSP0/CPU0:Client#show flowspec ipv4 nlri
AFI: IPv4
 NLRI (Hex dump) :        0x07810308810d
  Actions        :Traffic-rate: 314152 bps  (bgp.1)
RP/0/RSP0/CPU0:Client#
```

| Type (1B) | Option 1 (1B) | ICMP |
|---|---|---|
| 7 | 1000 0001 | 03 |
| 0 x07 | **0x81** | **0x03** |
| 8 | 100 0001 | 13 |
| 0 x08 | **0x81** | **0x0d** |

**0x07810308810d**

| Option Byte | | | | | | | |
|---|---|---|---|---|---|---|---|
| | End | And | Len | 0 | Lt “<“ | Gt “>” | Eq “=“ |
| 81 | 1 | 0 | 00 | 0 | 0 | 0 | 1 |

# Configuring BGP FlowSpec

## Configuring a Type 9 Match "TCP Flag Component"

```
RP/0/0/CPU0:Ctrl(config)#class-map type traffic match-all MATCHING-TYPE9
RP/0/0/CPU0:Ctrl(config-cmap)#match tcp-flag 2
RP/0/0/CPU0:Ctrl(config-cmap)#
```

```
RP/0/RP0/CPU0:Client#sh flowspec ipv4 detail
AFI: IPv4
 Flow           :TCPFlags:=0x02
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :                  8/496
Dropped              :                  0/0
RP/0/RP0/CPU0:Client#sh flowspec ipv4 nlri
AFI: IPv4
 NLRI (Hex dump) :       0x098102
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
RP/0/RP0/CPU0:Client#
```

| Type (1B) | Option 1 (1B) | Flag |
|---|---|---|
| 9 | 1000 0001 | x02 |
| 0 **x09** | **0x81** | **0x02** |

**0x098102**

| Option Byte | | | | | | | |
|---|---|---|---|---|---|---|---|
| | e bit | a bit | Len | 0 | 0 | Not bit | m bit |
| 81 | 1 | 0 | 00 | 0 | 0 | 0 | 1 |

- Ex: http://rapid.web.unc.edu/resources/tcp-flag-key/
  - **0x02: SYN**
  - 0x12: SYN-ACK
  - 0x10: ACK
  - 0x01: FIN
  - 0x04: RST

# Configuring BGP FlowSpec

## Configuring a Type 10 Match “Packet Length”

```
RP/0/0/CPU0:Ctrl(config)#class-map type traffic match-all MATCHING-TYPE10
RP/0/0/CPU0:Ctrl(config-cmap)#match packet length 100
RP/0/0/CPU0:Ctrl(config-cmap)#
```

```
RP/0/RP0/CPU0:Client#show flowspec afi-all detail
AFI: IPv4
 Flow            :Length:=100
  Actions        :Traffic-rate: 314152 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :                         0/0
   Transmitted          :                         0/0
   Dropped              :                         0/0
RP/0/RP0/CPU0:Client#show flowspec ipv4 nlri
AFI: IPv4
 NLRI (Hex dump) :       0x0a8164
  Actions        :Traffic-rate: 314152 bps  (bgp.1)
RP/0/RP0/CPU0:Client#
```

| Type (1B) | Option 1 (1B) | Pkt Length |
|---|---|---|
| 10 | 1000 0001 | 100 |
| 0 x0a | 0x81 | 0x64 |

0x0a8164

| Option Byte | | | | | | | |
|---|---|---|---|---|---|---|---|
| | End | And | Len | 0 | Lt “<“ | Gt “>” | Eq “=“ |
| 81 | 1 | 0 | 00 | 0 | 0 | 0 | 1 |

# Configuring BGP FlowSpec

## Configuring a Type 11 Match “IPv4/IPv6 DSCP”

```
RP/0/0/CPU0:Ctrl(config)#class-map type traffic match-all MATCHING-TYPE11
RP/0/0/CPU0:Ctrl(config-cmap)#match dscp ef
RP/0/0/CPU0:Ctrl(config-cmap)#
```

```
RP/0/RP0/CPU0:Client#show flowspec afi-all detail
AFI: IPv4
 Flow           :DSCP:=46
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :                       0/0
   Transmitted          :                       0/0
   Dropped              :                       0/0
RP/0/RP0/CPU0:Client#show flowspec afi-all nlri
AFI: IPv4
 NLRI (Hex dump) :       0x0b812e
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
RP/0/RP0/CPU0:Client#
```

| Type (1B) | Option 1 (1B) | DSCP |
|---|---|---|
| 11 | 1000 0001 | ef |
| 0 **x0b** | **0x81** | **0x2e** |

**0x0b812e**

| Option Byte | | | | | | | |
|---|---|---|---|---|---|---|---|
| | End | And | Len | 0 | Lt “<“ | Gt “>” | Eq “=“ |
| 81 | 1 | 0 | 00 | 0 | 0 | 0 | 1 |

# Configuring BGP FlowSpec

## Configuring a Type 12 Match “IPv4 Fragment”

```
RP/0/0/CPU0:Ctrl(config)#class-map type traffic match-all MATCHING-TYPE12
RP/0/0/CPU0:Ctrl(config-cmap)#match fragment-type is-fragment last-fragment
RP/0/0/CPU0:Ctrl(config-cmap)#
```

