# Estimating the Impact of a Hijack: Measurement Bias and How to Avoid It

**Pavlos Sermpezis**

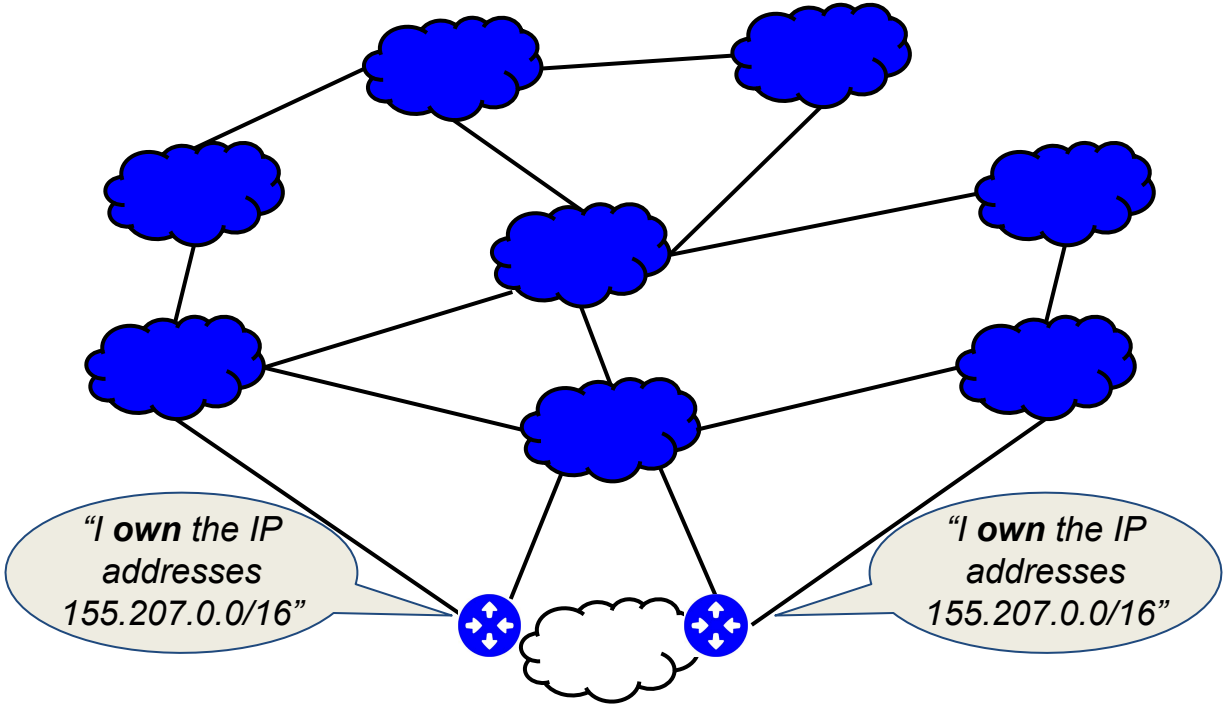
DataLab (datalab.csd.auth.gr)
Informatics Dept.
Aristotle University of Thessaloniki, Greece

* presentation based on
  - project (ongoing): ***"AI4NetMon"***, *RIPE NCC RACI project funding 2021, https://sermpezis.github.io/ai4netmon/*
  - paper: *P. Sermpezis et al.,* ***"Estimating the Impact of BGP Prefix Hijacking"****, IFIP Networking, 2021. [PDF]*

