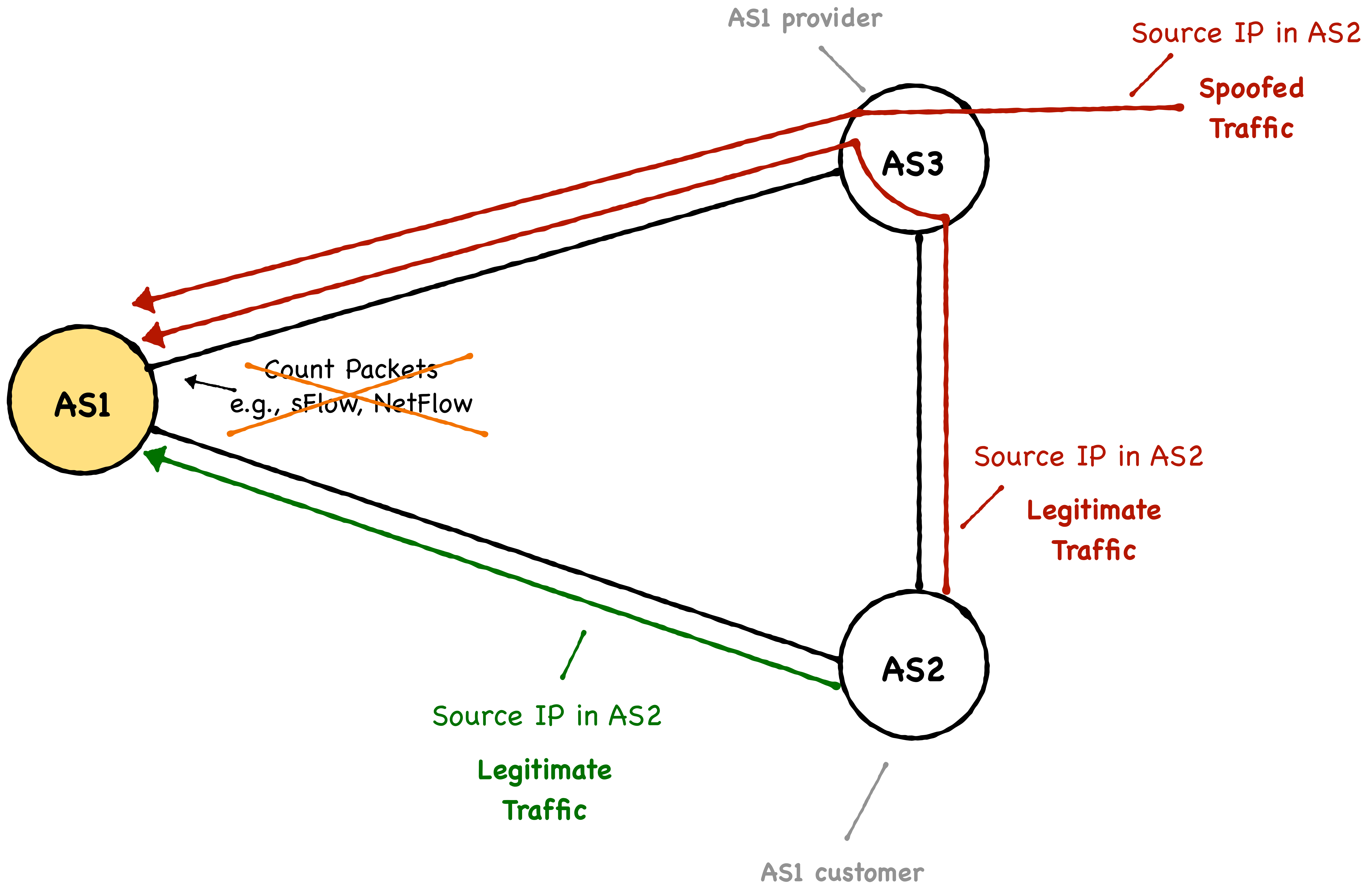
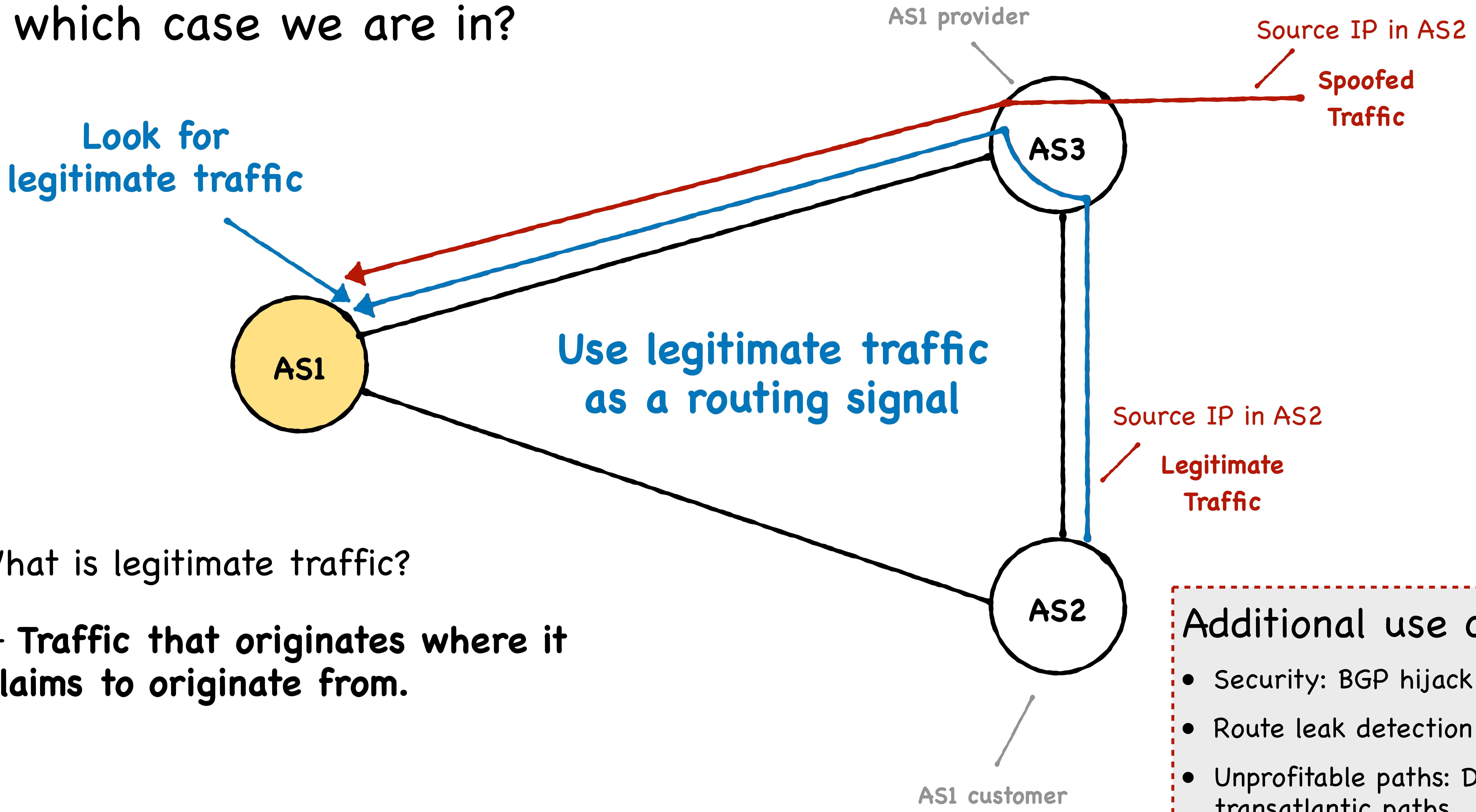
Bad Packets Come Back, Worse Ones Don't

Petros Gigis, Mark Handley, Stefano Vissicchio

University College London


AS1 provider
Source IP in AS2
Spoofed
Traffic
AS3
AS1
~~Count Packets~~
~~e.g., sFlow, NetFlow~~
Source IP in AS2
Legitimate
Traffic
AS2
Source IP in AS2
Legitimate
Traffic
AS1 customer

How can we distinguish which case we are in?



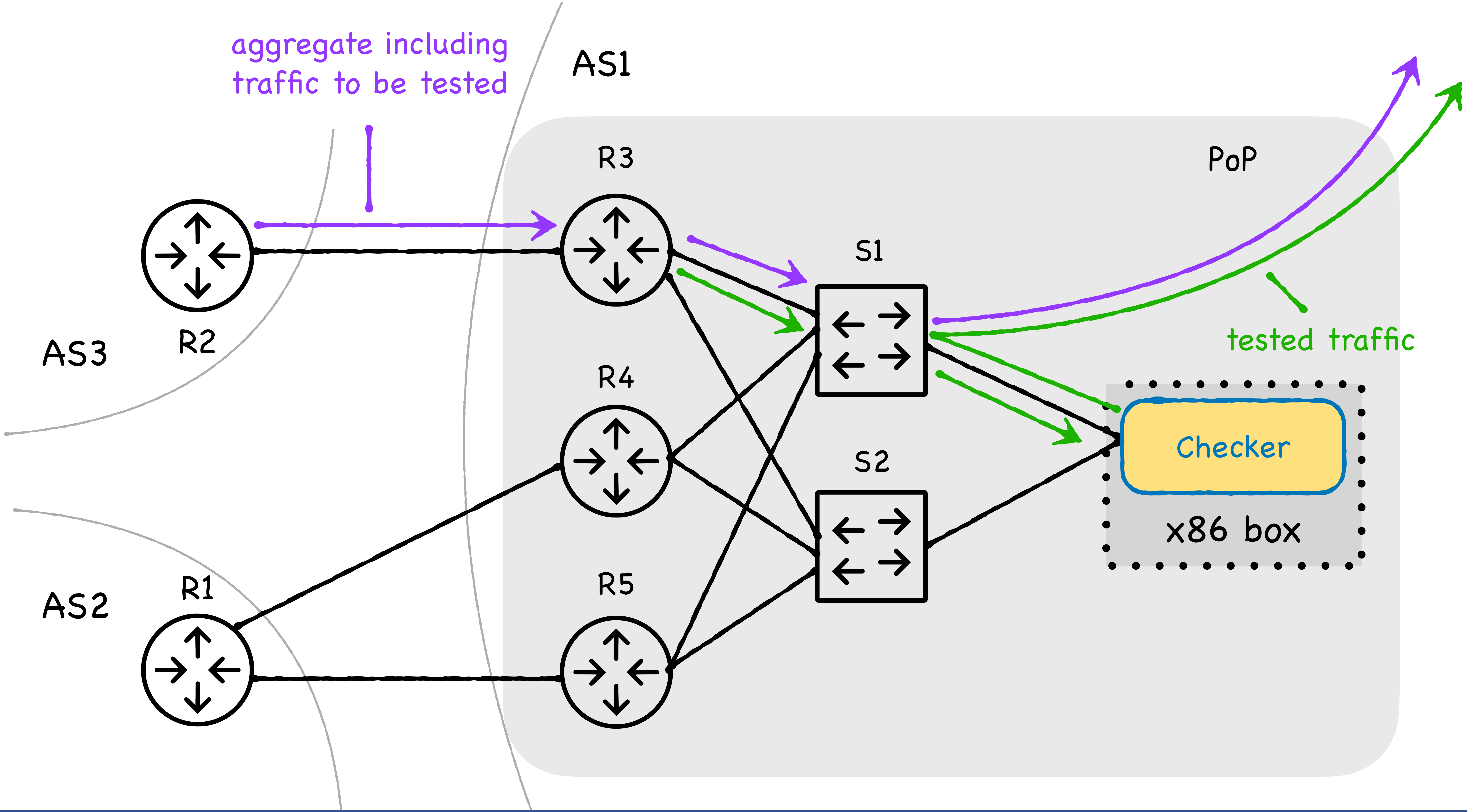
What is legitimate traffic?

– Traffic that originates where it claims to originate from.

Additional use cases:

- Security: BGP hijack detection
- Route leak detection
- Unprofitable paths: Detecting expensive transatlantic paths

Redirecting traffic to an analysis box



How to detect legitimate traffic?

