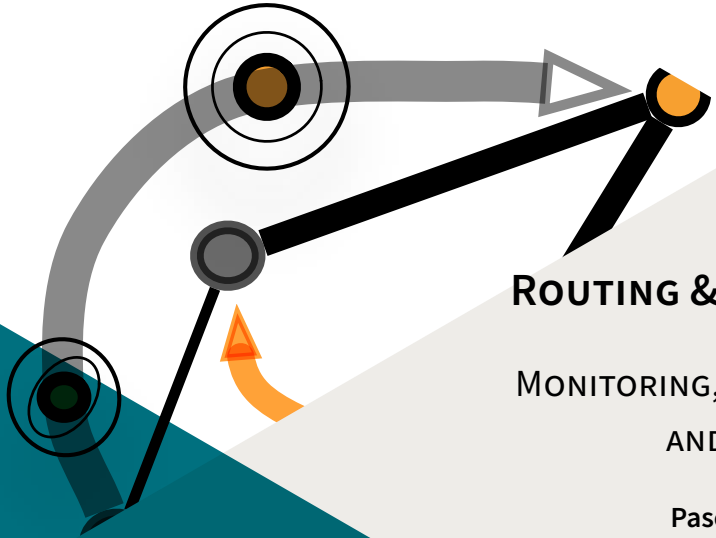
Université de Strasbourg



Routing & Measurements

Monitoring, troubleshooting and... Testing!

Pascal Mérindol

26/6/2023

A brief presentation: who is this guy?

My main Research Contributions

Université de Strasbourg

Two Main Research Topics to Improve the Internet

Both about Routing but from two distinct perspectives

1. **Routing Algorithms & Protocols**: **Compute and Deploy Valuable IP Routes**
   - Fast and correct convergence for several change models: link and node failures, maintenances or BGP hot potato
   - Multi-metric and multi-paths for traffic engineering
2. **IP Measurements** in Transit Networks: **Topology Discovery & Large Scale Monitoring**
   - Reveal Hidden MPLS Tunnels and Forwarding Detours
   - Troubleshoot ISP (Internet Service Provider) Networks

Introduction & Agenda

Measurements in the wild

Active probing vs. Passive parsing

How to capture Internet maps and troubleshoot networks?

- *Looking glasses* = BGP routing : control plane!
- `traceroute` = IP forwarding : data plane!

Control Plane > 130.79.0.0/16 (STBG prefix)

```
130.79.0.0/16       *[BGP/170] 5d 15:35:20, localpref 300, from 90.147.84.8
                      AS path: 20965 2200 2259 I, validation-state: unverified
                    > to 90.147.80.170 via ae0.1, Push 473220
                      to 90.147.82.114 via et-2/0/0.1, Push 101310
```

Data Plane (beginning): traceroute Nap@Garr > 130.79.207.98

```
1  rx1-na1-rx2-na1.na1.garr.net (90.147.80.170)  19.668 ms  19.746
   MPLS Label=473220 CoS=0 TTL=1 S=1
2  90.147.82.94 (90.147.82.94)  4.459 ms  4.335
    MPLS Label=388429 CoS=0 TTL=1 S=1
```

Measurements in the wild

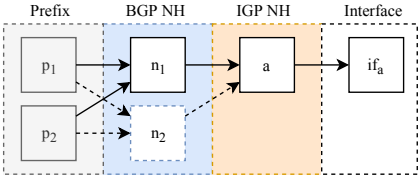
Data Plane towards STBG' Mini-Internet (remainder)

```
3  rs1-rm02-re1-mi02.mi02.garr.net (185.191.181.69)  19.403 ms  19.479
4  garr.rt1.mil2.it.geant.net (62.40.125.180)  19.802 ms  19.575
5  ae2.mx1.gen.ch.geant.net (62.40.98.232)  27.367 ms  26.806
6  renater-gw.mx1.gen.ch.geant.net (62.40.124.62)  29.338 ms  29.209
7  te-0-0-0-11-ren-nr-dijon-rtr-091.noc.renater.fr (193.55.204.8)  34.431 ms
te0-1-0-0-ren-nr-dijon-rtr-091.noc.renater.fr (193.51.177.73)  34.092 ms
te-0-0-0-11-ren-nr-dijon-rtr-091.noc.renater.fr (193.55.204.8)  75.960 ms
    MPLS Label=24146 CoS=0 TTL=1 S=1
...
9  te0-1-0-9-strasbourg-rtr-091.noc.renater.fr (193.55.203.175)  34.408 ms
te-0-0-0-9-ren-nr-strasbourg-rtr-091.noc.renater.fr (193.55.203.171)  41.147 ms
te0-1-0-10-strasbourg-rtr-091.noc.renater.fr (193.55.203.177)  34.755 ms
10  osiris-primaire-vl2500-te1-3-strasbourg-rtr-021.noc.renater.fr (193.51.183.129)
...
```