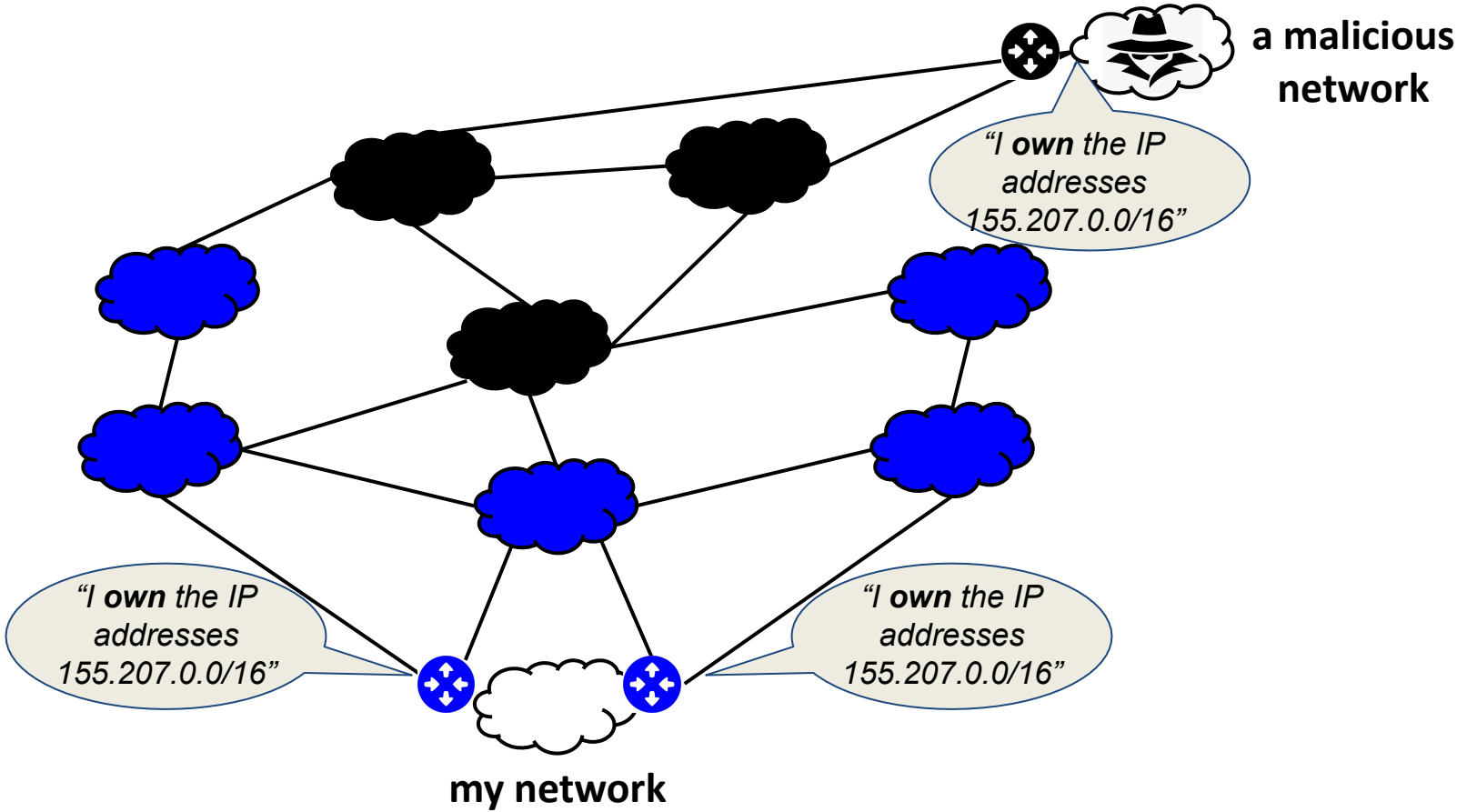
DATA&WEB SCIENCE LABORATORY

ARISTOTLE UNIVERSITY OF THESSALONIKI

# BGP prefix hijacking



# BGP prefix hijacking



# How do we defend?

- **Detection**
  - ✓ public monitoring infrastructure (RIPE RIS, RouteViews, etc.)
  - ✓ a lot of research; state-of-the-art: near real-time detection (in a few seconds) [1]

- **Mitigation** (filtering, deaggregation, outsourcing to large ISPs, blackholing, etc.)
  - ❓ different actions → different costs… *which one to choose?*
  - ❓ ok, I took an action… *was it effective? is the problem solved?*