```
RP/0/RP0/CPU0:Client#sh flowspec ipv4 detail
AFI: IPv4
 Flow            :Frag:=LF:IsF
  Actions        :Traffic-rate: 314152 bps   (bgp.1)
  Statistics                              (packets/bytes)
   Matched                :                       0/0
   Transmitted            :                       0/0
   Dropped                :
0/0RP/0/RP0/CPU0:Client#sh flowspec ipv4 nlri
AFI: IPv4
 NLRI (Hex dump) :        0x0c810a
  Actions        :Traffic-rate: 314152 bps   (bgp.1)
RP/0/RP0/CPU0:Client#
```

| Type (1B) | Option 1 (1B) | Pkt Length |
|---|---|---|
| 11 | 1000 0001 | LF + IsF |
| 0 **x0c** | **0x81** | **0x0a** |

**0x0c810a**

| Option Byte | | | | | | | |
|---|---|---|---|---|---|---|---|
| | End | And | Len | 0 | Lt “<“ | Gt “>” | Eq “=“ |
| 81 | 1 | 0 | 00 | 0 | 0 | 0 | 1 |

| Bitmask | | | | | | | |
|---|---|---|---|---|---|---|---|
| | 0 | 0 | 0 | lf | ff | isf | df |
| 0a | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

For Your Reference

# Action Redirect: Digging Deeper

## Controller Configuration

```
policy-map type pbr test
 class type traffic test
  redirect nexthop route-target 1:1
 !
end-policy-map
```

## Client View (Debug all Flowspec Events)

```
bgp[1052]: FlowSpec: Updating NLRI Proto:=6,DPort:=80 for TBL default:IPv4.
flowspec_mgr[1094]: FlowSpec: Client bgp.1 NLRI Proto:=6,DPort:=80 Update for TBL default:IPv4.
flowspec_mgr[1094]: FlowSpec: Registered for AFI IPv4 RT ASN2-1:1.
flowspec_mgr[1094]: FlowSpec: Added client bgp.1 flow active Proto:=6,DPort:=80 with actions RT-ASN2-1:1 from TBL
default:IPv4.
flowspec_mgr[1094]: FlowSpec: Finished receving 1 IPC msgs for conn 0x20000099, 0:No error.
bgp[1052]: FlowSpec: Notifying client bgp.1 for Register RT ASN2-1:1 (AFI IPv4).
```

In this case, we used 2-byte long ASN for the Route Target definition.

It's transported with extended community 0x8008

For Your Reference

# Action Redirect: Digging Deeper

## Controller Configuration

```
policy-map type pbr test
 class type traffic test
  redirect nexthop route-target 123456789:1
 !
end-policy-map
```

## Client View (Debug all Flowspec Events)

```
bgp[1052]: FlowSpec: Updating NLRI Proto:=6,DPort:=80 for TBL default:IPv4.
flowspec_mgr[1094]: FlowSpec: Client bgp.1 NLRI Proto:=6,DPort:=80 Update for TBL default:IPv4.
flowspec_mgr[1094]: FlowSpec: Registered for AFI IPv4 RT ASN4-123456789:1.
flowspec_mgr[1094]: FlowSpec: Added client bgp.1 flow active Proto:=6,DPort:=80 with actions RT-ASN4-123456789:1
from TBL default:IPv4.
bgp[1052]: FlowSpec: Notifying client bgp.1 for Register RT ASN4-123456789:1 (AFI IPv4).
flowspec_mgr[1094]: FlowSpec: Finished receving 1 IPC msgs for conn 0x20000099, 0:No error.
```

In this case, we used 4-byte long ASN for the Route Target definition.

It's transported with extended community 0x8208

# Back-Up Slides
# Monitoring

# Show Commands to Check BGP Flowspec Operation

- First, we verify the BGP session for the address-family Flowspec

```
RP/0/RP0/CPU0:Client#show bgp ipv4 flowspec

BGP router identifier 3.3.3.3, local AS number 2
BGP generic scan interval 60 secs
Non-stop routing is enabled
BGP table state: Active
Table ID: 0x0   RD version: 16
BGP main routing table version 16
BGP NSR Initial initsync version 0 (Reached)
BGP NSR/ISSU Sync-Group versions 16/0
BGP scan interval 60 secs

Status codes: s suppressed, d damped, h history, * valid, > best
              i - internal, r RIB-failure, S stale, N Nexthop-discard
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network            Next Hop            Metric LocPrf Weight Path
*> SPort:=80/24       0.0.0.0                                  0 1 i

Processed 1 prefixes, 1 paths
RP/0/RP0/CPU0:Client#
```

# Configuring BGP FlowSpec on IOS XR Routers

## Verifying the Session Establishment (on Client)

```
RP/0/RP0/CPU0:Client#sh bgp ipv4 flowspec summary

BGP router identifier 3.3.3.3, local AS number 1
BGP generic scan interval 60 secs
Non-stop routing is enabled
BGP table state: Active
Table ID: 0x0   RD version: 7072
BGP main routing table version 7072
BGP NSR Initial initsync version 0 (Reached)
BGP NSR/ISSU Sync-Group versions 7072/0
BGP scan interval 60 secs

BGP is operating in STANDALONE mode.

Process       RcvTblVer   bRIB/RIB   LabelVer  ImportVer  SendTblVer  StandbyVer
Speaker            7072       7072       7072       7072        7072        7072

Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
25.2.1.11         0     1  106269  105679     7072    0    0     1w1d        1001

RP/0/RP0/CPU0:Client#
```

# Show Commands

- Then, we can get more details for this particular rule

```
RP/0/RP0/CPU0:Client#show bgp ipv4 flowspec SPort:=80/24 detail

BGP routing table entry for SPort:=80/24
NLRI in Hex: 068150/24
Versions:
  Process           bRIB/RIB  SendTblVer
  Speaker                 16          16
    Flags: 0x04001001+0x00000000;
Last Modified: Feb  5 04:00:37.373 for 00:03:29
Paths: (1 available, best #1)
  Not advertised to any peer
  Path #1: Received by speaker 0
  Flags: 0x4000000001060001, import: 0x20
  Not advertised to any peer
  1
    0.0.0.0 from 25.2.1.11 (6.6.6.6)
      Origin IGP, localpref 100, valid, external, best, group-best
      Received Path ID 0, Local Path ID 1, version 16
      Extended community: FLOWSPEC Traffic-rate:1,39269
RP/0/RP0/CPU0:Client#
```

# Show Commands

- Globally, we verify which interfaces are enable for FlowSpec