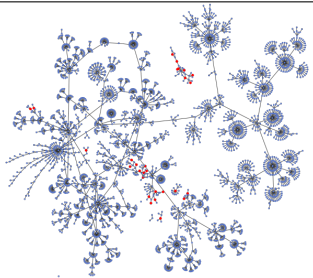
Context & Recap

Agenda

1. Context, Motivation & Background
2. Measuring and Improving Routing Systems
3. **Testing and Checking Routing Systems**



A hierarchical forwarding model



A sample of the Internet

Internet: the Worldwide Distributed System

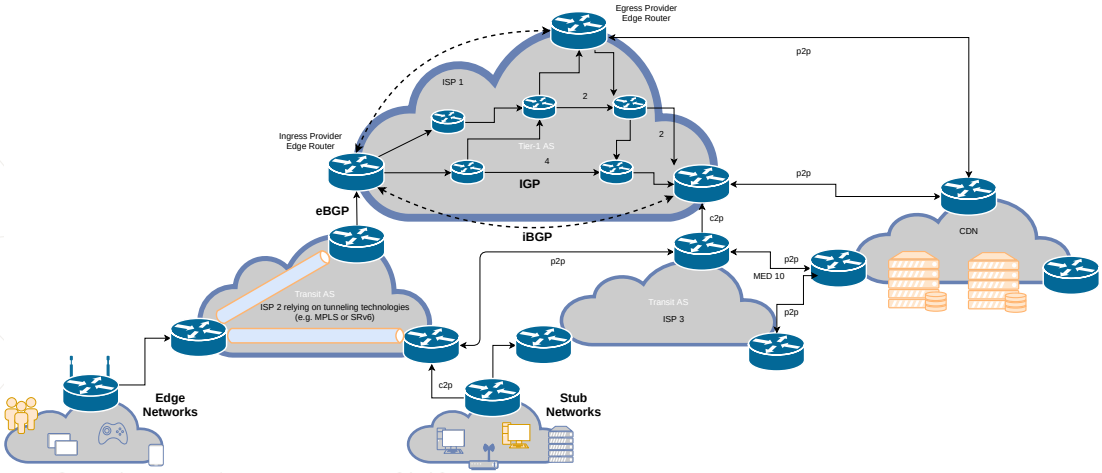
- **Large scale**: soon $100,000$ Domains (including $\approx [100...10,000]$ routers), $1M$ IP prefixes and much more end-devices than human beings!
- **Complex**: numerous heterogeneous hardware and software components

Internet: the Worldwide Distributed System

- **Large scale**: soon $100,000$ Domains (including $\approx [100...10,000]$ routers), $1M$ IP prefixes and much more end-devices than human beings!
- **Complex**: numerous heterogeneous hardware and software components
- **Innovation is difficult and challenging**:
  - The universal common technology for connecting all devices lies in the **IP data-plane**
  - The **IP control-plane (i.e. the routing)** provides means to install and manage forwarding routes, e.g. with MPLS or Segment Routing (SR) w.r.t. the type of service

The Big Picture

Complex ecosystems



Two Scales and Several Technologies

Two scales of routing operations:

- inter-domain or AS-wide level: **BGP**
- intra-domain or router level within an AS: **IGP**

Two Scales and Several Technologies

Two scales of routing operations:

- inter-domain or AS-wide level: **BGP**
- intra-domain or router level within an AS: **IGP**

Several technologies and paradigms:

- **Hop-by-hop** packet forwarding (for best-effort traffic), or **loose source routing**, e.g. with **SR or MPLS tunnels**
- With packet encapsulation, one can enable **Traffic Engineering (TE)** for premium flows, or deploy **Load Balancing (LB) and Fast-ReRoute** (FRR) for all services.
  A single deviation from the best-effort usual IP forwarding scheme is often enough

Background: Internet Routing Protocols

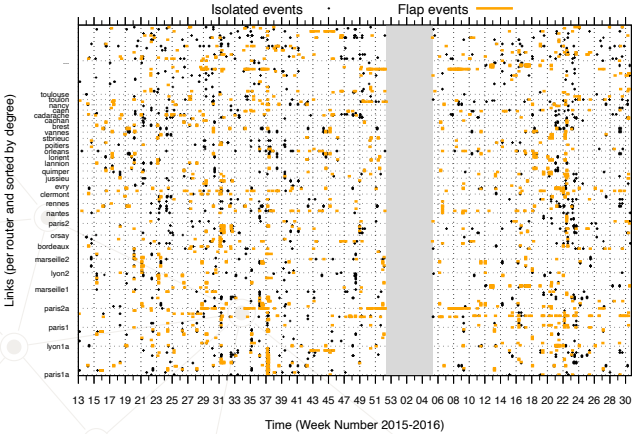
Two respective paradigms for inter- and intra-domain routing:

- with BGP, preferences are local and selfish: the path vector protocol may diverge
  - sufficient conditions for convergence a priori met in practice (*valley free* routes and usual economical incentives)
- within IGP link-state routing protocols (e.g. OSPF, IS-IS), the objective is globally consistent, typically a *minimum function* applied on an *additive metric*: $(min, +)$
  - convergence is granted (thanks to the *isotonicity* & *monotony* of routing operations), but **anomalies, e.g. forwarding loops and path sub-optimality,** still occur during transient periods of change