  **→ we need to know the impact of the hijack (before/after its mitigation) !!!**

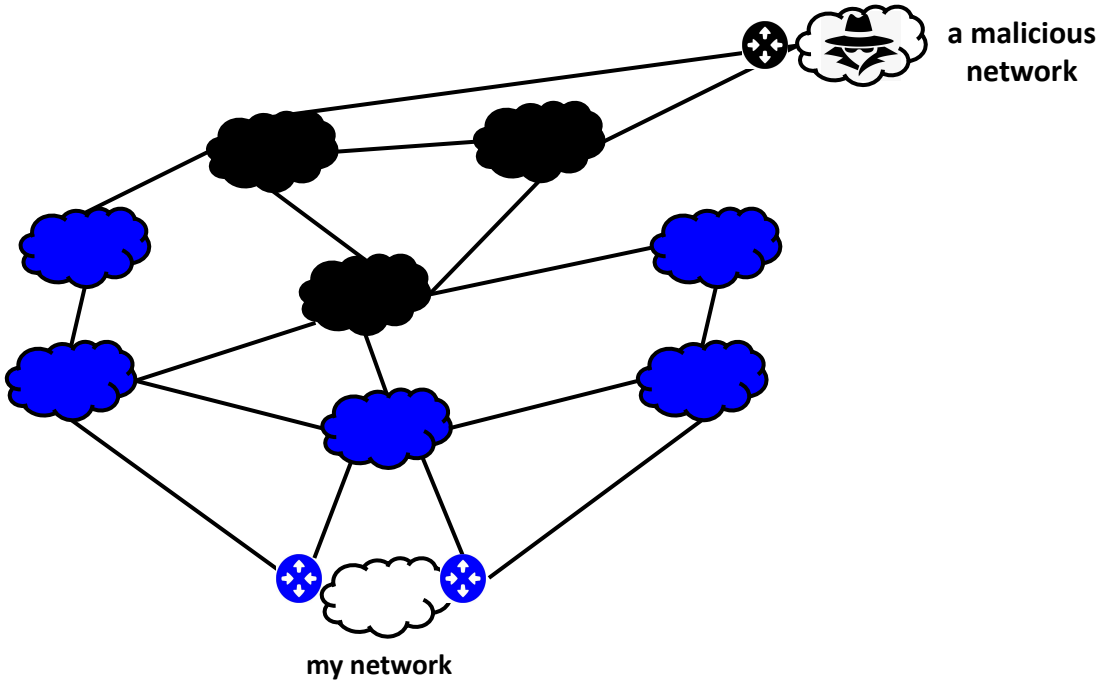
[1] ARTEMIS, open-source software, https://bgpartemis.org/

# Impact “definition”

a simple definition:

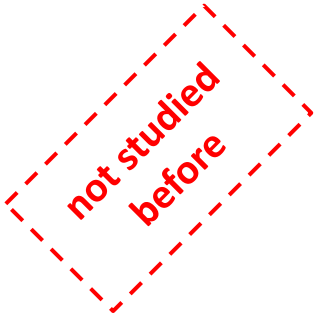
**“impact =**
**#ASes infected by the hijack”**