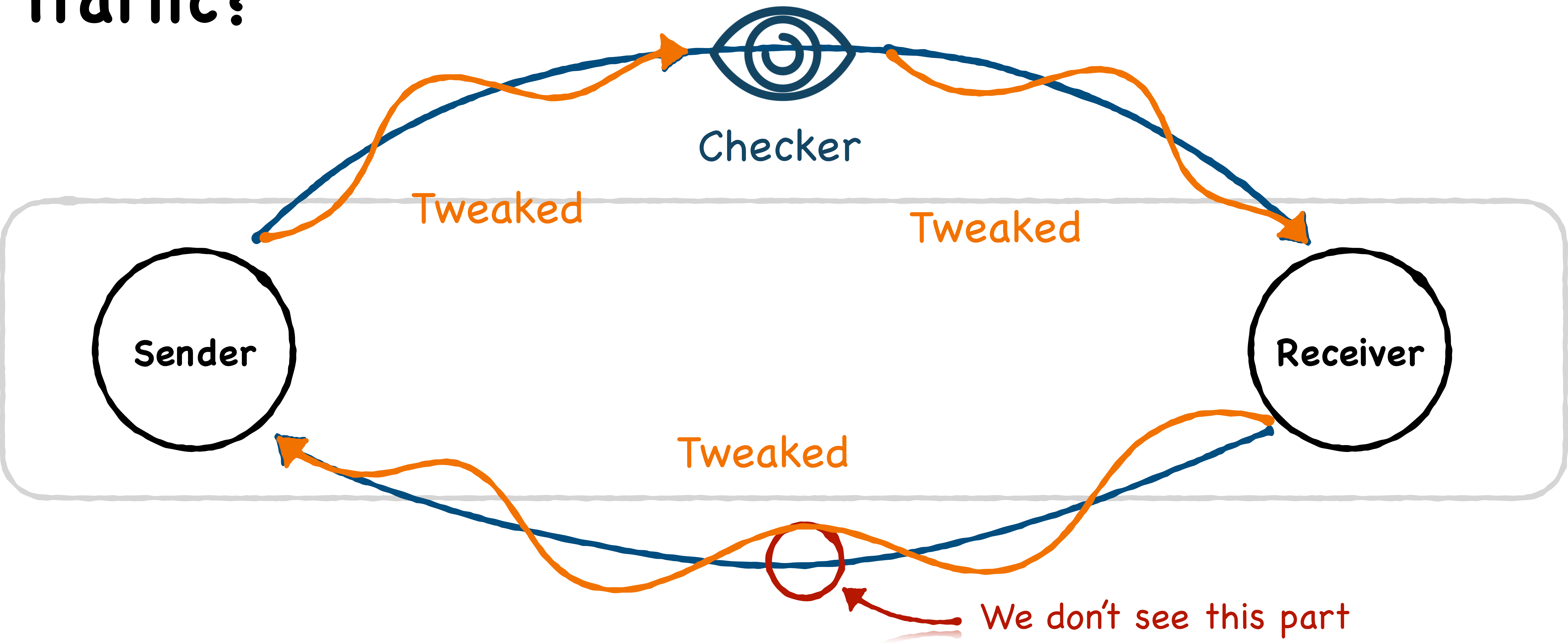
- Hard to distinguish from spoofed traffic.
- Expected to respond to feedback in a closed-loop communication.

💡 **Closed-loop traffic** can be used as **a proxy to detect legitimate traffic**.

How to detect closed-loop traffic?

– Tweak traffic.

TCP is the perfect candidate.



What is the easiest way to tweak TCP traffic?

— Drop a data packet.

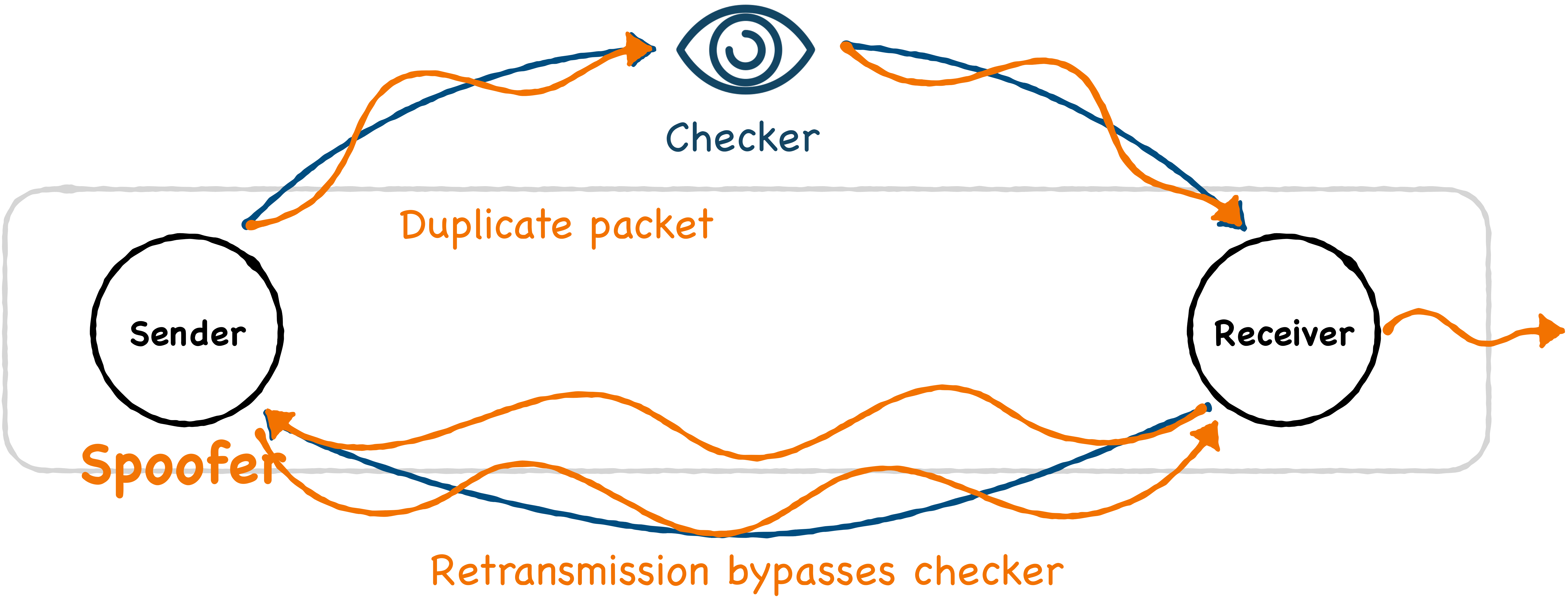
Is dropping one packet enough?

- So, we drop a data packet:
  - If a **retransmission is observed**, the **flow is closed-loop**.
  - If **no retransmission is observed**, the **flow is not closed-loop (spoofed).**

- **What could go wrong?**

When not closed-loop →

When closed-loop →

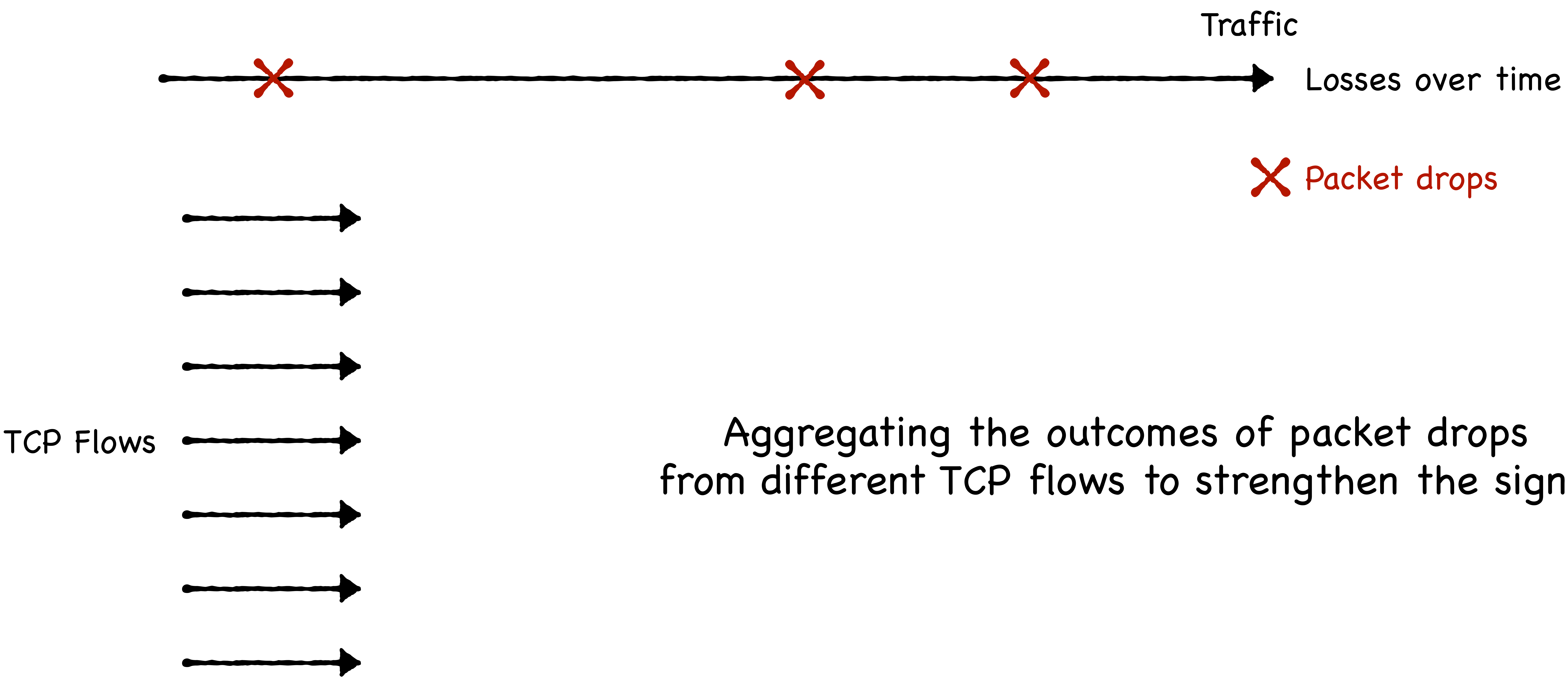


The signal from **a single** data packet drop is **weak and noisy**!

How can we improve this?

💡 Drop **a few** data packets to **gain confidence.**

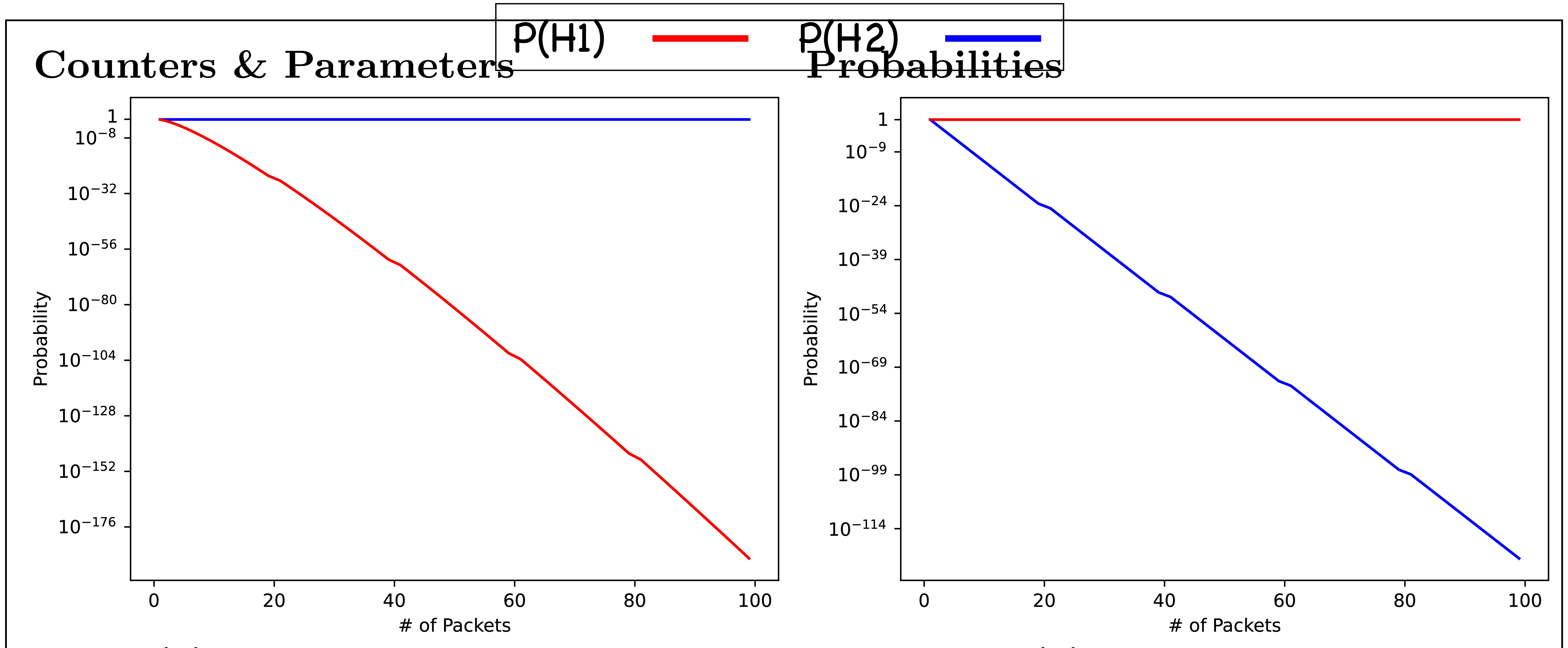
Our approach in practice



Aggregating the outcomes of packet drops from different TCP flows to strengthen the signal.