Routing Changes: Illustration in RENATER

Université de Strasbourg

The **DCART** project deployed in the French Educational & Research Network

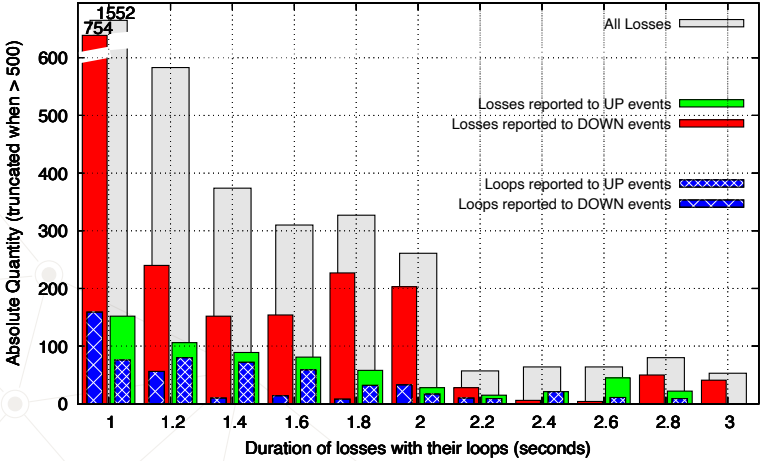


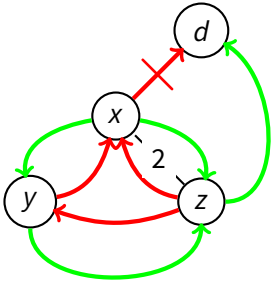
RENATER

GÉANT
Networks • Services • People

Topological changes are common and links are often **flapping**

Routing Changes: Illustration in RENATER





A not so simple ECMP forwarding loop

Forwarding loops do occur, in particular when routers reboot!

Routing Divergence: a brief Recall

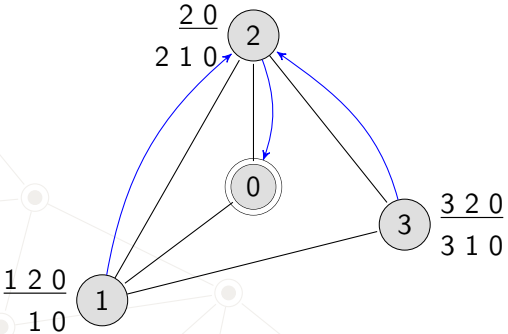
Standard BGP route ranking process

Step	Criterion	
1	Prefer highest `local-pref`	(economical relationships)
2	Prefer shortest `as-path`	(global optimality)
3	Prefer lowest `origin`	
4	Prefer lowest `med`	(cold potato routing)
5	Prefer routes learned over `ebgp`	
6	Prefer lowest `igp` cost	(hot potato routing)
7	Prefer lowest `router-id`	(arbitrary tie-break)

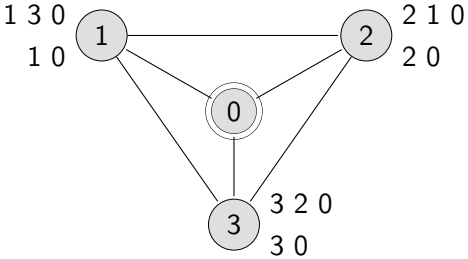
The first criterion is enough to make BGP diverge, as well as applying filtering, as both isotonicity and monotony are lost...

BGP Divergence: a Practical Illustration

A Good Gadget (GG):



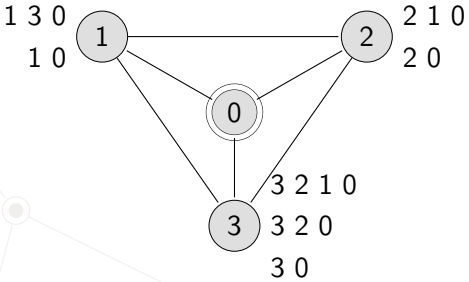
A Bad Gadget (BG):

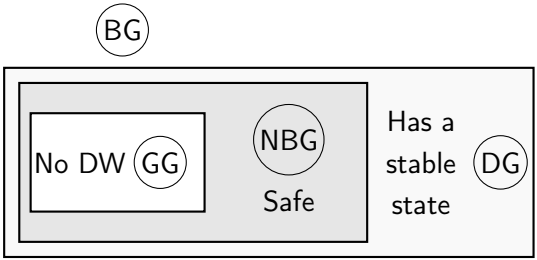


Do we have a stable state here?

BGP Divergence: a brief Summary

A Not so Bad Gadget (NBG):





The small Universe of Gadgets:

- What means No DW?
- What can be DG?

Let's come back to Measurements!