*(e.g., here, in the figure, impact = the number of black-colored ASes)*

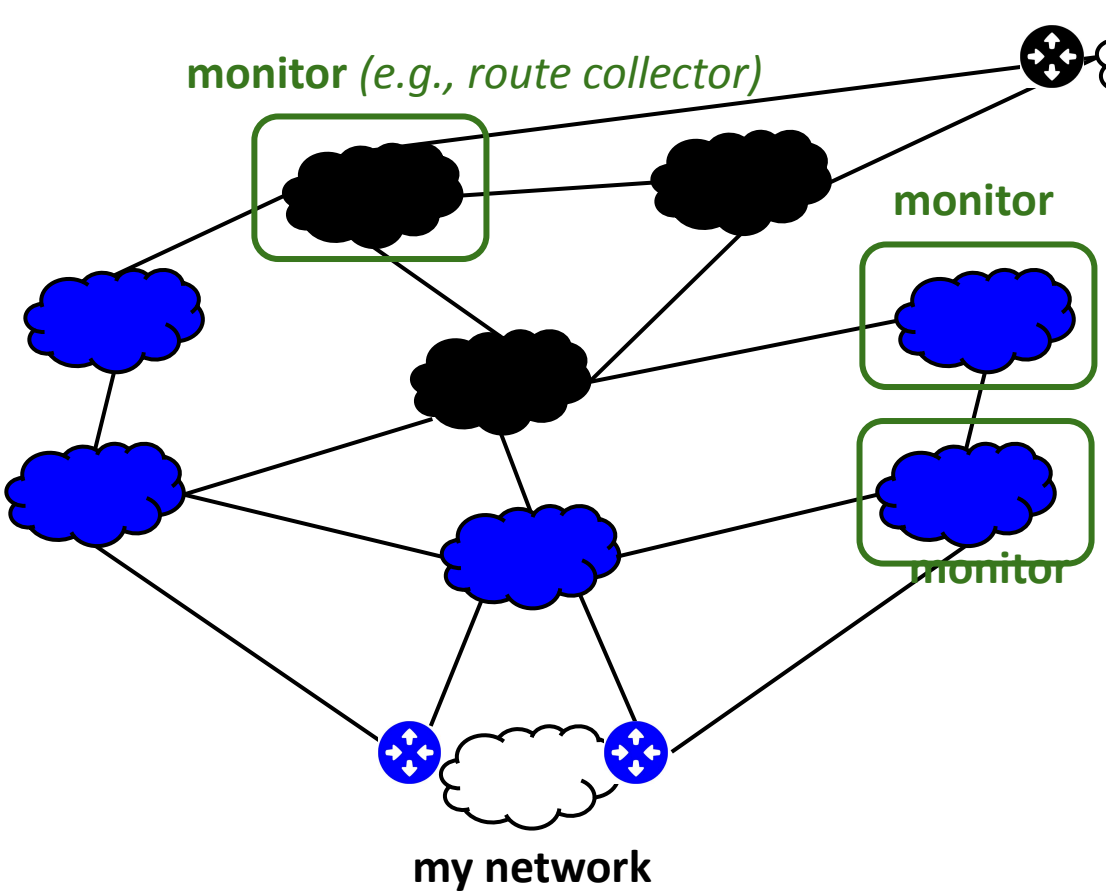


# In this work...

- **Goal → Estimate the impact of an ongoing hijack through measurements**
  - If we use Internet monitoring infrastructure (e.g., route collectors), will estimations be accurate?
  - How to design efficient estimation methodologies?