```
RP/0/RP0/CPU0:Client#show policy-map transient targets type pbr

1) Policymap: __bgpfs_default_IPv4    Type: pbr
     Targets (applied as main policy):
       HundredGigE0/1/0/0 input
       HundredGigE0/0/0/0 input
       ServiceInfra7 input
       TenGigE0/2/0/5 input
       TenGigE0/2/0/8 input
       TenGigE0/2/0/4 input
     Total targets: 6

RP/0/RP0/CPU0:Client#
```

# Show Commands

- We verify also how are reconstructed these policies

```
RP/0/RP0/CPU0:Client#show policy-map transient type pbr pmap-name
__bgpfs_default_IPv4

policy-map type pbr __bgpfs_default_IPv4
 handle:0x36000002
 table description: L3 IPv4 and IPv6
 class handle:0x7600000a  sequence 1024
   match source-port 80
  police rate 314152 bps
    conform-action transmit
    exceed-action drop
   !
 !
 class handle:0xf6000002  sequence 4294967295 (class-default)
 !
 end-policy-map
!
RP/0/RP0/CPU0:Client#
```

# Show Commands

- Globally, we verify which interfaces are enable for FlowSpec

```
RP/0/RP0/CPU0:Client#show flowspec afi-all detail

AFI: IPv4
 Flow           :SPort:=80
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :                        0/0
   Transmitted          :                        0/0
   Dropped              :                        0/0
RP/0/RP0/CPU0:Client#

RP/0/RP0/CPU0:Client#show flowspec ipv4 nlri

AFI: IPv4
 NLRI (Hex dump) :       0x068150
  Actions       :Traffic-rate: 314152 bps  (bgp.1)
RP/0/RP0/CPU0:Client#
```

# Show Commands

```
RP/0/RP0/CPU0:Client#show flowspec ipv4 internal
AFI: IPv4
 Flow            :SPort:=80
  Actions        :Traffic-rate: 314152 bps  (bgp.1)
   Client Version: 0
   Unsupported:    FALSE
   RT:
    VRF Name Cfg:   0x00
    RT Cfg:         0x00
    RT Registered:  0x00
    RT Resolved:    0x00
   Class handles:
    Handle [0]:        300000007600000a
   Class Handle Version:     1
   Sequence:                 1024
   Synced:                   TRUE
   Match Unsupported:        None
   Ref Count:                1
   Last Error:               0:No error
   Last Batch:               9
  Statistics                          (packets/bytes)
   Matched             :                    0/0
   Transmitted         :                    0/0
   Dropped             :                    0/0
RP/0/RP0/CPU0:Client#
```

# Show Commands

- On a CRS client, we check the TCAM usage on the linecard

```
RP/0/RP0/CPU0:CRS-3#show contr pse tcam summary location 0/0/CPU0

<SNIP>

TCAM Device Information for Ingress PSE, CAM bank 1:
Device size: 20M (256K array entries of 80-bits), 261122 available
Current mode of operation: Turbo
<SNIP>
Feature specific information:
<SNIP>
        Flowspec IPv4 (id 32):
                Owner client id: 20.   Limit 245760 cells
                Total 1 regions using 4 CAM cells
<SNIP>
```

# Show Commands

- On a ASR9000 client, we can also check the TCAM entries in some extend

```
RP/0/RSP0/CPU0:ASR9000#sh prm server tcam summary all PBR np0 location 0/0/CPU0

                Node: 0/0/CPU0:
----------------------------------------------------------------
TCAM summary for NP0:

  TCAM Logical Table: TCAM_LT_L2 (1)
    Partition ID: 0, priority: 2, valid entries: 1, free entries: 2047
    Partition ID: 1, priority: 2, valid entries: 0, free entries: 2048
    Partition ID: 2, priority: 1, valid entries: 0, free entries: 2048
    Partition ID: 3, priority: 1, valid entries: 0, free entries: 8192
    Partition ID: 4, priority: 0, valid entries: 1, free entries: 83967
  TCAM Logical Table: TCAM_LT_ODS2 (2), free entries: 89723, resvd 128
    ACL Common Region: 448 entries allocated. 448 entries free
    Application ID: NP_APP_ID_PBR (5)
      Total: 0 vmr_ids, 0 active entries, 0 allocated entries.
  TCAM Logical Table: TCAM_LT_ODS8 (3), free entries: 15204, resvd 127
    ACL Common Region: 448 entries allocated. 448 entries free
    Application ID: NP_APP_ID_PBR (5)
      Total: 1 vmr_ids, 2 active entries, 2 allocated entries.

RP/0/RSP0/CPU0:ASR9000#
```

# Show Commands

- On a NCS6000 client too

```
attach location 0/1/CPU0
pbtm_show -n 0 -s

NPU:0 Dev:0 Num Cblks:64 InUse:Y Num SubCblks:128 SubCblks Used:3

Idx  Idx  Sub   In   Unit    Alloc     Res   Num    Num    Use
     HW   cblk  use  size   feature    Size  Cells  Free    %
===  ===  ====  ===  ====  ==========  ====  =====  =====  ====
  0    0     0    Y  160b     ACLv4     16B  2048   1974   4%
---  ---  ----  ---  ----  ----------  ----  -----  -----  ----
  1    1     0    Y  640b     ACLv6     16B  2048   2040   1%
---  ---  ----  ---  ----  ----------  ----  -----  -----  ----
63   63     1    Y  160b               16B  2048   2044   1%
---  ---  ----  ---  ----  ----------  ----  -----  -----  ----