Internet Mapping and Troubleshooting

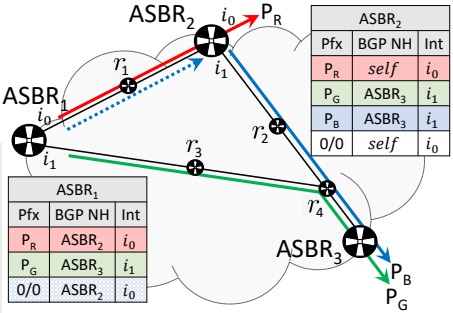
Several Technical Challenges

- Topology Discovery
  - Reveal hidden MPLS tunnels (TNT)
  - Internet maps and L2 network analysis (MERLIN)
- Network Monitoring and Analysis
  - Detect forwarding detours and BGP lies
  - Finely monitor intra-domain traffic (DCART)

Originality with respect to related works

Fine-grained passive and active network correlations using several sources of data: novel probing tools and methods exploiting error replies

The Problem of Forwarding Detours



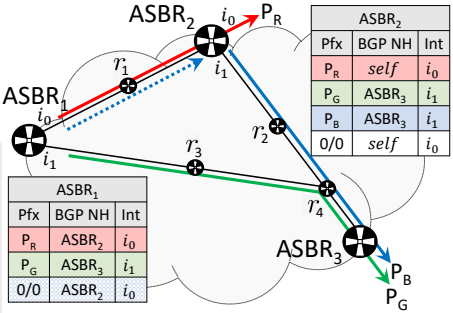
ASBR$_2$		
Pfx	BGP NH	Int
P$_R$	*self*	i_0
P$_G$	ASBR$_3$	i_1
P$_B$	ASBR$_3$	i_1
0/0	*self*	i_0

ASBR$_1$		
Pfx	BGP NH	Int
P$_R$	ASBR$_2$	i_0
P$_G$	ASBR$_3$	i_1
0/0	ASBR$_2$	i_0

Partial FIBs and Detours:

- ASBR$_1$ uses ASBR$_2$ as a default next-hop...
- ...but, instead of sending the traffic externally, ASBR$_2$ then considers ASBR$_3$ for prefix P$_B$
- the **default route** of ASBR$_1$ visible because of the Forwarding Detour (FD) to reach ASBR$_3$
- for prefixes P$_G$ and P$_B$, traffic is forwarded differently $\Rightarrow$ multi-path forwarding pattern

The Problem of Forwarding Detours



ASBR$_2$		
Pfx	BGP NH	Int
P$_R$	*self*	i_0
P$_G$	ASBR$_3$	i_1
P$_B$	ASBR$_3$	i_1
0/0	*self*	i_0

ASBR$_1$		
Pfx	BGP NH	Int
P$_R$	ASBR$_2$	i_0
P$_G$	ASBR$_3$	i_1
0/0	ASBR$_2$	i_0

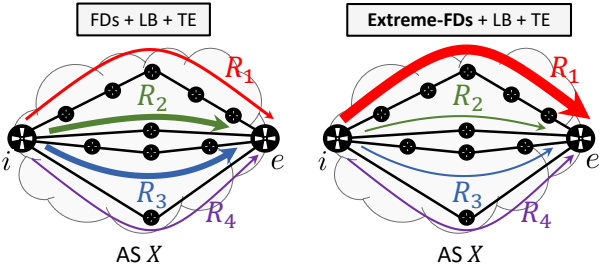
Partial FIBs and Detours:

- ASBR$_1$ uses ASBR$_2$ as a default next-hop...
- ...but, instead of sending the traffic externally, ASBR$_2$ then considers ASBR$_3$ for prefix P$_B$
- the **default route** of ASBR$_1$ visible because of the Forwarding Detour (FD) to reach ASBR$_3$
- for prefixes P$_G$ and P$_B$, traffic is forwarded differently $\Rightarrow$ multi-path forwarding pattern

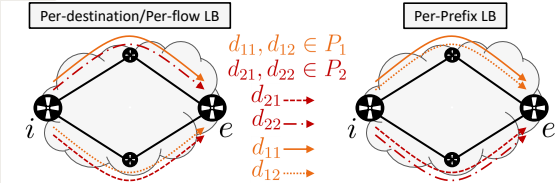
Detect and quantify FD by distinguishing them from LB and TE

Partial FIB and default routes result in FD, but several Load Balancing flavors (ECMP variants) and Traffic Engineering practices may result in the same effects at first glance

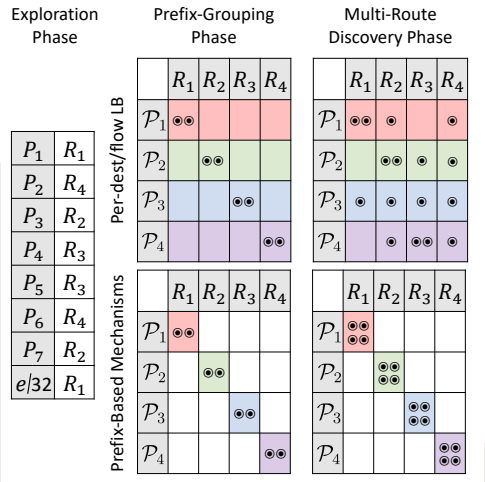
Load Balancing, TE or Forwarding Detours?