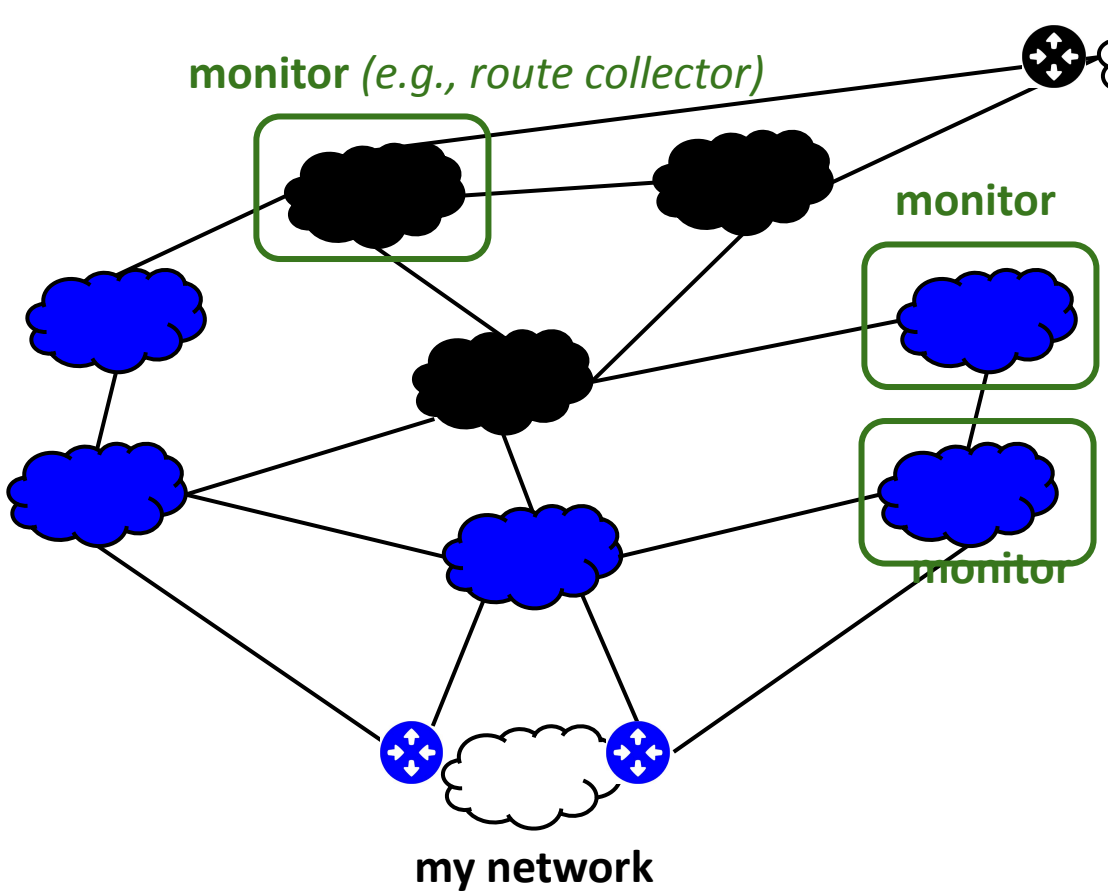
# Impact estimation: an example




$$\text{actual impact} = \frac{3}{5 + 3} = 0.375$$

$$\text{estimated impact} = \frac{1}{2 + 1} = 0.333$$

# Impact estimation: an example



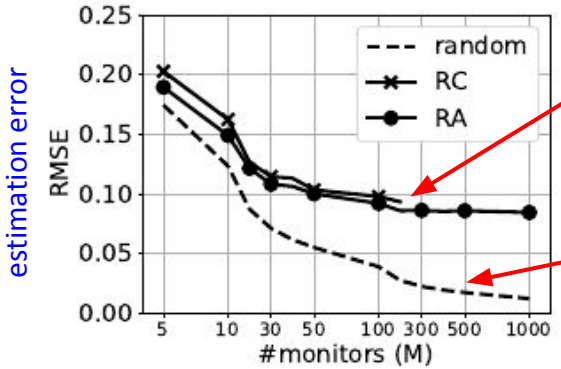

$$\text{actual impact} = \frac{3}{5+3} = 0.375$$

$$\text{estimated impact} = \frac{1}{2+1} = 0.333$$

actual vs. estimated value
→ estimation error

# Estimation accuracy: measurement bias

- **Sampling in theory (random sampling) vs. Sampling in practice (public infrastructure)**
  - with public infrastructure (Route Collectors, RIPE Atlas probes)




**public infrastructure**

**theory** (random sampling)

**Key findings:**

- The error of public infrastructure does not get better than **9%-10%**
  → due to **location bias**; public infrastructure is not deployed uniformly around the world
- **50 samples** from public monitors are enough
  → insights for lightweight measurements

# How to avoid the bias?

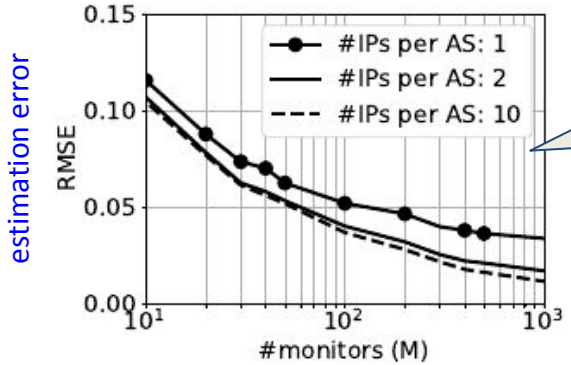
- Approach 1: do **ping measurements** instead of public infrastructure measurements...
  - we can ping any AS → random sampling (i.e., best performance)
  - be careful of ping failures
  - *Goal*: propose an accurate methodology (who to ping? how many pings?)

- Approach 2: based on **public-infrastructure**…
  - *Goal*: identify correlations & remove the measurement bias

# Approach 1: ping-based impact estimator

**Ping-based impact estimator**

1. Find "pingable" IP addresses for every AS [*ANT Lab's IP hitlist*]
2. Ping multiple ($N_{IP}$) IP addresses per AS
3. If at least one ping reply from an AS → the AS is not affected by the hijack



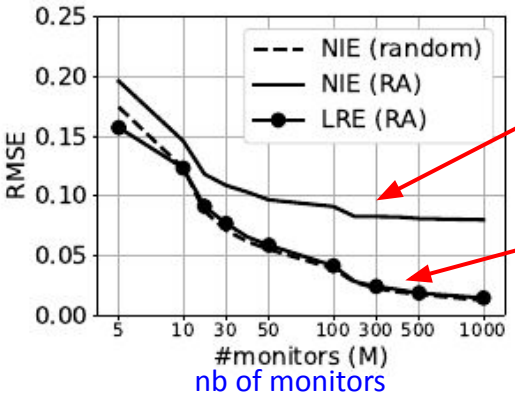

**Key findings:**

- Ping $N_{IP}$ **= 2** "pingable" IP addresses in each AS for a good accuracy *(no significant gains for $N_{IP}$>3)*

# Approach 2: public infrastructure estimator

**Public infrastructure measurements + Machine Learning (ML)**

1. Collect past measurements of hijacks from public infrastructure
2. Fit an ML model (least-squares estimator "LRE")
3. Collect measurements for the ongoing hijack
4. Estimate the impact using the model




**public infrastructure**

**public infrastructure with LRE**

**Key findings:**

- LRE eliminates the bias!
- Training the LRE with only a few (~20) past measurements was ok in our experiments

# Summarizing...

**Take-home messages:**

- Estimations with Internet monitoring infrastructure are *biased* (~10% error)
- Proposed solutions:
  - *without infrastructure:* use pings from your own network; a few 100s of pings → ~2% error
  - *with infrastructure:* "smartly" remove bias with machine learning