NPU:0 Dev:1 Num Cblks:64 InUse:Y Num SubCblks:128 SubCblks Used:2

Idx  Idx  Sub   In   Unit    Alloc     Res   Num    Num    Use
     HW   cblk  use  size   feature    Size  Cells  Free    %
===  ===  ====  ===  ====  ==========  ====  =====  =====  ====
  0  128     0    Y  160b     ACLv4     16B  2048   2046   1%
---  ---  ----  ---  ----  ----------  ----  -----  -----  ----
  1  129     0    Y  640b     ACLv6     16B  2048   2040   1%
---  ---  ----  ---  ----  ----------  ----  -----  -----  ----
```

# Show Commands

- To help TAC progress faster to identify a problem

```
On the Controller:
- show run class-map
- show class-map

On the Client:
- debug flowspec all
- show flowspec trace manager event error
- show flowspec trace client event error
- show flowspec client internal
- show logging | inc FLOW
- show flowspec vrf all afi-all summary internal
- show flowspec vrf all afi-all internal
- show tech flowspec
```

# Show Commands

- To measure the traffic matched, no SNMP but CLI and Netconf/XML.

```
RP/0/RP0/CPU0:Client#show flowspec ipv4 detail

AFI: IPv4
 Flow           :Dest:25.1.104.0/24
  Actions       :Traffic-rate: 100000 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :          21946725652/13958117514672
   Transmitted          :              236878/150654408
   Dropped              :          21946488774/13957966860264
 Flow           :Proto:=17,DPort:=53
  Actions       :Traffic-rate: 1234000000 bps  (bgp.1)
  Statistics                          (packets/bytes)
   Matched              :                 0/0
   Transmitted          :                 0/0
   Dropped              :                 0/0
RP/0/RP0/CPU0:Client#
```

Counters for each rule are available per VRF / address-family, not per interface.

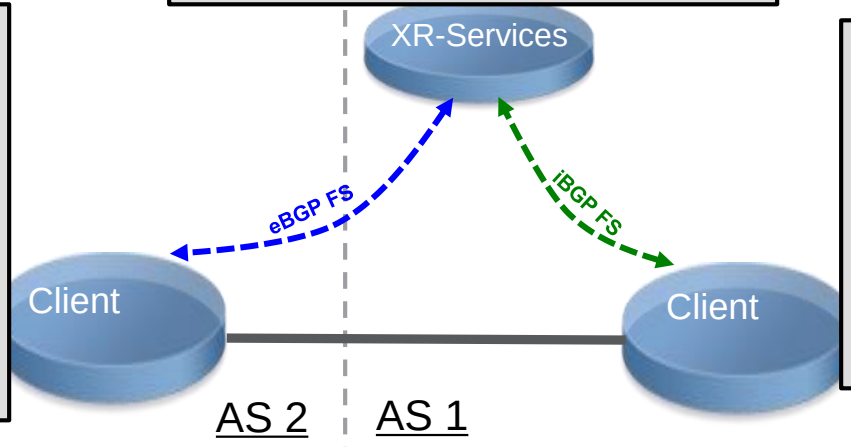
# eBGP FlowSpec

Controller

```
router bgp 1
 neighbor 25.2.1.3
  remote-as 2
  update-source GigabitEthernet0/0/0/0
  address-family ipv4 flowspec
   route-policy pass-all in
   route-policy pass-all out
   next-hop-unchanged
  !
 neighbor 25.2.1.4
  remote-as 1
  update-source GigabitEthernet0/0/0/0
  address-family ipv4 flowspec
```

Client eBGP

```
router bgp 2
address-family ipv4 flowspec
 !
 neighbor 25.2.1.11
  remote-as 1
  update-source TenGigE0/2/0/8
  address-family ipv4 unicast
  !
  address-family ipv4 flowspec
   route-policy pass-all in
   route-policy pass-all out
   validation disable
  !
 !
```




Client iBGP

```
router bgp 1
address-family ipv4 flowspec
 !
neighbor 25.2.1.10
  remote-as 1
  update-source TenGigE0/2/0/6
  address-family ipv4 unicast
  !
  address-family ipv4 flowspec
  !
 !
!
```

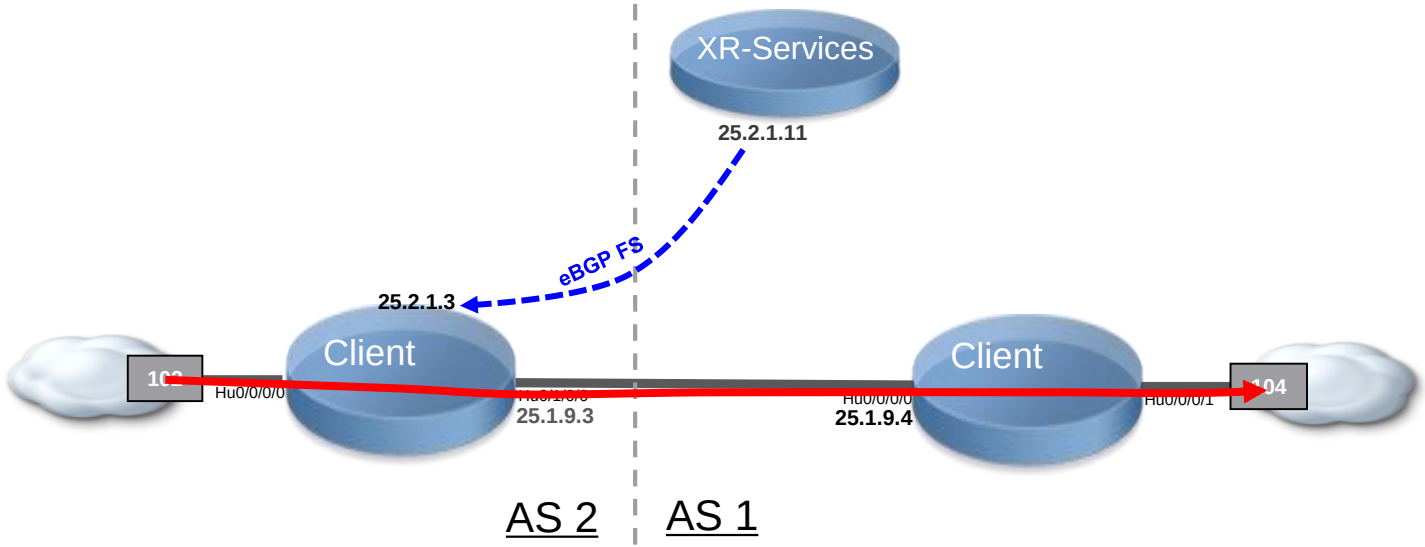
# eBGP FlowSpec: Validate Disable

Without the “Validate disable”, a check on AS Path is done and the route is not accepted.