Our tool is designed to discrimate extreme deflections from usual LB/TE practices



Our Methodology in a Nutshell



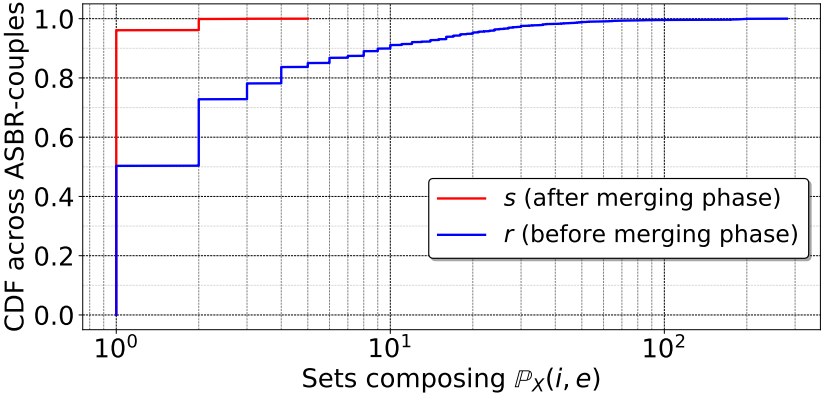
Several phases:

1. **Exploration phase**: sample blindly the IP space
2. **Prefix grouping phase**: group prefix sharing the same route
3. **Multi-route discovery phase**: look for route variation inside each set of prefixes
4. **The verdict**: find a lonely direct trace towards the egress

Routes/prefixes Partitions and merging

Some Results

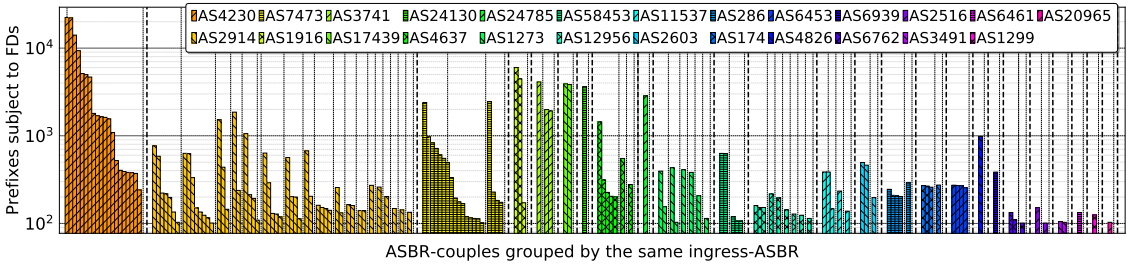
About Merging



LB is frequent and mostly occur per destination/flow: the rest looks to be extreme FD!

Some Results

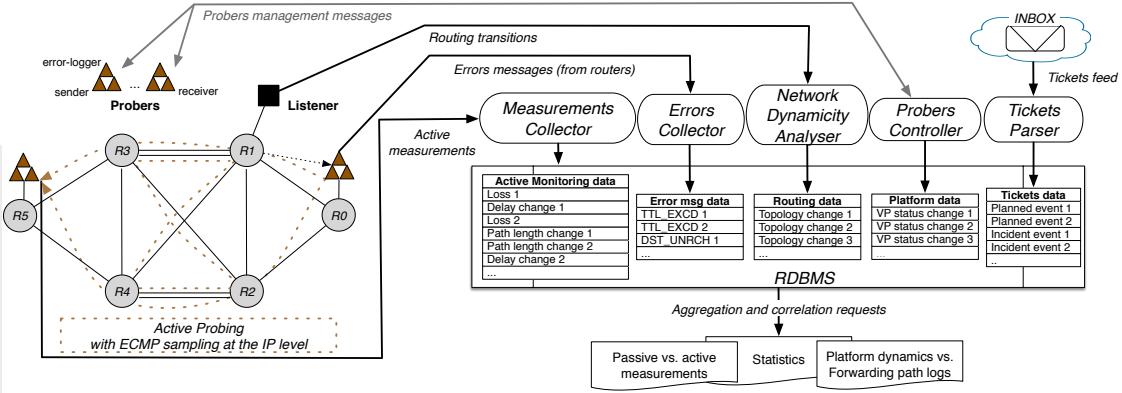
A closer look at FD



FD seems to be heterogeneous even inside a given AS but in practice we observe only extreme ones (no doubt with per-prefix or TE practices)

Let’s now move to Monitoring with DCART!

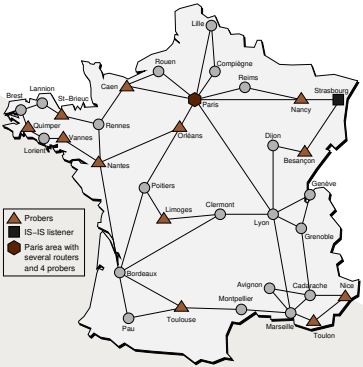
Monitoring an IGP network with DCART



Multiple sources of data to be correlated and interpreted!