**Ongoing work:**

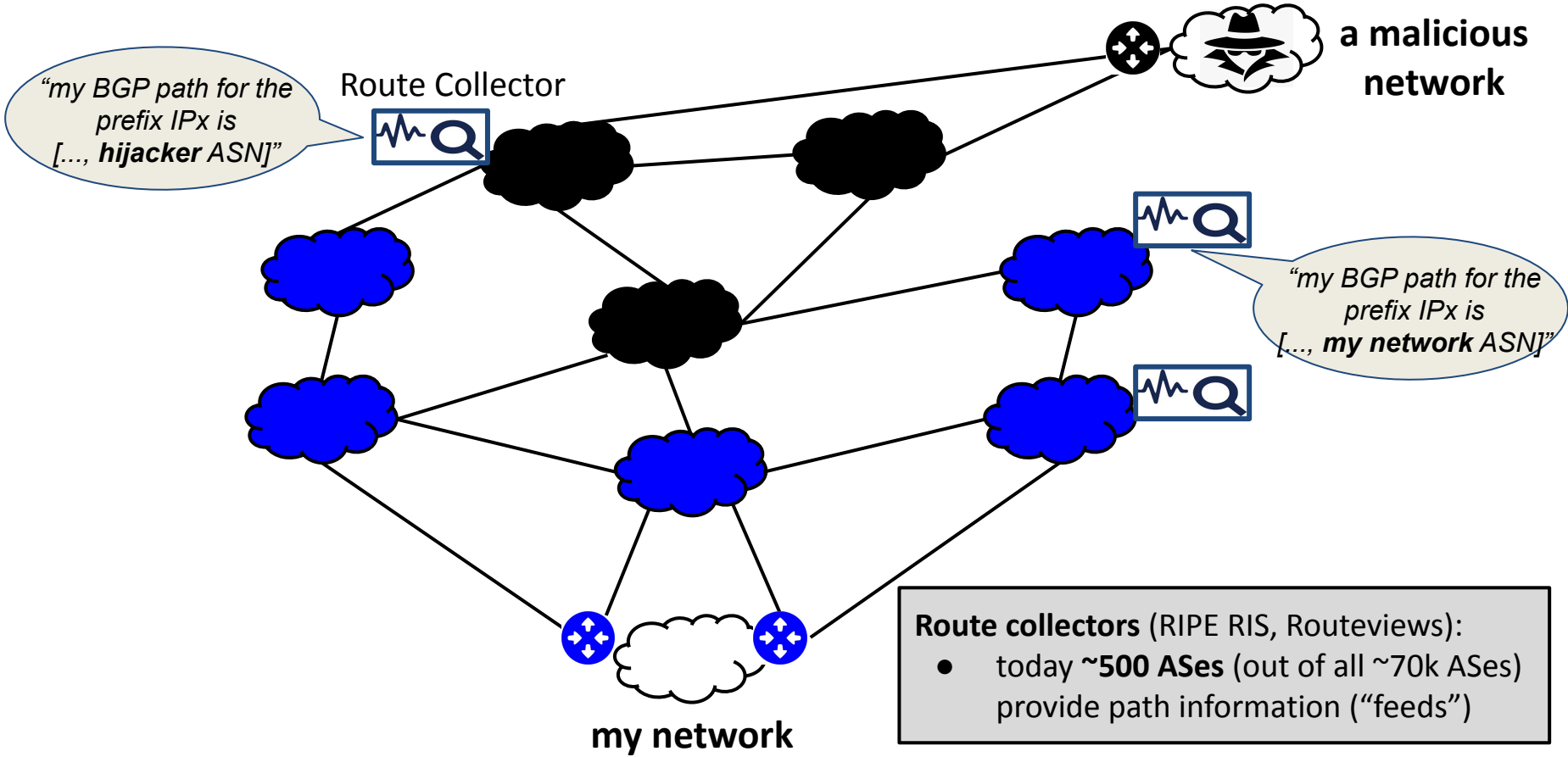
- "AI4NetMon" project *(funded by RIPE NCC, RACI funding 2021)*
  - goal: reduce bias in infrastructure - more info: *https://sermpezis.github.io/ai4netmon/*

**Feedback**

- How useful do you find the proposed solutions?
- Other important use cases (i.e. not hijacking) to study the bias?
  → *tell us what's important for you in our **survey**! [https://forms.gle/9xJpYFw3PBo8KShz9]*
- Contact us: *sermpezis@csd.auth.gr*
- Join the "RIS JAM" chat at RIPE83 [Thursday, 25 Nov. @ 17:00-18:30 (UTC+1)]