```
RP/0/RP0/CPU0:Client#sh bgp ipv4 flowspec Dest:25.1.104.1/32,Proto:=17,Length:>=500&<=1550/128 detail

BGP routing table entry for Dest:25.1.104.1/32,Proto:=17,Length:>=500&<=1550/128
NLRI in Hex: 0120190168010381110a1301f4d5060e/128
Versions:
  Process           bRIB/RIB  SendTblVer
  Speaker                 8           8
    Flags: 0x04000001+0x00000200;
Last Modified: Feb  8 10:56:01.372 for 00:01:42
Paths: (1 available, no best path)
  Not advertised to any peer
  Path #1: Received by speaker 0
  Flags: 0x4000080000020001, import: 0x20
  Not advertised to any peer
  1
    0.0.0.0 from 25.2.1.11 (6.6.6.6)
      Origin IGP, localpref 100, valid, external, invalid flowspec-path
      Received Path ID 0, Local Path ID 0, version 0
      Extended community: FLOWSPEC Traffic-rate:1,12500
RP/0/RP0/CPU0:Client#
```

# eBGP FlowSpec: Next-Hop Unchanged

- Without the “NH unchanged” configuration, the NH action will not work on eBGP
- NH will be, by default, positioned as the peer address



# eBGP FlowSpec: Next-Hop Unchanged

## Controller

```
policy-map type pbr TEST
 class type traffic MATCHING-RULE1
  redirect nexthop 25.3.9.4
 !
 class type traffic class-default
 !
 end-policy-map
```

## eBGP Client

```
RP/0/RP0/CPU0:Client#sh bgp ipv4 flowspec
<SNIP>
 Network            Next Hop            Metric LocPrf Weight Path
*> Dest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550/128
                     25.2.1.11                                0 1 i

Processed 1 prefixes, 1 paths
RP/0/RP0/CPU0:Client#
```

We configure next-hop-unchanged on the controller:

```
RP/0/0/CPU0:Ctrl#conf
Tue Feb 10 03:55:22.423 UTC
RP/0/0/CPU0:Ctrl(config)#router bgp 1
RP/0/0/CPU0:Ctrl(config-bgp)#neighbor-group ebgp-
flowspec
RP/0/0/CPU0:Ctrl(config-bgp-nbrgrp)#address-family
ipv4 flowspec
RP/0/0/CPU0:Ctrl(config-bgp-nbrgrp-af)#next-hop-
unchanged
RP/0/0/CPU0:Ctrl(config-bgp-nbrgrp-af)#commit
RP/0/0/CPU0:Ctrl(config-bgp-nbrgrp-af)#
```