RENATER: THE FRENCH NREN

OUR NETWORK UNDER SCRUTINY



Our main goals are about studying:

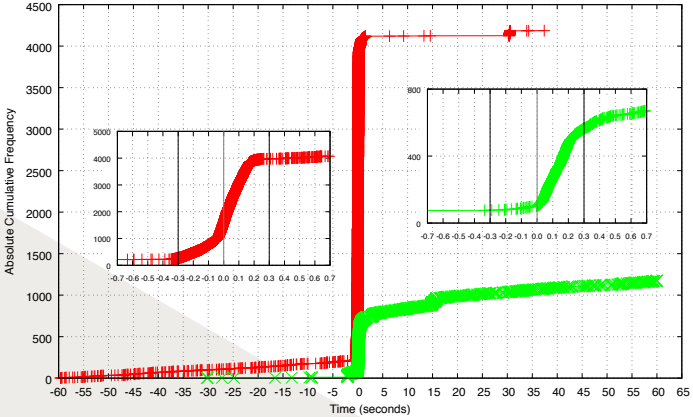
- the origin of losses
- the nature of outages/maintenances
- the impact of routing changes on losses
- forwarding loops
- the accuracy of the ticketing system

Again, the main feature of DCART lies in its ability to correlate distinct events

IS-IS with MPLS LDP generalized in the backbone

About Correlations (for Isolated events)

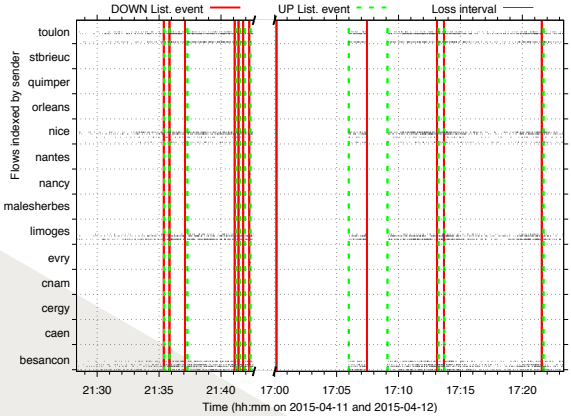
Time-based but not only...



An interval of $\pm$ 300ms looks enough with `NTP`

About Flaps

Numerous but what else?

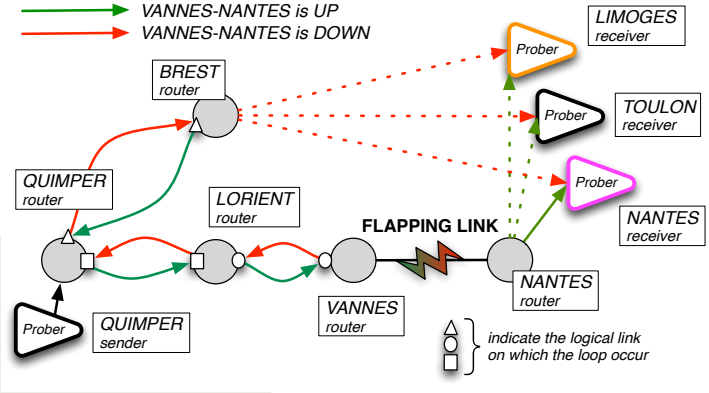


Faulty links: most losses occur when they are back...

Why not disabling them? The fail-stop model does not suffice!

About Forwarding Loops

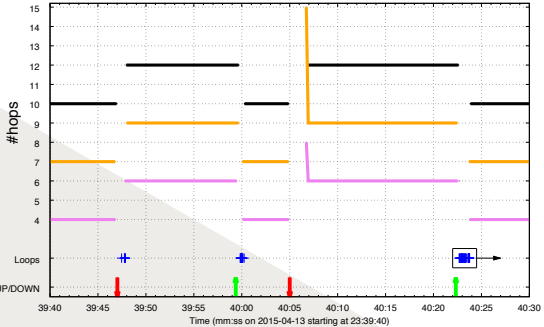
A zoom on some of them!

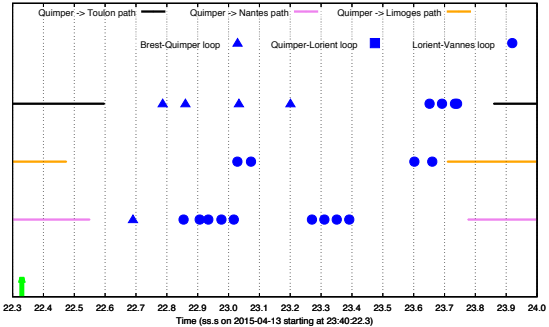


Our setup to reveal such data-plane transient anomalies

We rely on `Time-Exceeded` messages (triggered with carefully designed probes)!

Cascading Forwarding Loops





Losses after up events are also related to forwarding loops!

Let's now look at emulation tools!

Understand Protocol Implementations...

...and explore new ones?

Considering routing emulation platforms is benefitial

Understanding tricky configurations and experience new ones:

- reproduce patterns observed in the wild
- favor innovation
- offer flexible pedagogical tools

Two examples to deal with the first item: `GNS3` and `ContainerLab`

$\Longrightarrow$ Closed but **Handle Comprehensive Production OS**

But let's now look at two distinct platforms developed at ETHZ!