# Backup slides

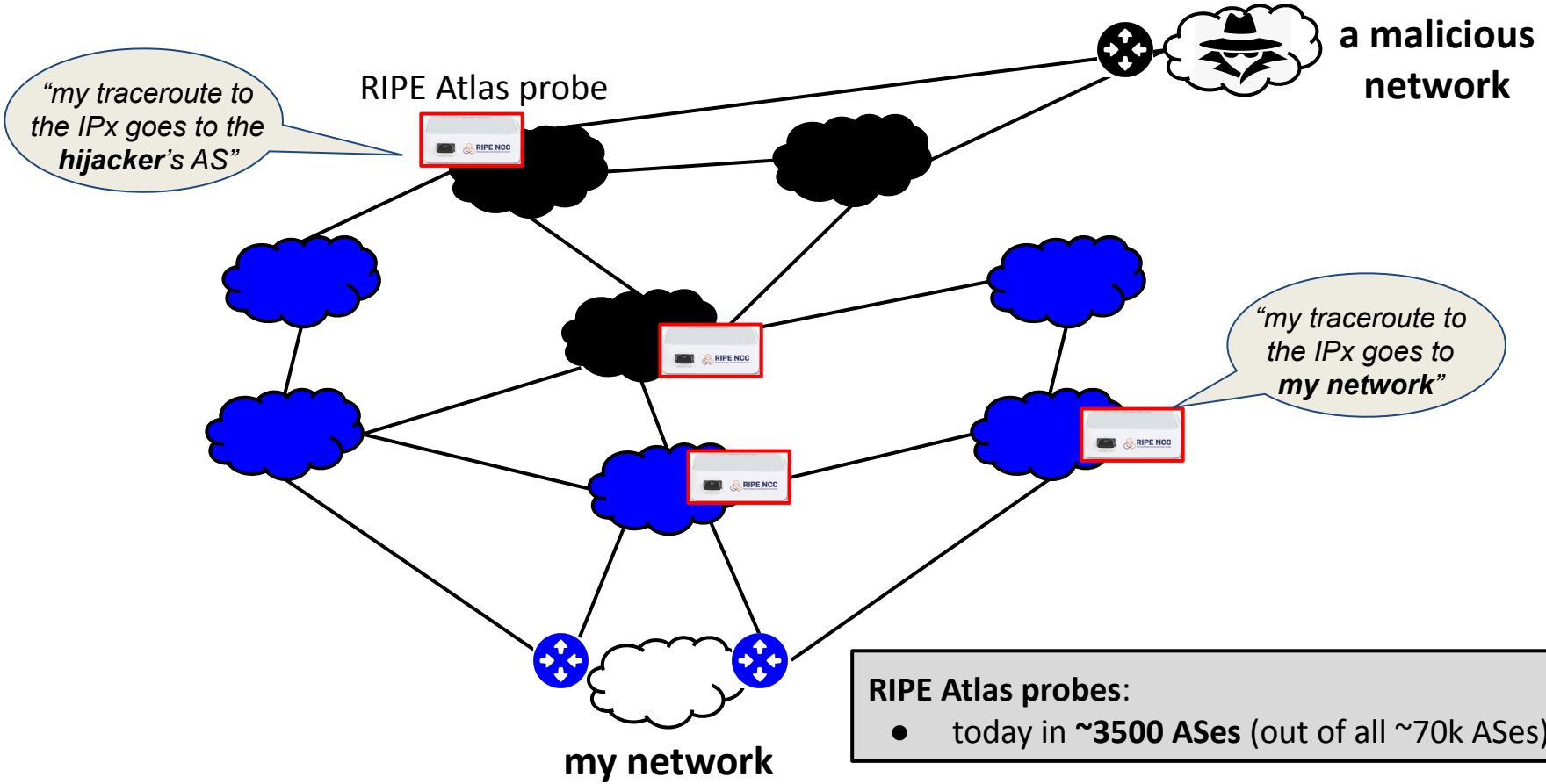
# Measurements: BGP paths




**Route collectors** (RIPE RIS, Routeviews):

- today **~500 ASes** (out of all ~70k ASes) provide path information ("feeds")