(The) **Penny** drops

- **Approach:** Statistical model **comparing two competing hypotheses:**
  - **H1**: hypothesis that the traffic **is closed-loop**.
  - **H2**: hypothesis that the traffic **is not closed-loop (spoofed).**



(a) Not closed-loop traffic

(b) Closed-loop traffic

Does it work?

- Complications:
  - Deal with (i) **the TCP protocol**, (ii) **the network conditions** and (iii) **malicious sources**.
- Evaluation with NS-3 simulator:
  - **Multiple TCP variants**: NewReno, Cubic, …
  - **Diverse network conditions**: upstream/downstream losses, queues, …
  - **Varied input traffic**: closed-loop, worst-case not closed-loop, mixed traffic, short/long flows, …
  - **Different Penny parameters**: packet drop rate, timers, …

Summary of evaluation results

- **Worst-case chances of false alarms are 1 in 1 million tests.**

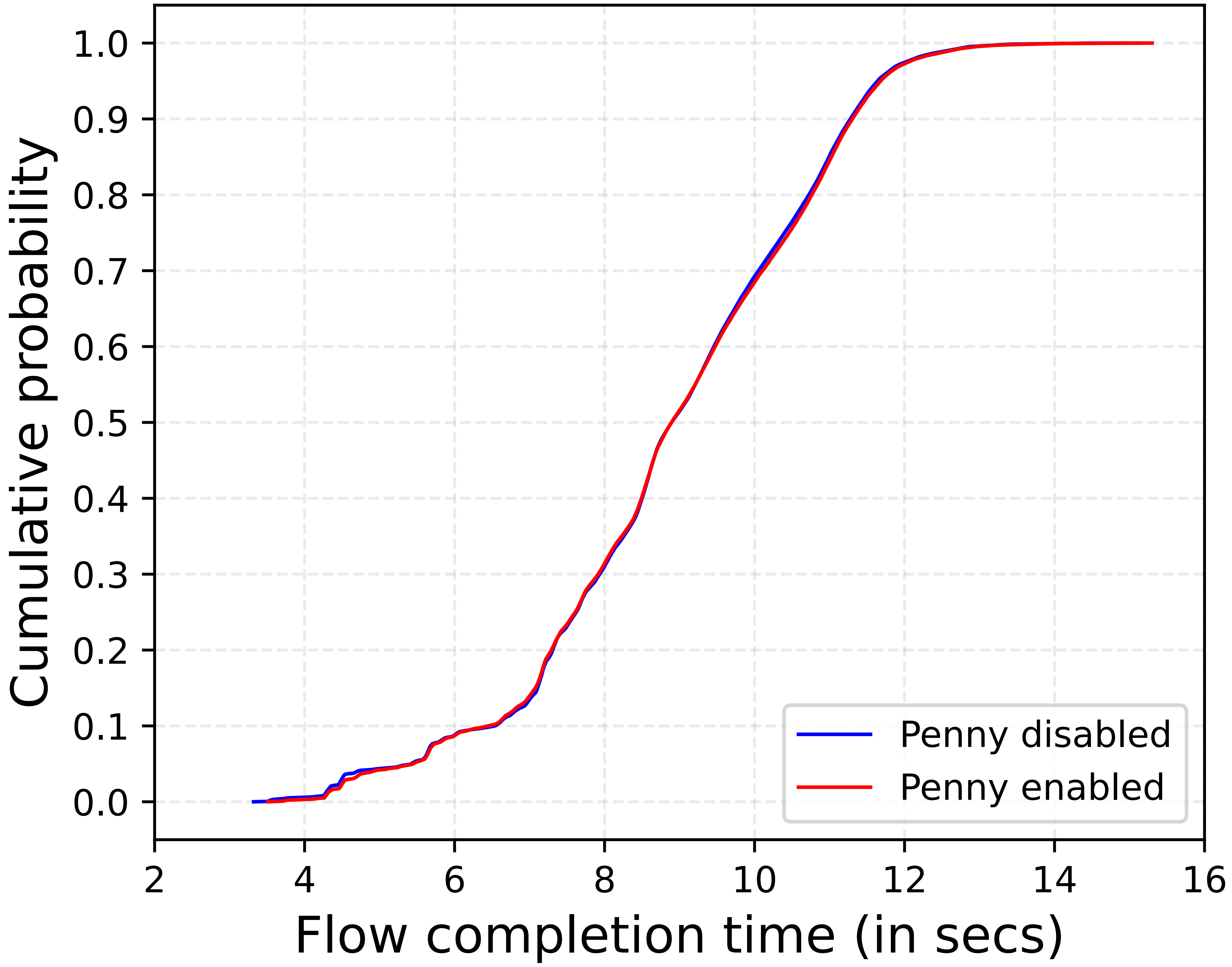
- Penny **works even in cases of mixed traffic.**
  - Remember: we are looking for legitimate traffic.
  - Can find legitimate traffic in aggregates with 90% spoofed traffic.

- Penny has a **very low impact** on the completion times of TCP flows.
  - We drop ~12 packets per test!

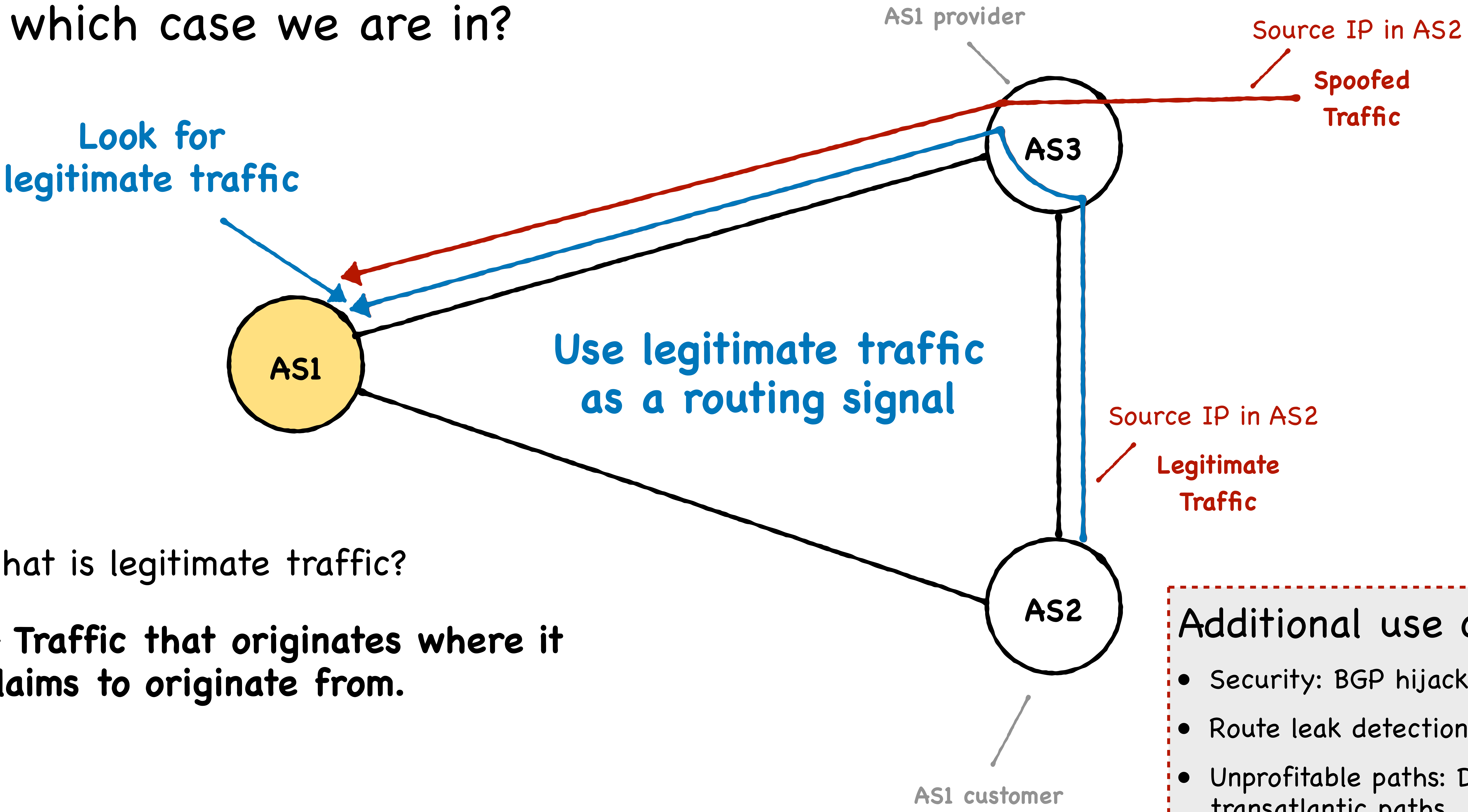
Penny's impact on flow aggregates

- Experiment setup:
  - TCP background traffic
  - 100 non-spoofed TCP flows

- Penny has a **negligible impact** on TCP flow completion times when running **on aggregates**.



How can we distinguish which case we are in?