```
RP/0/RP0/CPU0:Client#sh bgp ipv4 flowspec
<SNIP>
 Network            Next Hop            Metric LocPrf Weight Path
*> Dest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550/128
                     25.3.9.4                                 0 1 i
RP/0/RP0/CPU0:Client#sh flows ipv4 det
AFI: IPv4
 Flow           :Dest:25.1.102.1/32,Proto:=17,Length:>=500&<=1550
  Actions       :Nexthop: 25.3.9.4   (bgp.1)
  Statistics                         (packets/bytes)
   Matched             :             10964755/15306797980
   Dropped             :                    0/0
RP/0/RP0/CPU0:Client#
```

# IOS XR Implementation

## Application on Interface

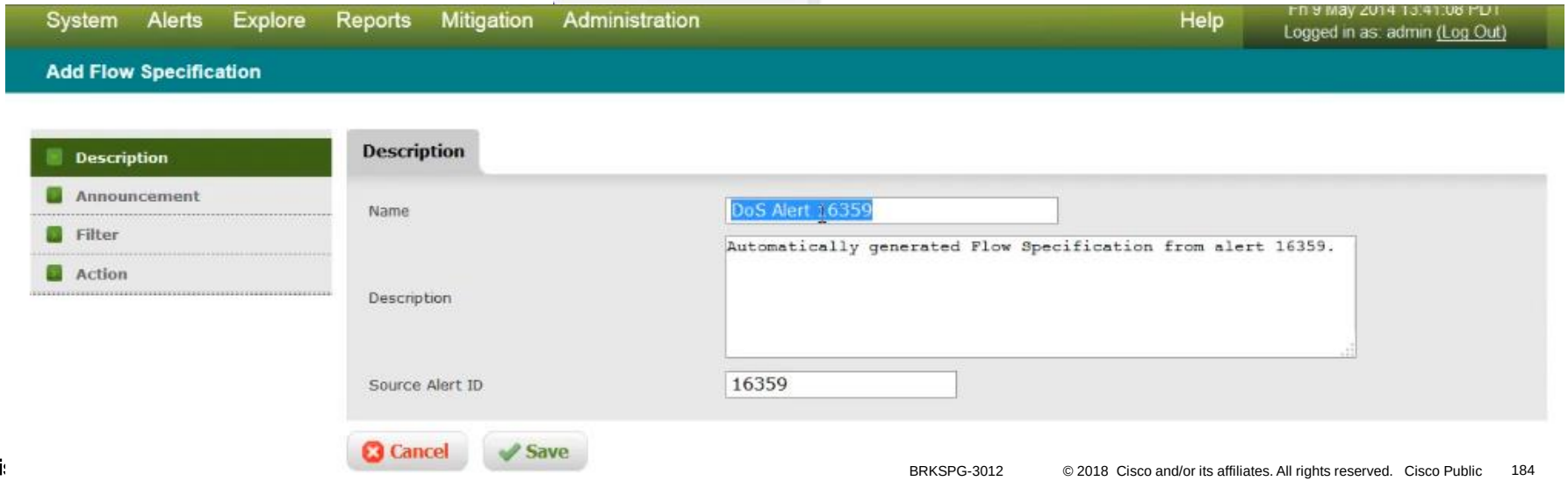
- Uses the PBR infrastructure with similar performance penalty than other PBR features like ABF. Performance cost will vary depending upon the action