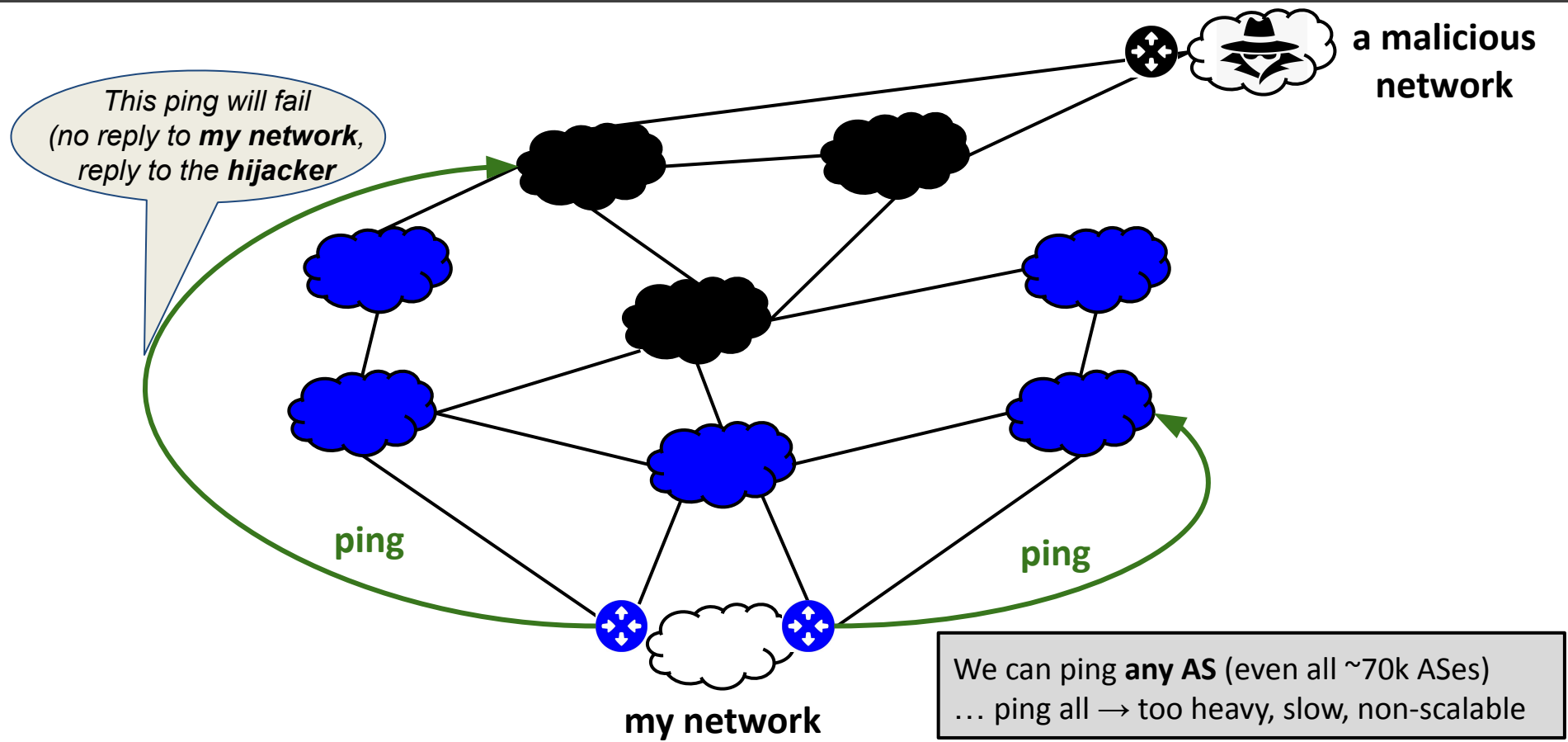
# Measurements: traceroutes




**RIPE Atlas probes**:

- today in **~3500 ASes** (out of all ~70k ASes)

# Measurements: pings




We can ping **any AS** (even all ~70k ASes)
… ping all → too heavy, slow, non-scalable

# Hijack impact estimation with measurements

- Hijack impact == number of infected ASes
  - ***"infected AS"*** == an AS that routes its traffic to the hijacker AS

$$\textbf{actual impact} = \frac{\#\ \text{infected } \textbf{ASes}}{\#\ \text{total } \textbf{ASes}}$$

- Estimate hijack impact
  - measure some ASes (BGP paths, traceroute, pings)
  - measured AS == ***"monitor"***

$$\textbf{estimated impact} = \frac{\#\ \text{infected } \textbf{monitors}}{\#\ \text{total } \textbf{monitors}}$$