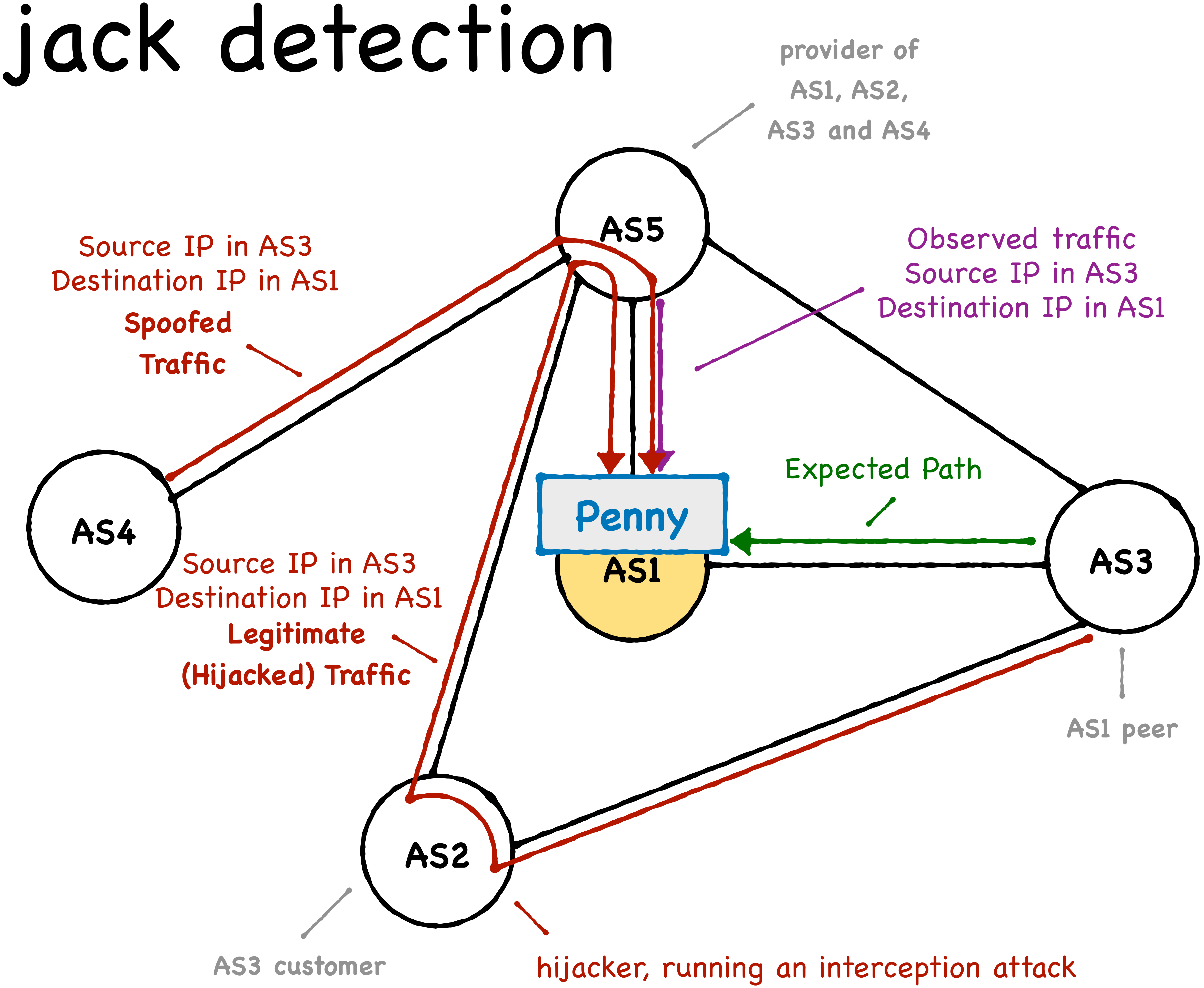
What is legitimate traffic?

– Traffic that originates where it claims to originate from.

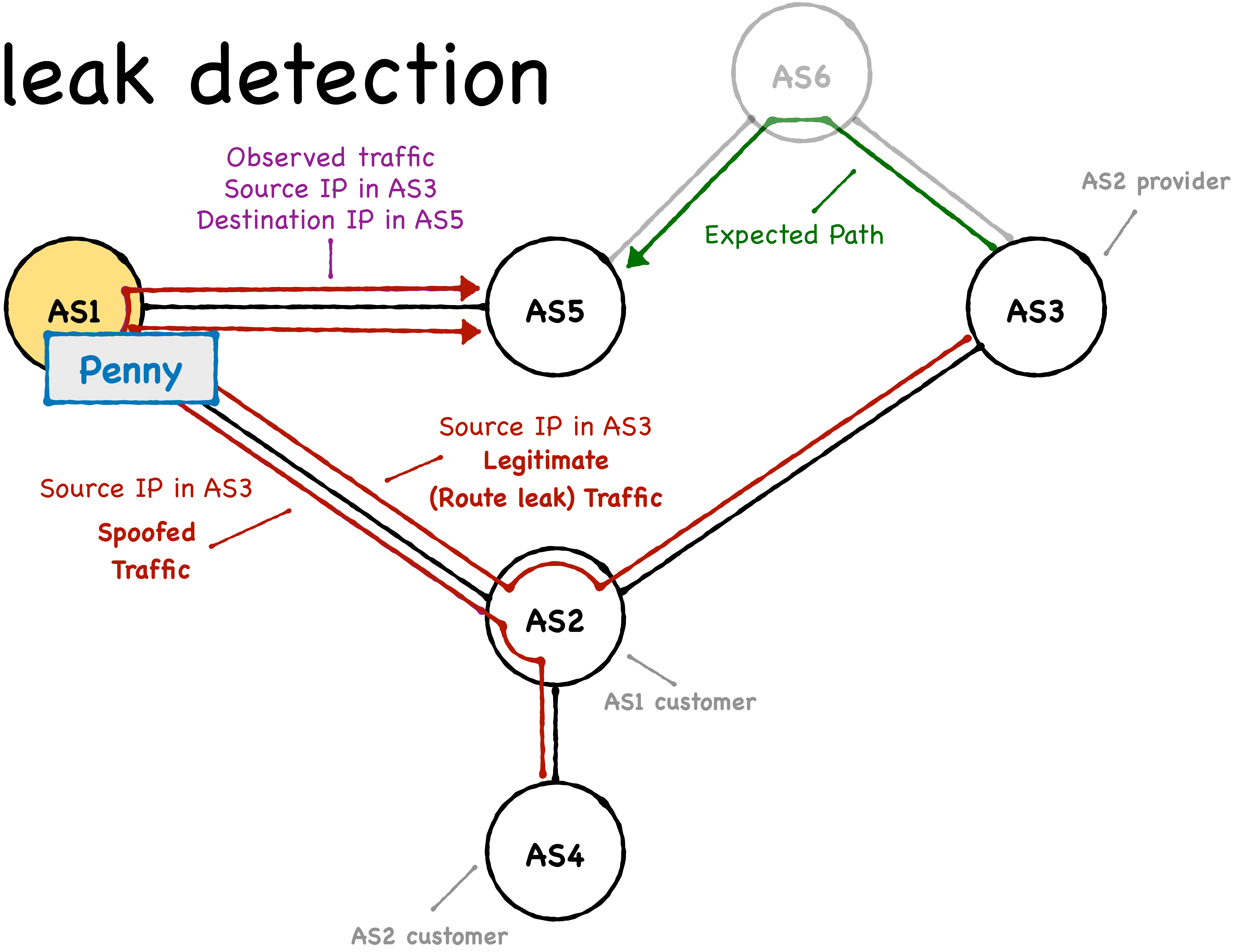
Additional use cases:

- Security: BGP hijack detection
- Route leak detection
- Unprofitable paths: Detecting expensive transatlantic paths

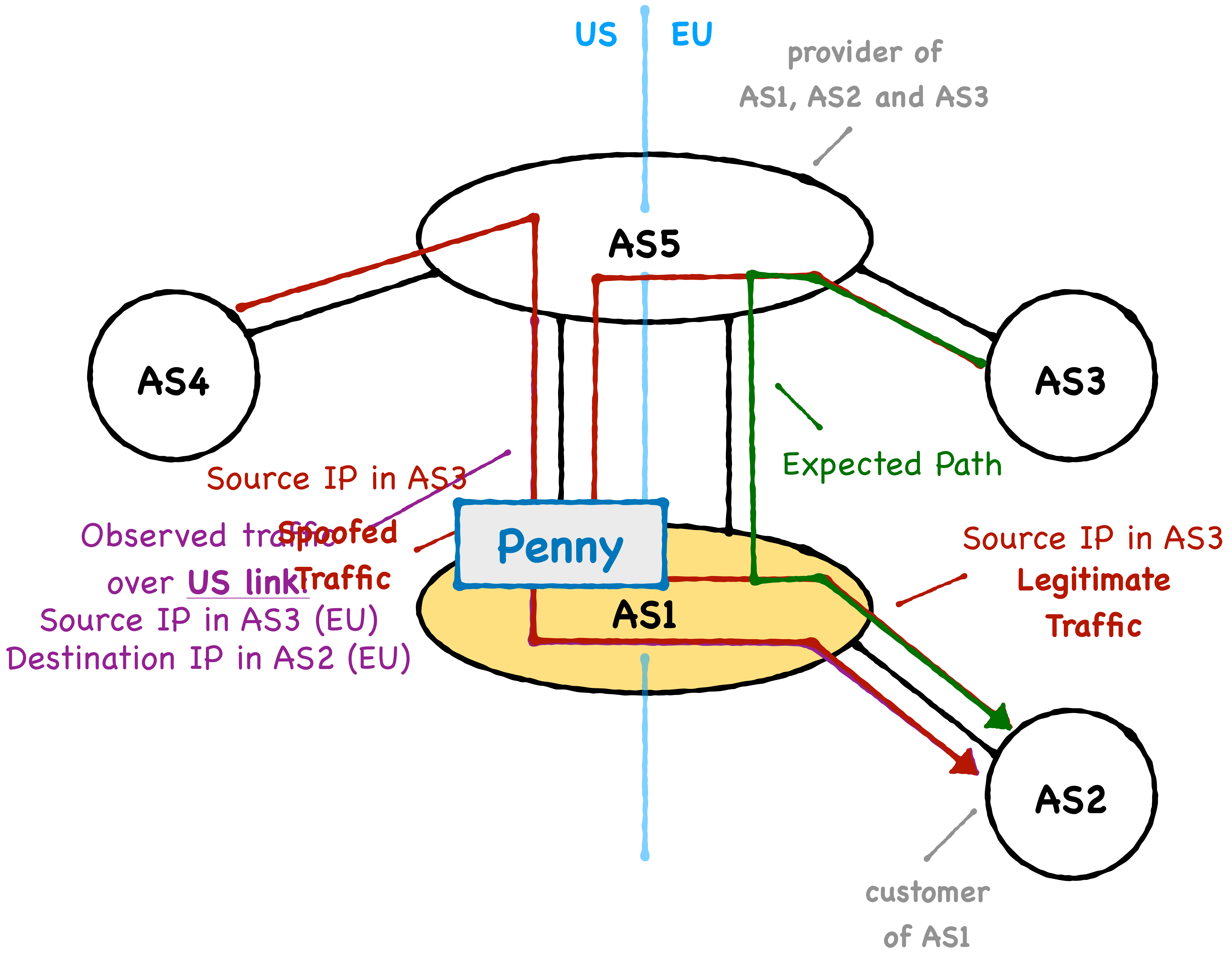
BGP hijack detection



Route leak detection



Detecting expensive transatlantic paths



provider of AS1, AS2 and AS3

Expected Path

Source IP in AS3

Spoofed Traffic

Observed traffic over **US link!**

Source IP in AS3 (EU)

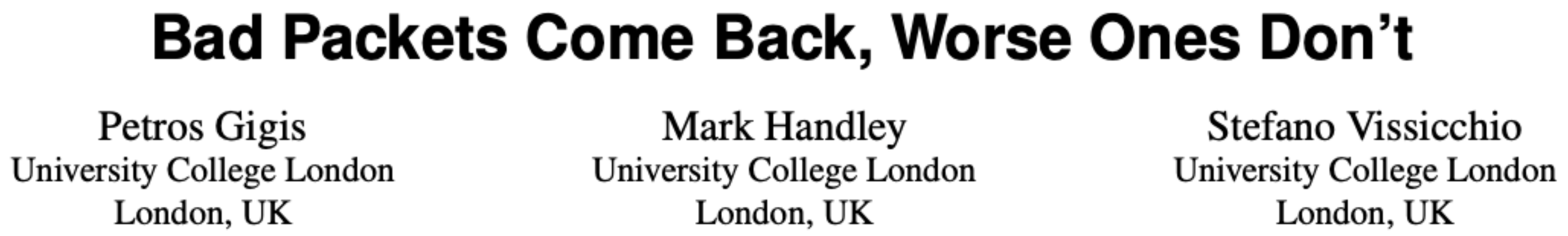
Destination IP in AS2 (EU)

Source IP in AS3

Legitimate Traffic

customer of AS1

Takeways


Bad Packets Come Back, Worse Ones Don't

Petros Gigis
University College London
London, UK

Mark Handley
University College London
London, UK

Stefano Vissicchio
University College London
London, UK

- Detecting non-spoofed traffic might be useful to detect and identify routing incidents/misconfigurations.
- Non-spoofed traffic aggregates can be detected reliably and "cheaply" by dropping a few packets.
  - Penny is our proof-of-concept.
- Would something like this be useful to you?
- Can you think of other use cases?