  - DSCP marking will be least expensive
  - redirect action pointing to recursive TE tunnel path being most expensive
- Can coexist with other features like QoS or ACL
- Interface can be in the Global Routing Table or on a VRF (L3VPN or VRF-Lite)

# Back-Up Slides
# 3rd Party Controller

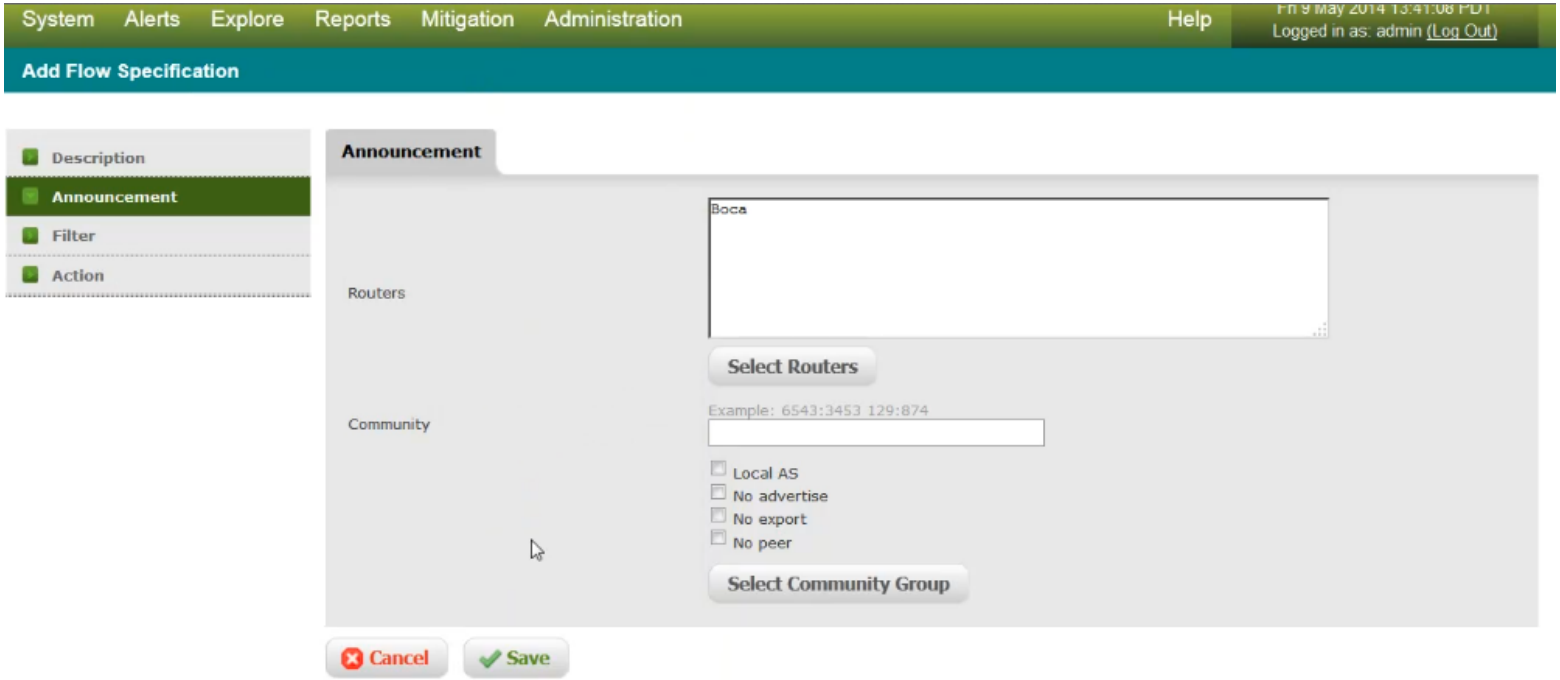
# BGP FlowSpec with 3rd Party Apps

- BGP FlowSpec is based on IETF standard
- It can interoperate with non-Cisco devices compliant to the standards
- Following list in offering a few controllers examples and is non-exhaustive
  - Arbor SP
  - ExaBGP
  - YABGP
  - Open Day Light

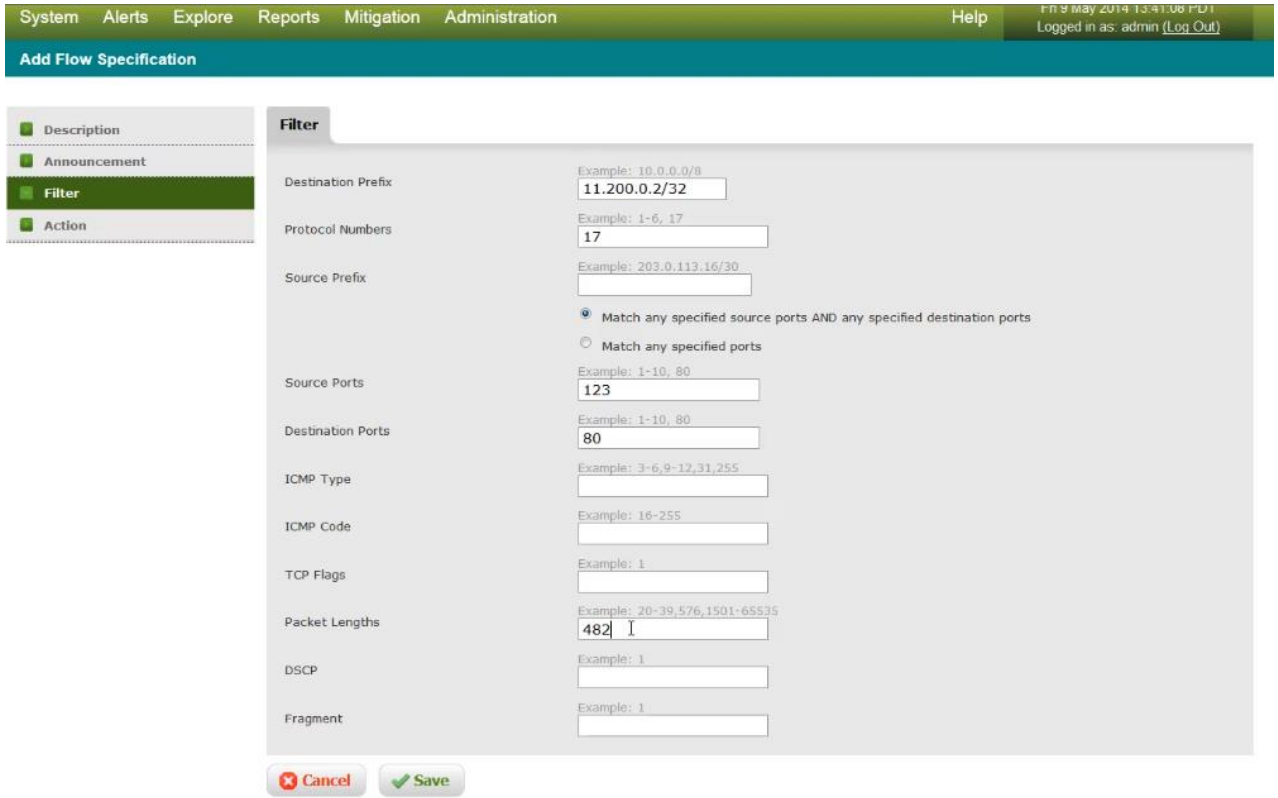
# Using Arbor SP

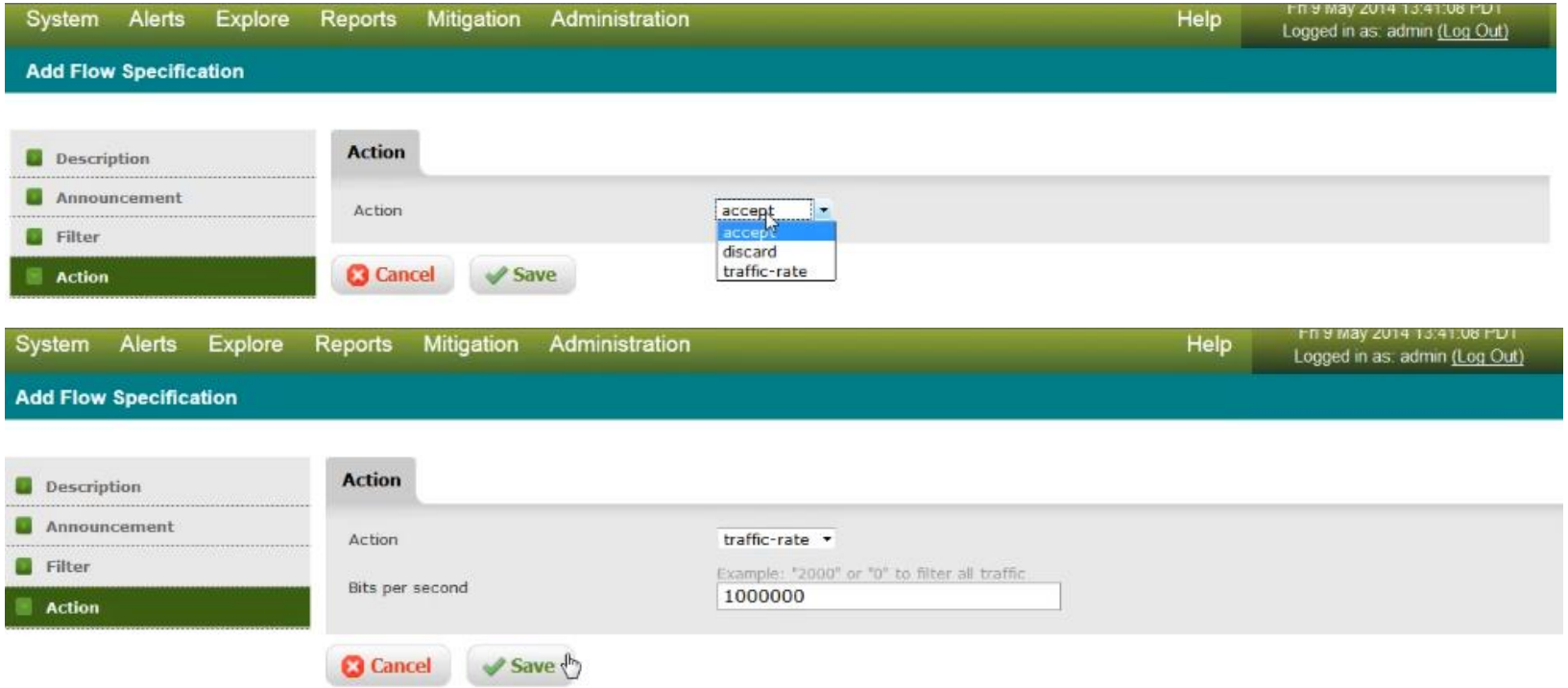
# Using Arbor SP



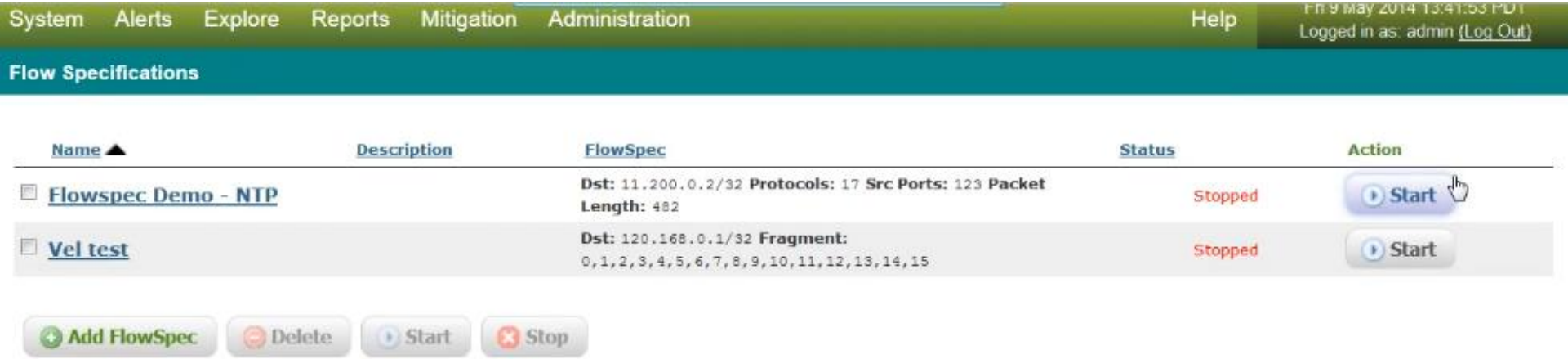
# Using Arbor SP



# Using Arbor SP



# Using Arbor SP



For Your Reference

# Using ExaBGP