$\Longrightarrow$ **Open, Scalable & Flexible** but not Complete

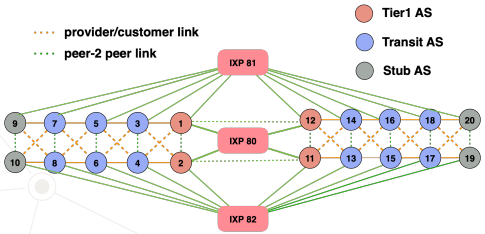
Handling the control plane..

...up to the data plane!

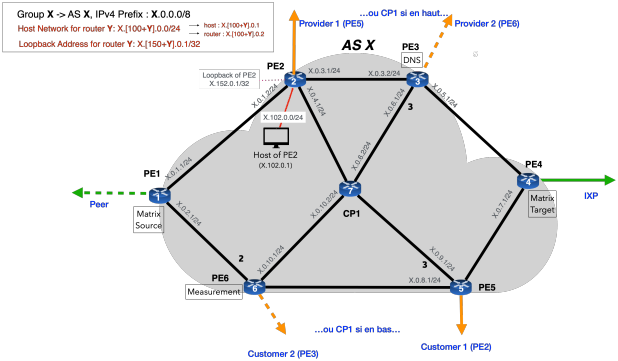
1. `Mini-Internet`: Configure and/or extend the control plane!
   $\Longrightarrow$ FRR + OVS + Docker, etc.
   `https://github.com/nsg-ethz/mini_internet_project`
2. `P4-Utils`: Control and data-plane are fully open...i.e., empty?!
   $\Longrightarrow$ MiniNet + BMV2 + Control $\leftrightarrow$ Data Langages (e.g., with Python), etc.
   `https://github.com/nsg-ethz/p4-utils`

Our Lab with Mini-Internet

A good mix of standard IP technology



eBGP



OSPF + MPLS + iBGP

A webserver to ease the analysis

Matrix & Looking Glass

routing project [•] matrix [ ] looking glass [ ] connections [ ] krill

connectivity matrix

updates every minute, last updated on 2023-06-15 at 20:58.

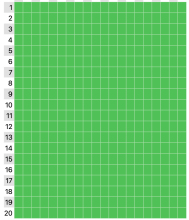
This connectivity matrix indicates the networks that each group

- can reach with a valid AS-level path (🟩);
- can reach with an invalid AS-level path (🟨);
- cannot reach (🟥).

We determine reachability by sending periodic pings between all networks, if the ping succeeds, we consider the AS reachable. In addition to the ping, we compare the BGP looking glass outputs with the project topology to determine whether the path between two ASes is valid, i.e. if it does not violate any policies.

Note that the period for pings between two ASes can be higher than the matrix update frequency, and it may take a few matrix updates until a change for a particular pair of ASes is visible.

100% 0% 0%



routing project [ ] matrix [•] looking glass [ ] connections [ ] krill

looking glass

Looking Glass servers (LG servers) are servers on the Internet running one of a variety of publicly available Looking Glass software implementations. They are commonly deployed by autonomous systems (AS) to offer access to their routing infrastructure in order to facilitate debugging network issues. A Looking Glass server is accessed remotely for the purpose of viewing routing information. Essentially, the server acts as a limited, read-only portal to routers of whatever organization is running the LG server.

Looking Glass Server
wikipedia.org

AS 1 Router PE2

```
2023-06-15T20:59:40
BGP table version is 45, local router ID is 1.152.0.1, vrf id 0
Default local pref 100, local AS 1
Status codes:  s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes:  i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

   Network          Next Hop            Metric LocPrf Weight Path
* i1.0.0.0/0       1.151.0.1                0    100      0 i
* i                1.151.0.1                0    100      0 i
*>                 0.0.0.0                  0         32768 i
*>i2.0.0.0/0       1.151.0.1                0     50      0 2 i
*>i3.0.0.0/0       1.151.0.1                0    100      0 3 i
*>i4.0.0.0/0       1.151.0.1                0    100      0 4 i
*>i5.0.0.0/0       1.151.0.1                     100      0 4 5 i
*>i6.0.0.0/0       1.151.0.1                     100      0 3 6 i
*>i7.0.0.0/0       1.151.0.1                     100      0 3 5 7 i
*>i8.0.0.0/0       1.151.0.1                     100      0 3 5 8 i
*>i9.0.0.0/0       1.151.0.1                     100      0 3 5 7 9 i
*>i10.0.0.0/0      1.151.0.1                     100      0 3 5 7 10 i
*>i11.0.0.0/0      1.151.0.1                0     50      0 11 i
* i 12.0.0.0/0     1.151.0.1                0     50      0 12 i
*>                 100.81.0.12              0     50      0 12 i
*> 13.0.0.0/0      100.81.0.12                    50      0 12 13 i
* i                1.151.0.1                      50      0 12 13 i
*> 14.0.0.0/0      100.81.0.14              0     50      0 14 i
*> 15.0.0.0/0      100.81.0.14                    50      0 14 15 i
*> 16.0.0.0/0      100.81.0.16              0     50      0 16 i
*> 17.0.0.0/0      100.81.0.16                    50      0 16 17 i
*> 18.0.0.0/0      100.81.0.18              0     50      0 18 i
*> 19.0.0.0/0      100.81.0.18                    50      0 18 19 i
*> 20.0.0.0/0      100.81.0.20              0     50      0 20 i

Displayed  20 routes and 24 total paths
```

policy analysis

updates every 5 minutes, last updated on 2023-06-15 at 20:59.