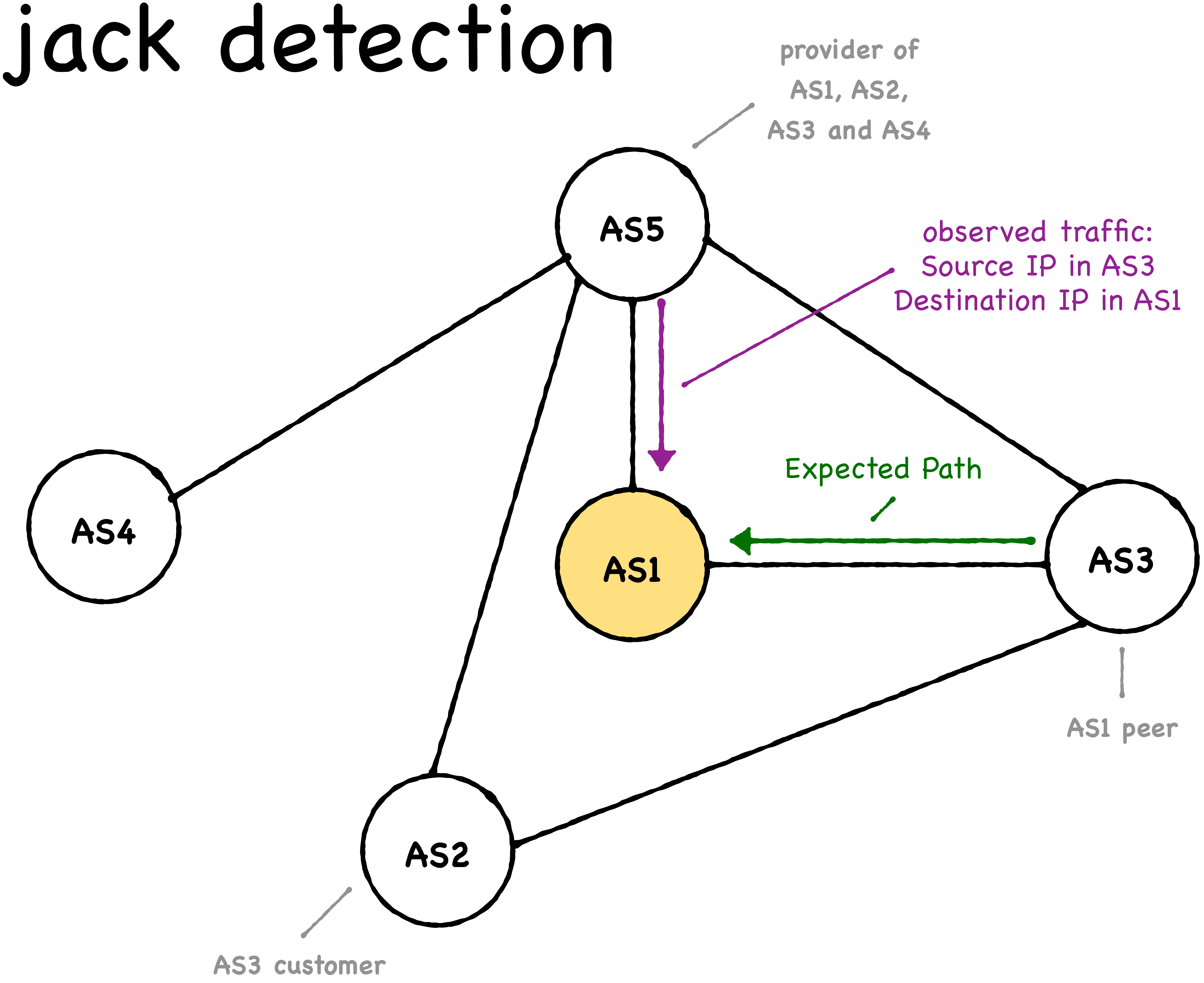
pgigis.github.io/penny

Petros Gigis p.gkigkis (at) cs.ucl.ac.uk pgigis.net

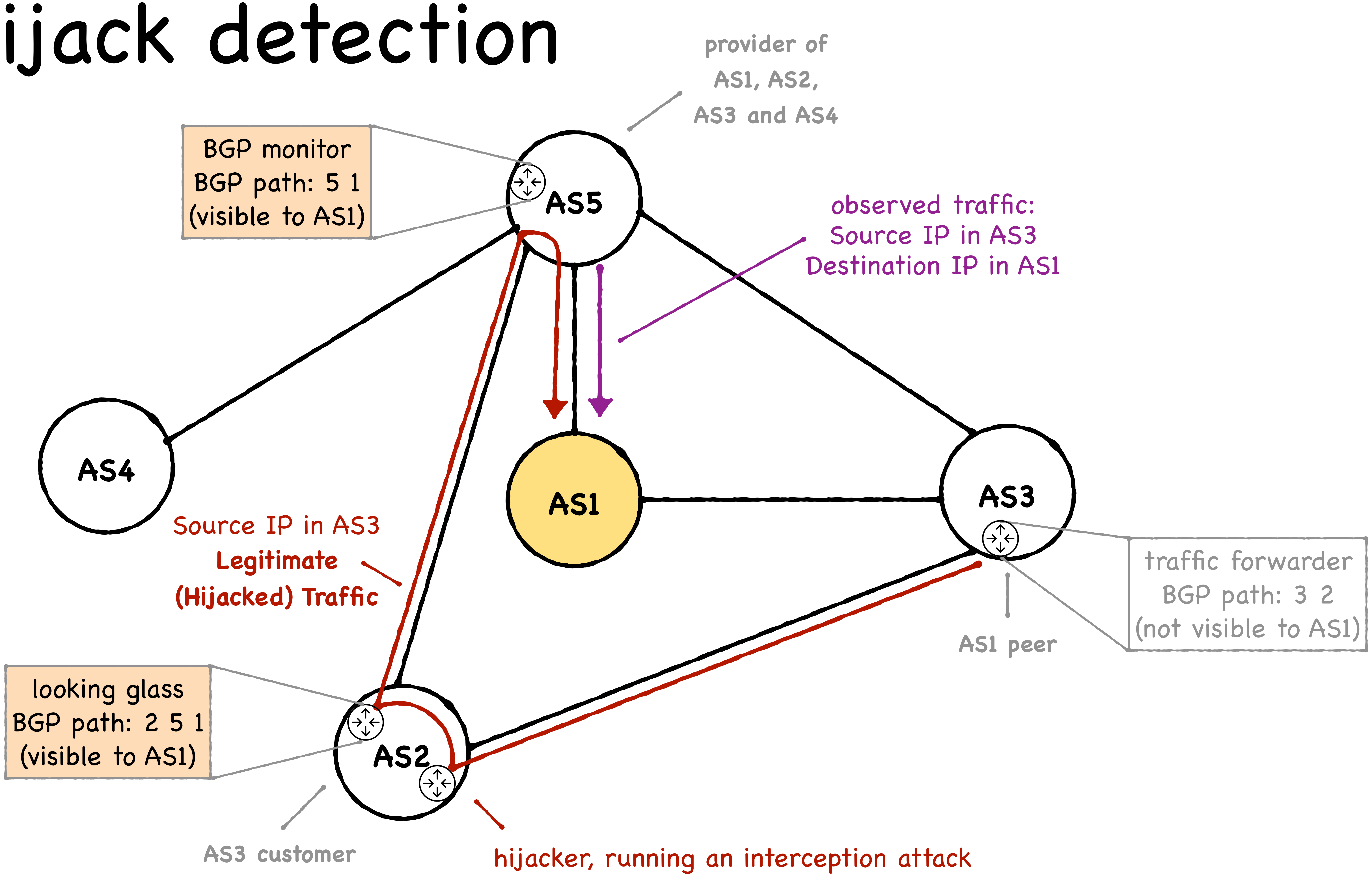
Backup Slides

Additional use cases in detail

BGP hijack detection


provider of AS1, AS2, AS3 and AS4
AS5
observed traffic: Source IP in AS3 Destination IP in AS1
Expected Path
AS4
AS1
AS3
AS1 peer
AS2
AS3 customer

BGP hijack detection



provider of AS1, AS2, AS3 and AS4

BGP monitor
BGP path: 5 1
(visible to AS1)

observed traffic:
Source IP in AS3
Destination IP in AS1

Source IP in AS3
Legitimate (Hijacked) Traffic

traffic forwarder
BGP path: 3 2
(not visible to AS1)

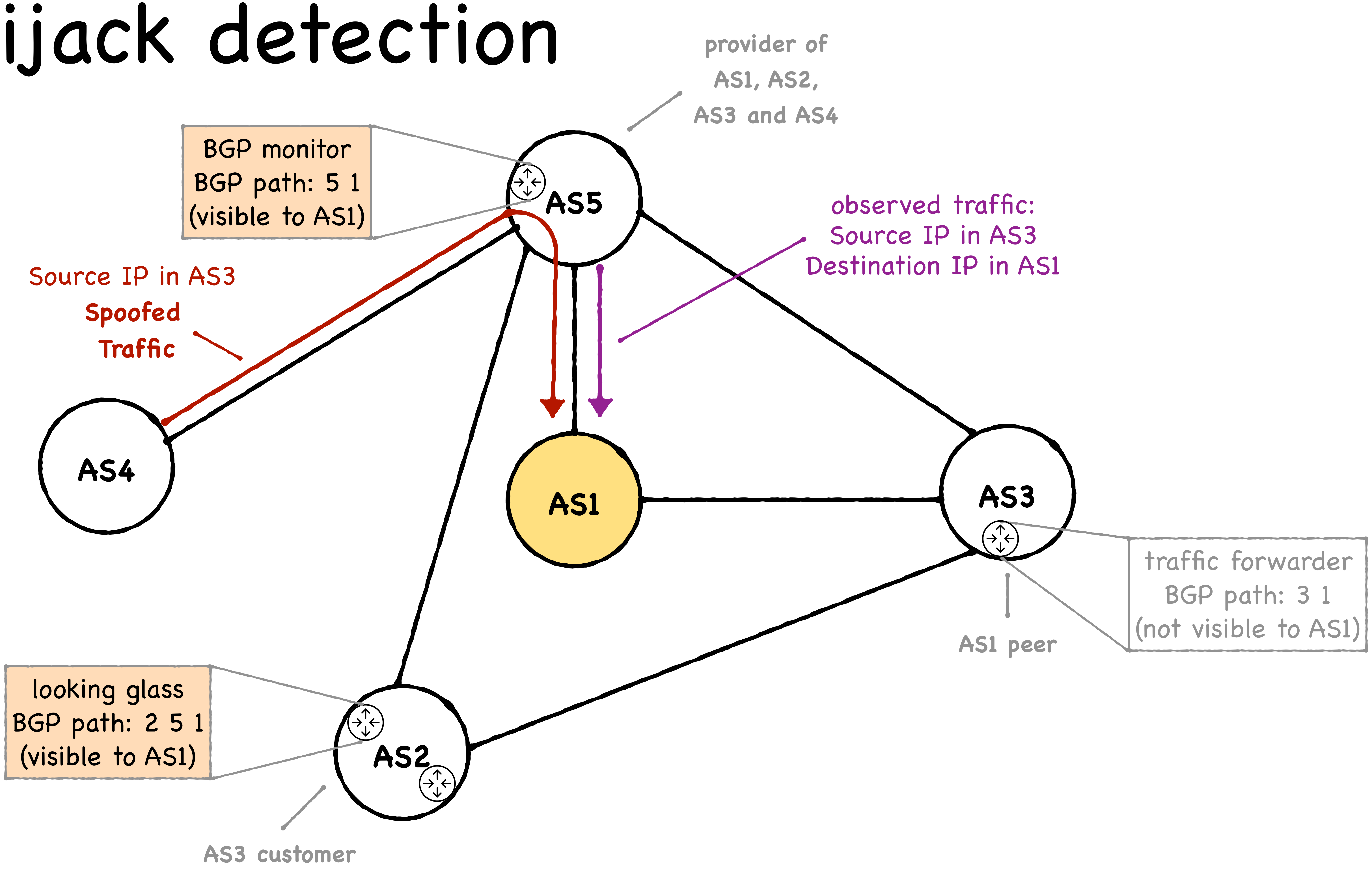
AS1 peer

looking glass
BGP path: 2 5 1
(visible to AS1)