```
neighbor 10.0.0.1 {
        description "xrv 5.2.0";
        router-id 192.168.2.26;
        local-address 192.168.2.26;
        local-as 65000;
        peer-as 65000;
        graceful-restart 5;

        flow {
         route name-of-the-route {
                match { ...
                  <<<description>>>
                }
                then { ...
                  <<<action>>>
                }
        }
```

```
flow {
                route name-of-the-route {
                        match {
                                source 10.0.0.1/32;
                                destination 192.168.0.1/32;
                                port =80 =8080;
                                destination-port >8080&<8088 =3128;
                                source-port >1024;
                         protocol [ tcp udp ];
                                packet-length >200&<300 >400&<500;
                                #fragment not-a-fragment;
                                fragment [ first-fragment last-fragment ];
                                icmp-type [ unreachable echo-request echo-reply ];
                                icmp-code [ host-unreachable network-unreachable ];
                                tcp-flags [ urgent rst ];
                                dscp [ 10 20 ];

                        }
                        then {
                                #rate-limit 9600;
                                #discard;
                                redirect 65500:12345;
                                #redirect 1.2.3.4:5678;
                                community [30740:0 30740:30740];
                                #extended-community [ origin:2345:6.7.8.9 origin:2.3.4.5:6789 ];
                        }
                }
        }
```

# Using Open Day Light

```
<flowspec-route xmlns="urn:opendaylight:params:xml:ns:yang:bgp-flowspec">
      <route-key>flow1</route-key>
      <flowspec>
        <destination-prefix>192.168.0.1/32</destination-prefix>
      </flowspec>
      <flowspec>
        <source-prefix>10.0.0.1/32</source-prefix>
      </flowspec>
      <flowspec>
        <protocol-ips>
          <op>equals end-of-list</op>
          <value>6</value>
        </protocol-ips>
      </flowspec>
      <flowspec>
        <ports>
          <op>equals end-of-list</op>
          <value>80</value>
        </ports>
      </flowspec>
      <flowspec>
        <destination-ports>
          <op>greater-than</op>
          <value>8080</value>
        </destination-ports>
        <destination-ports>
          <op>and-bit less-than end-of-list</op>
          <value>8088</value>
        </destination-ports>
      </flowspec>
```

**For Your Reference**

```
      <flowspec>
        <source-ports>
          <op>greater-than end-of-list</op>
          <value>1024</value>
        </source-ports>
      </flowspec>
      <flowspec>
        <types>
          <op>equals end-of-list</op>
          <value>0</value>
        </types>
      </flowspec>
      <flowspec>
        <codes>
          <op>equals end-of-list</op>
          <value>0</value>
        </codes>
      </flowspec>
      <flowspec>
        <tcp-flags>
          <op>match end-of-list</op>
          <value>32</value>
        </tcp-flags>
      </flowspec>
      <flowspec>
        <packet-lengths>
          <op>greater-than</op>
          <value>400</value>
        </packet-lengths>
        <packet-lengths>
          <op>and-bit less-than end-of-list</op>
           <value>500</value>
        </packet-lengths>
      </flowspec>
```


You're
Cisco live!