The BGP advertisements of **AS 1** do not seem to violate any policies.

Exercices: a fragile setup!

Three questions about tricky configurations

Before beginning, look around at the configurations of your AS using commands like `./goto.sh CP1 router` after your `ssh -p <2000+ASN>` connection.

$\Longrightarrow$ You will find several tutorials online (e.g., the ones of the ETHZ or Unistra).

$\Longrightarrow$ Usual probing & capturing tools are your best companions for troubleshooting: ...`ping`, `nping` (on hosts or container), `traceroute`, `tcpdump`...

Here are the three challenges of today:

1. change the IGP weight of PE6 $\rightarrow$ PE1 to 2 and try to understand what happen!
2. install a static route on PE5 via PE1 to a destination prefix directly reachable via your IXP (through PE4): what is the outcome and why there is no data-plane loops?
3. finally, look around with other groups, and try to generate forwarding oscillations, that is control-plane loops!

The tutorial before starting to play!

1. Check my website: `https://dpt-info.di.unistra.fr/~merindol`
2. Download the slides: `https://clarinet.icube.unistra.fr/~merindol/download/emul.pdf`
3. Download the VPN config: `https://clarinet.icube.unistra.fr/~merindol/download/pfSense-UDP4-1194-tmaSchool-config.ovpn`
4. Look at the board for the credentials!
5. Check the Mini-Internet website: `http://192.168.0.19:1980`
6. Organize the group and elect a leader that will send me an email: `merindol@unistra.fr`
7. Wait for my answer with `ssh` credentials and details!

Don't panic!

There is a backup plan...

With the very same VPN, you can access another Mini-Internet:

`http://mai-mini-internet.u-strasbg.fr`

1. But first add a static route (via the VPN - adapt the command to your system), e.g.:
   `sudo route -n add 130.79.207.98/32 192.168.100.1`
2. Send me an email for `ssh` credentials: `merindol@unistra.fr`
3. Access the server: `ssh -p <2000+ASN> <gASN>@130.79.207.98`
4. Look around with the `./goto.sh` script.

Look at the numerous tutorials online for help and ask me questions!

$\Longrightarrow$ The one we use this year with our students (english version): `https://git.unistra.fr/jr.luttringer/routage-inter-mini-internet/-/wikis/home`

Homework for the Summer

Practice P4 on your own to deploy and monitor SR!?

Download the VM publicly available at the ETHZ and first practice with `P4-Learning`
Try to deploy and monitor a SR like mechanism to control and verify routes...
...but playing around with P4 constraints and opportunities: no control loops, no division, limited ressources, but **fancy match-action tables**!

P4 & SR

More in details

At the moment, we look at a way to deploy *all* SR paths for a given P4 target (e.g., Tofino) for several objectives:

- IP Fast-ReRoute (TI-LFA)
- Minimize Delays (with IGP segments)
- Delay-Constrained Least Cost Paths (non dominated solutions)
- + generalizations...

We already develop an unified control plane sub-routine (**GoFor-SR**), but we now aim to deploy a complete prototype including the data-plane with P4!
One goal is to **enable LB at the source...to let it flow** (according to the expected end-to-end delay, let TCP dictate the load balancing)!

Conclusion

Measuring routing requires playing!

Emulation platforms offer the playground

Depending on the needs, several options are possible: from pedagogical training up to production networks, going through exploring new protocols!

In this summer school lab, we consider a middle range approach with an open control-plane (FRR) and the Linux data-plane. Nevertheless, it gives a good taste on how:

- reproduce/validate routing and forwarding phenomenon observed in the wild
- check configurations in a controlled environnement
- **develop and test new ideas**...

However, such tools are not really designed for performance evaluation: look at `NS3`, `OMNET++` or real (hardware) experiments for that!

Acknowledgements

Université de Strasbourg

Here is the list of people involved in these works:

- Julian Del. Fiore (the brillant PhD student)
- Valerio Persico (a great folk)
- François Clad (the Cisco folk)
- Stefano Vissichio (another great folk)
- Cristel Pelsser (a PhD advisor)
- Jean-Jacques Pansiot (another PhD advisor)
- Pierre David (a thrid great folk)
- Antonio Pescape (the boss!)

And the three papers mentioned in this talk:

- **TNSM 2021**: *The Art of Detecting Forwarding Detours*
- **TMA 2019**: *A BGP-lying Tale: Stop Blaming the Mapping*
- **ComNet 2018**: *A Fine-Grained Multi-Source Measurement Platform Correlating Routing Transitions with Packet Losses*