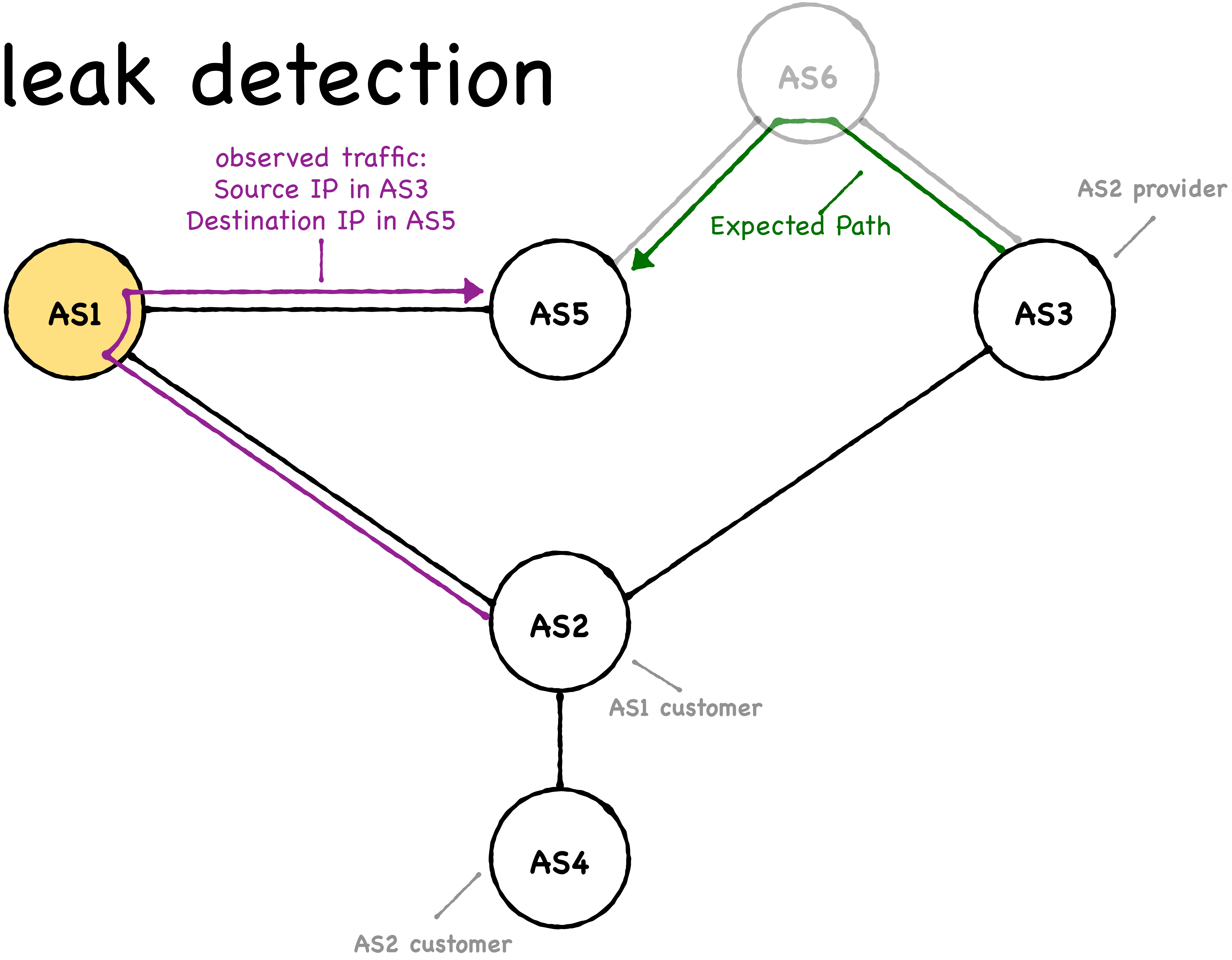
AS3 customer

hijacker, running an interception attack

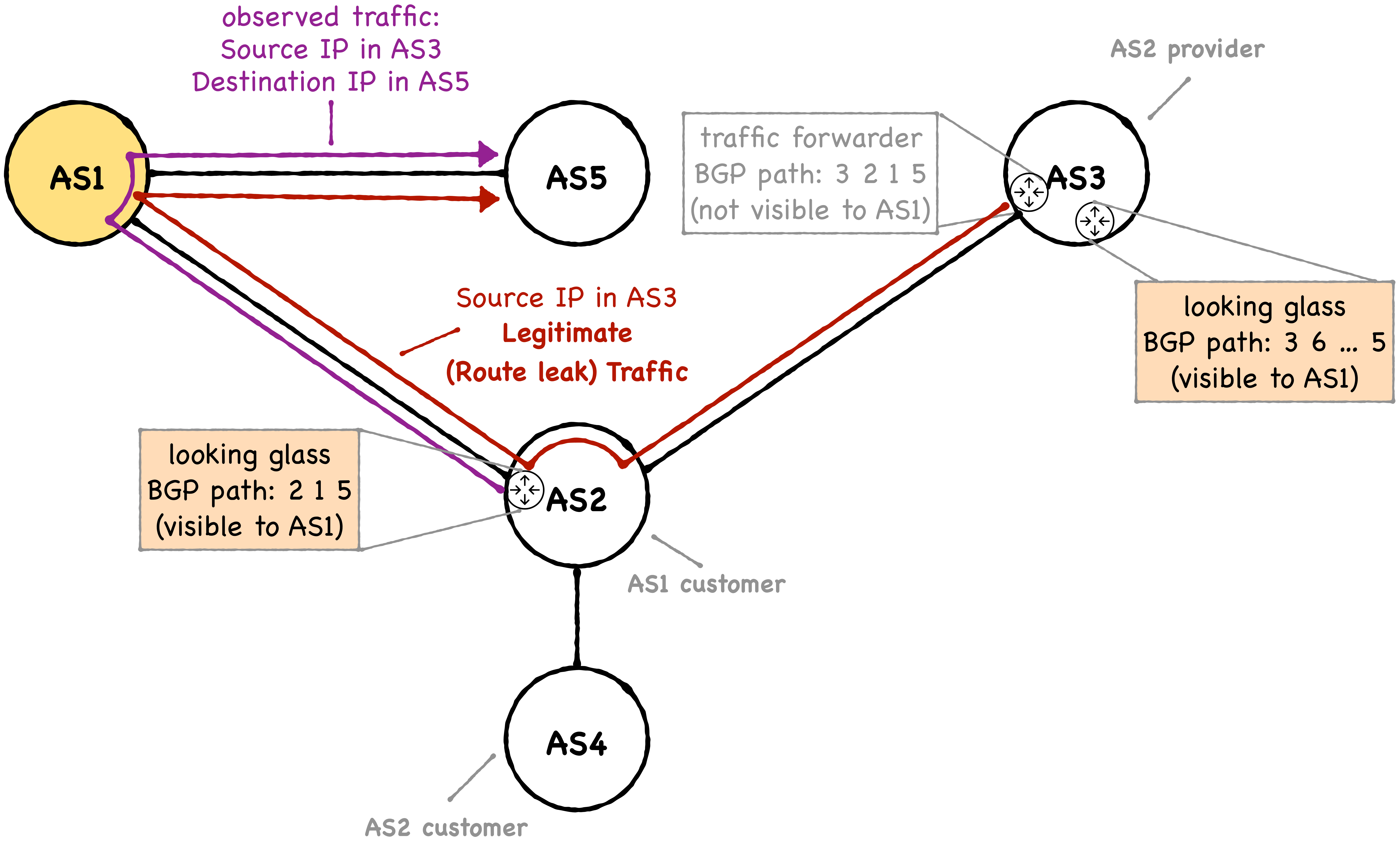
BGP hijack detection



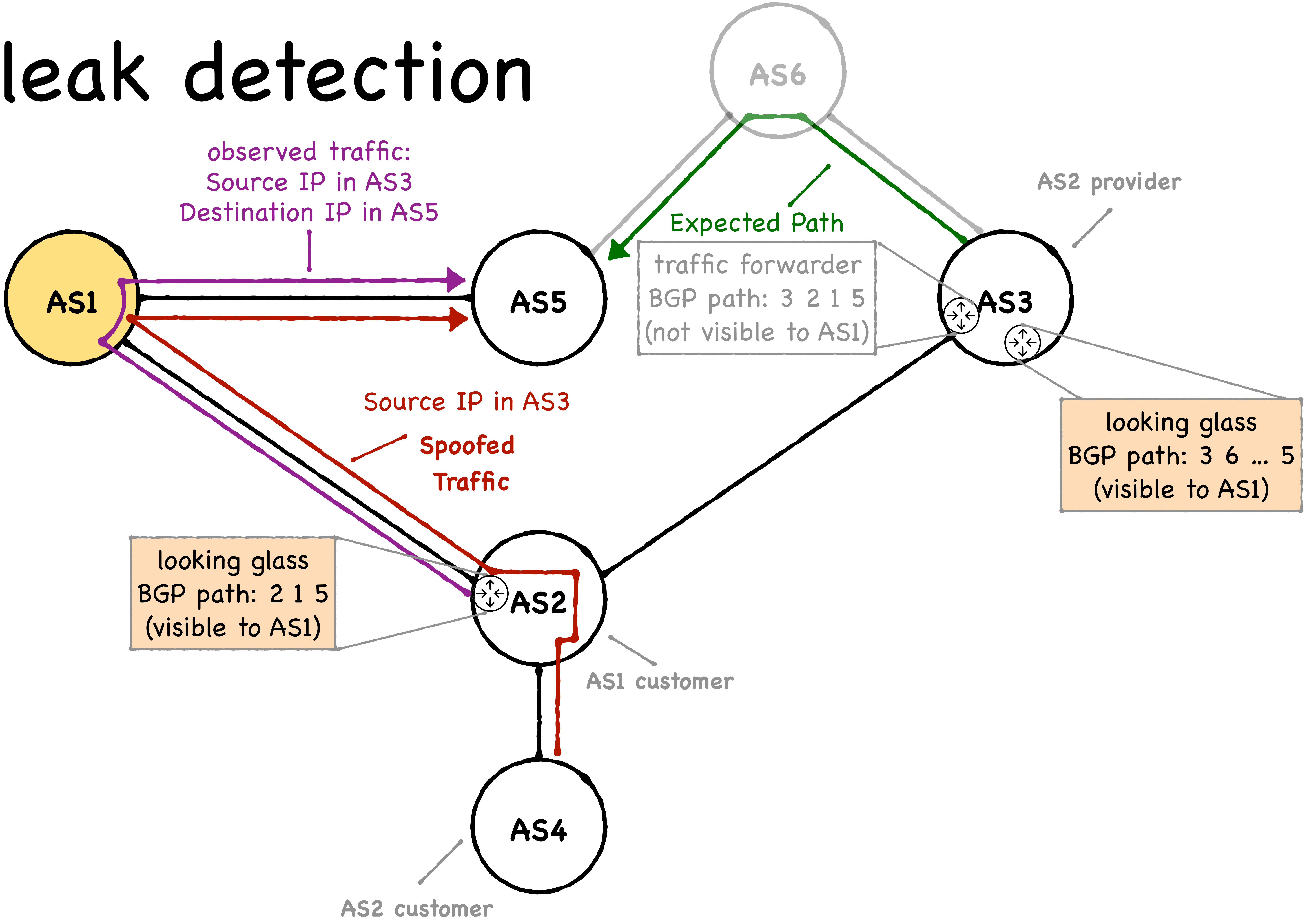
Route leak detection



Route leak detection



Route leak detection

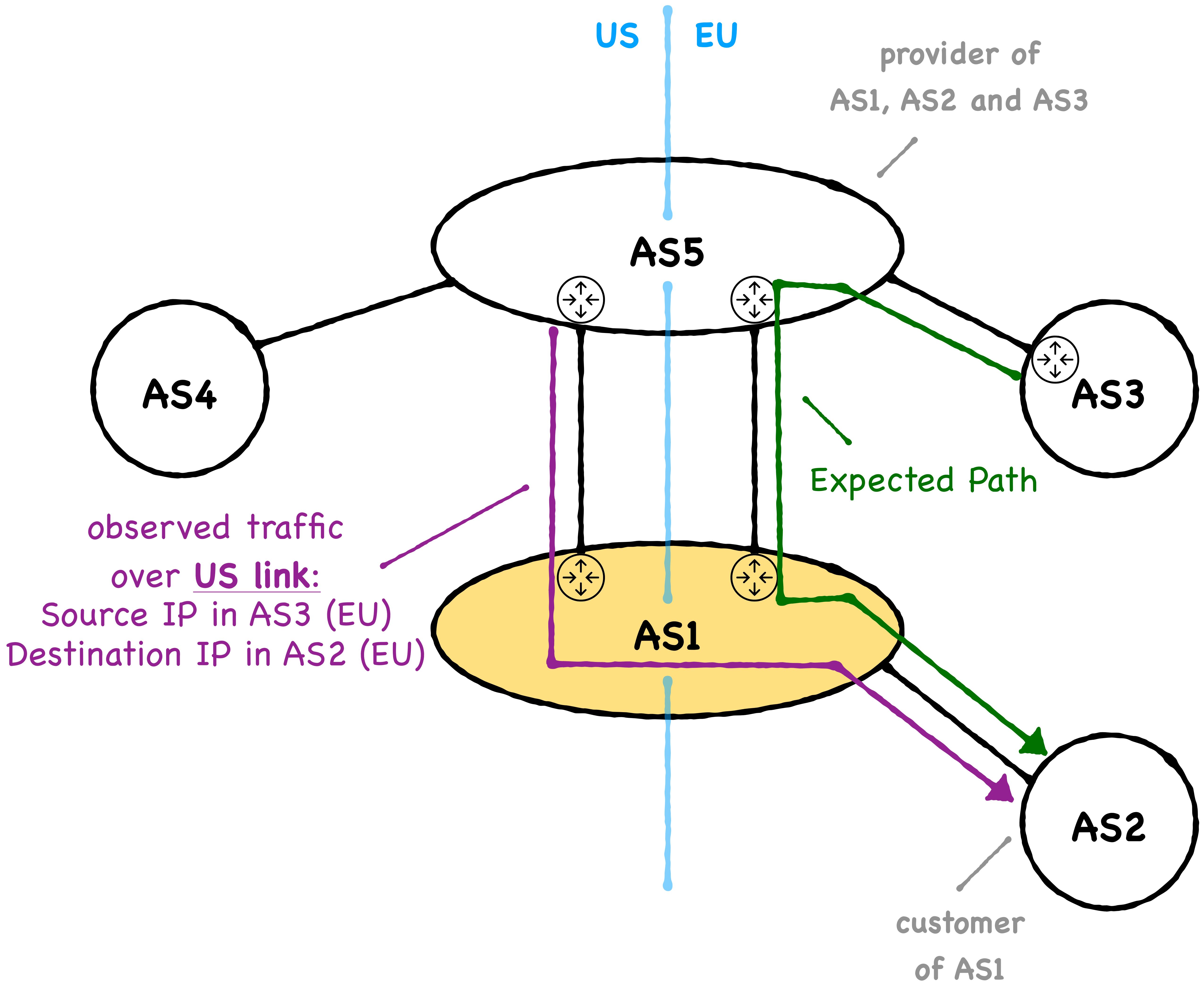


traffic forwarder
BGP path: 3 2 1 5
(not visible to AS1)

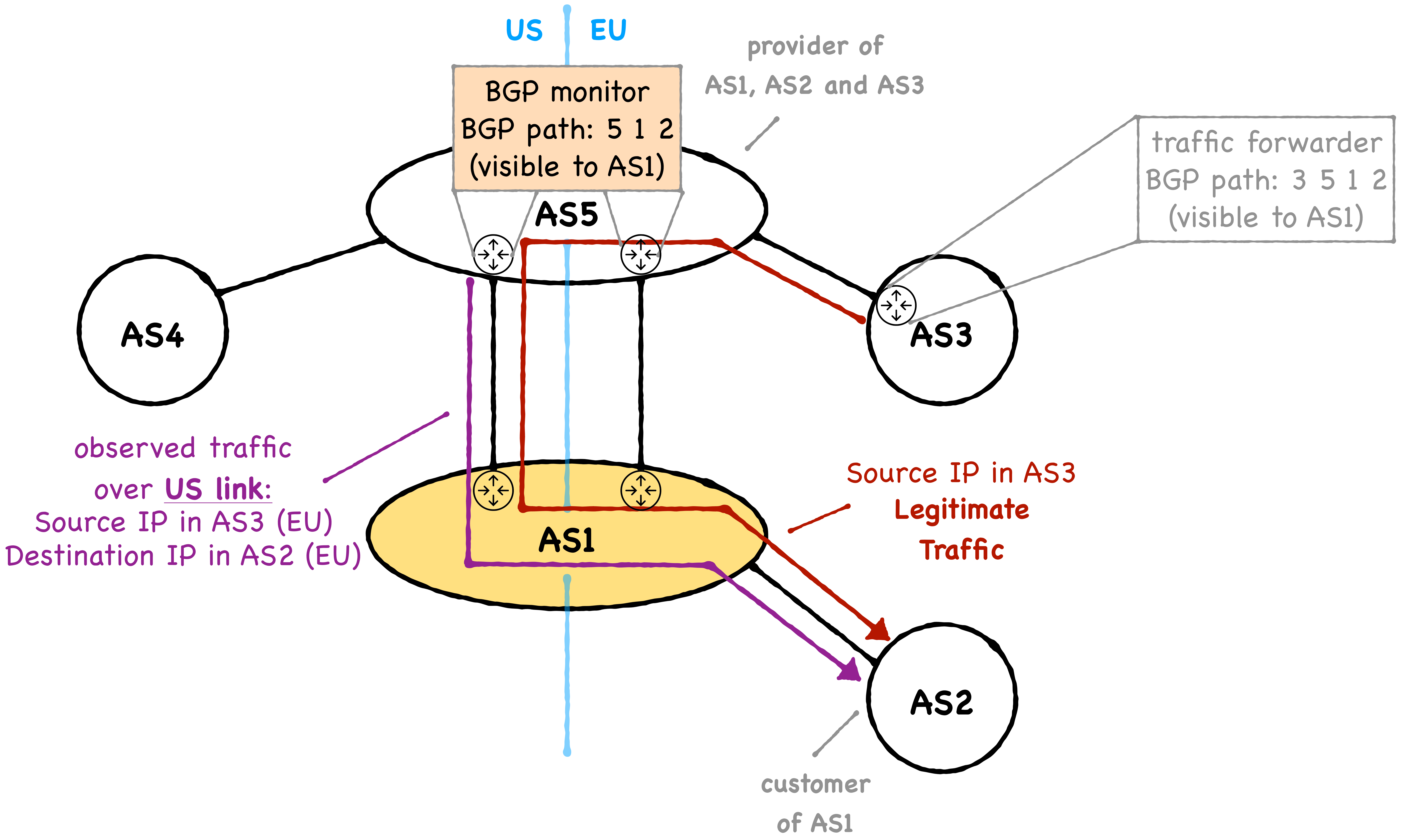
looking glass
BGP path: 3 6 … 5
(visible to AS1)

looking glass
BGP path: 2 1 5
(visible to AS1)

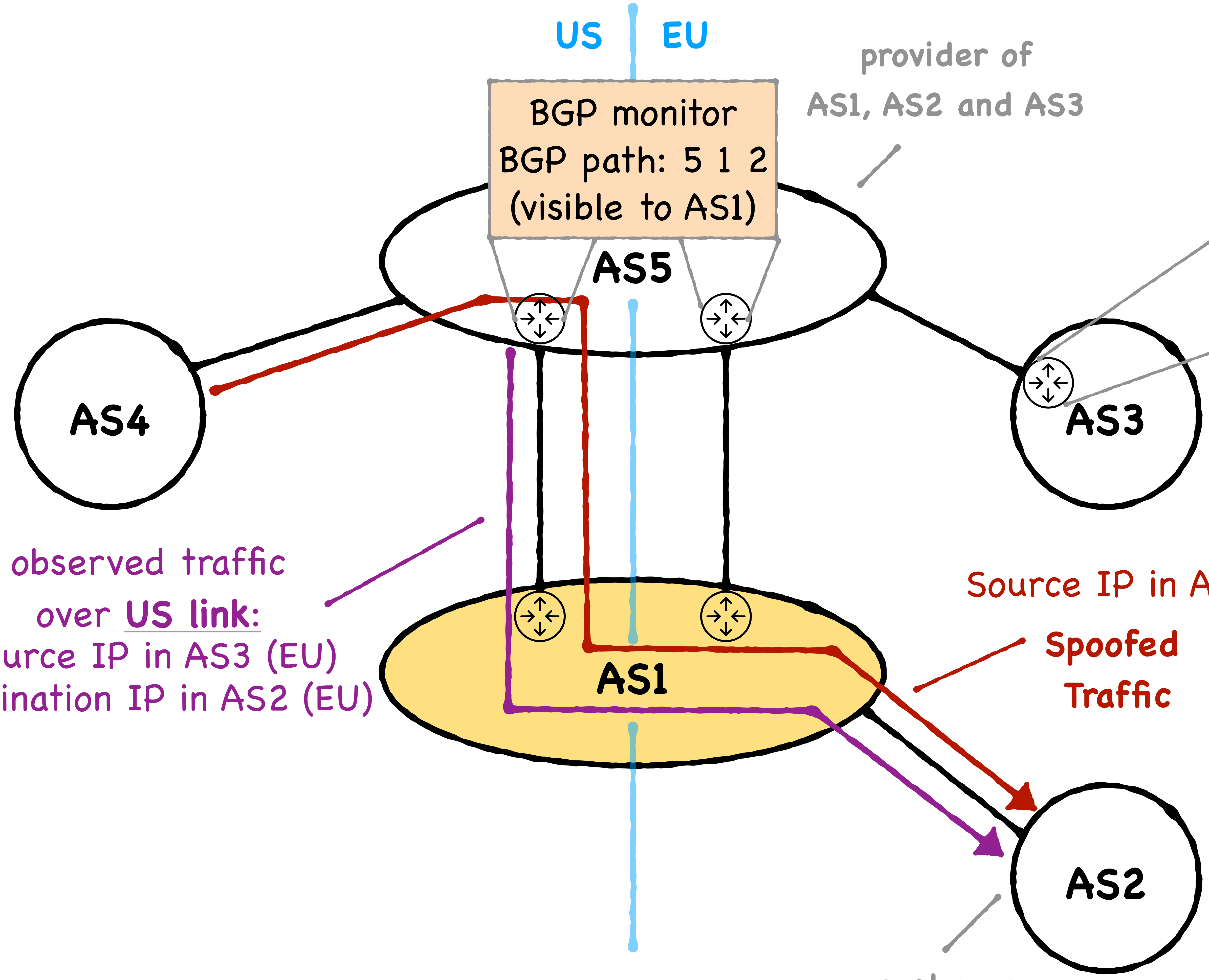
Detecting expensive transatlantic paths



Detecting expensive transatlantic paths



Detecting expensive transatlantic paths



BGP monitor
BGP path: 5 1 2
(visible to AS1)

provider of
AS1, AS2 and AS3

traffic forwarder
BGP path: 3 5 1 2
(visible to AS1)

observed traffic
over **US link**:
Source IP in AS3 (EU)
Destination IP in AS2 (EU)

Source IP in AS3
Spoofed
Traffic

customer
of AS1

Penny

Penny's statistical model

Hypotheses

H1: hypothesis that the flow under test is closed-loop

H2: hypothesis that the flow under test is not closed-loop

Parameters

p_{drop} : probability of dropping a TCP data packet

p_{noRTX} : probability miss a retransmission within a closed-loop flow

Measurement counters

n_{RTX} : # of observed retransmissions for packets we dropped

n_{noRTX} : # of packets we dropped for which we did not observe a retransmission

f_{dup} : fraction of observed packets with one or more duplicates

Probabilities

$$P(\text{H1}) = (p_{\text{noRTX}})^{n_{\text{noRTX}}}$$

$$P(\text{H2}) = (f_{\text{dup}})^{n_{\text{RTX}}}$$

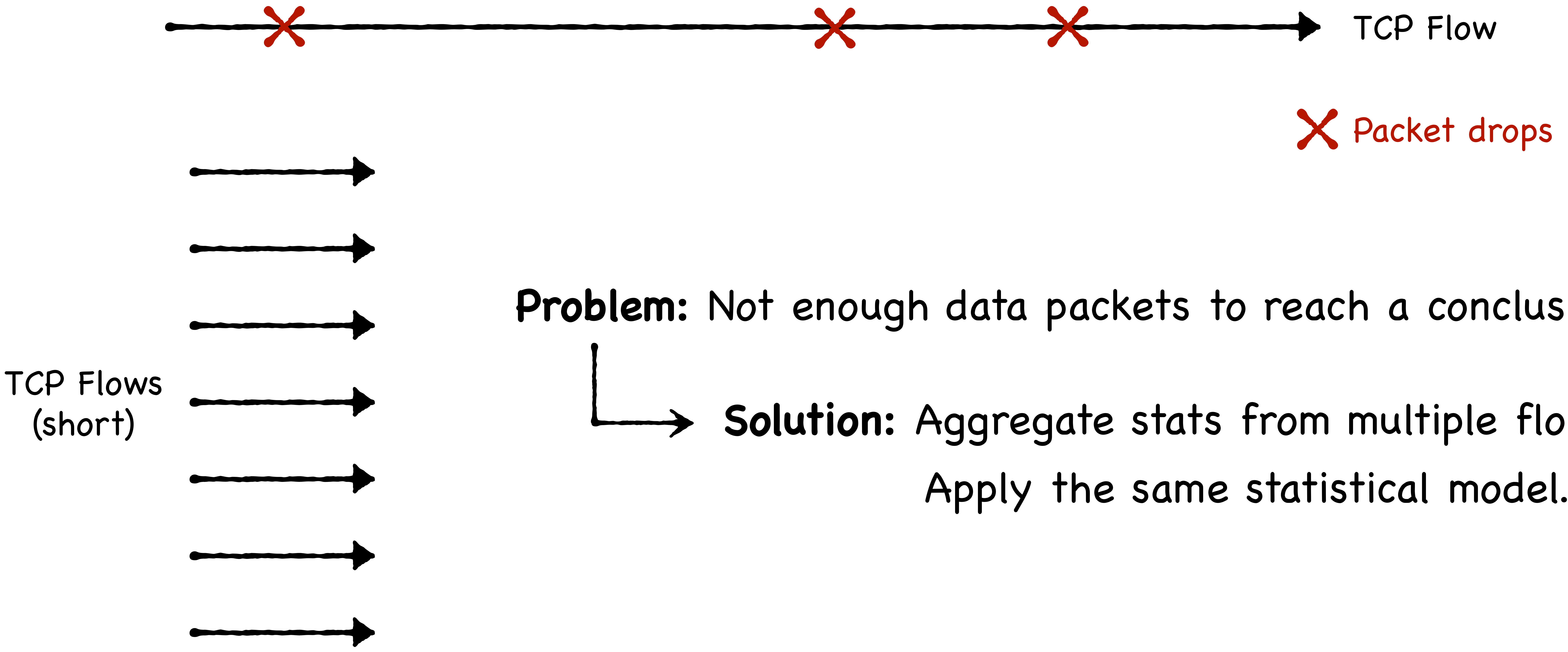
$$P(\text{genuine}) = P(H1)/(P(H1) + P(H2))$$

Procedure

$P(\text{genuine}) > 0.99 \implies$ closed-loop

$P(\text{genuine}) < 0.01$ or $f(\text{dup}) > 0.15 \implies$ not closed-loop

Dealing with short flows



Problem: Not enough data packets to reach a conclusion.

Solution: Aggregate stats from multiple flows.
Apply the same statistical model.

Penny at Runtime: The devil's in the details

Complications:

- **Dealing with the TCP protocol**
- **Dealing with network conditions**
- **Dealing with malicious sources**

Penny at Runtime: The devil's in the details

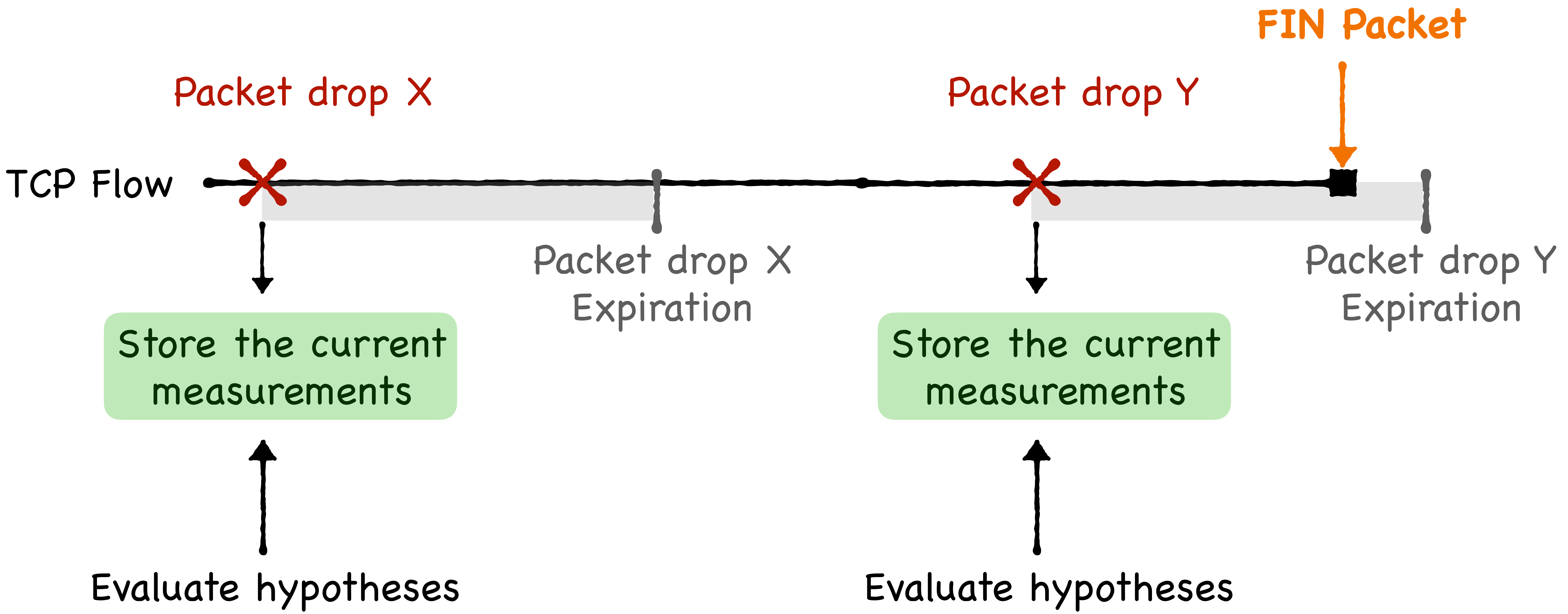
Complications:

- Dealing with the TCP protocol
- Dealing with network conditions
- Dealing with malicious sources

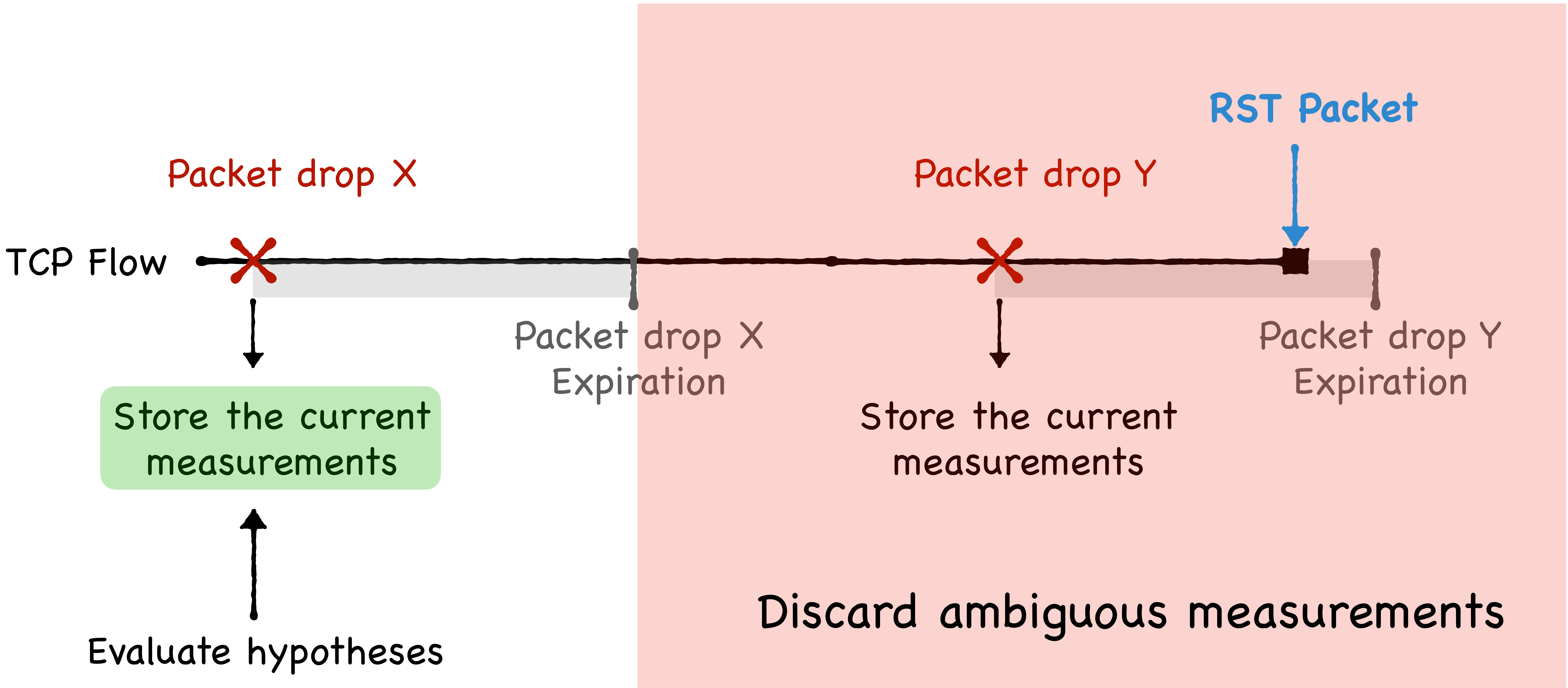
Mechanisms:

- **Selection of packets to drop**
- **Counter snapshots**
- **Conservative thresholds and parameters**

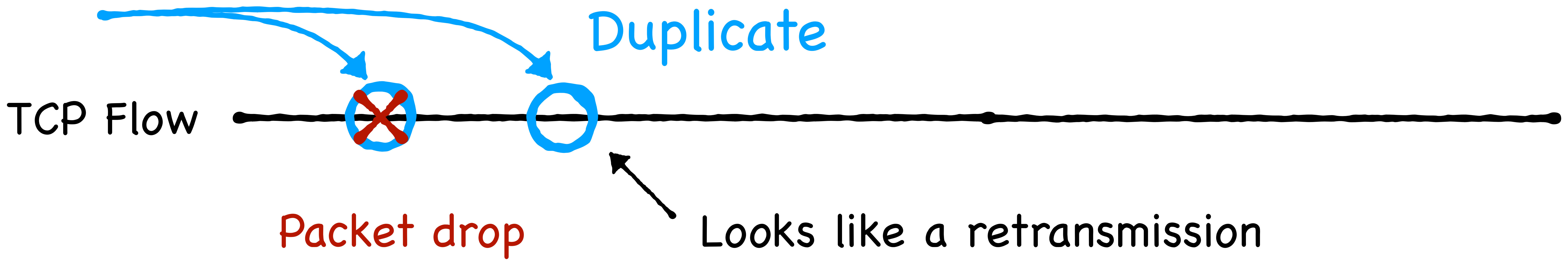
Waiting for retransmissions



Dealing with interrupted flows



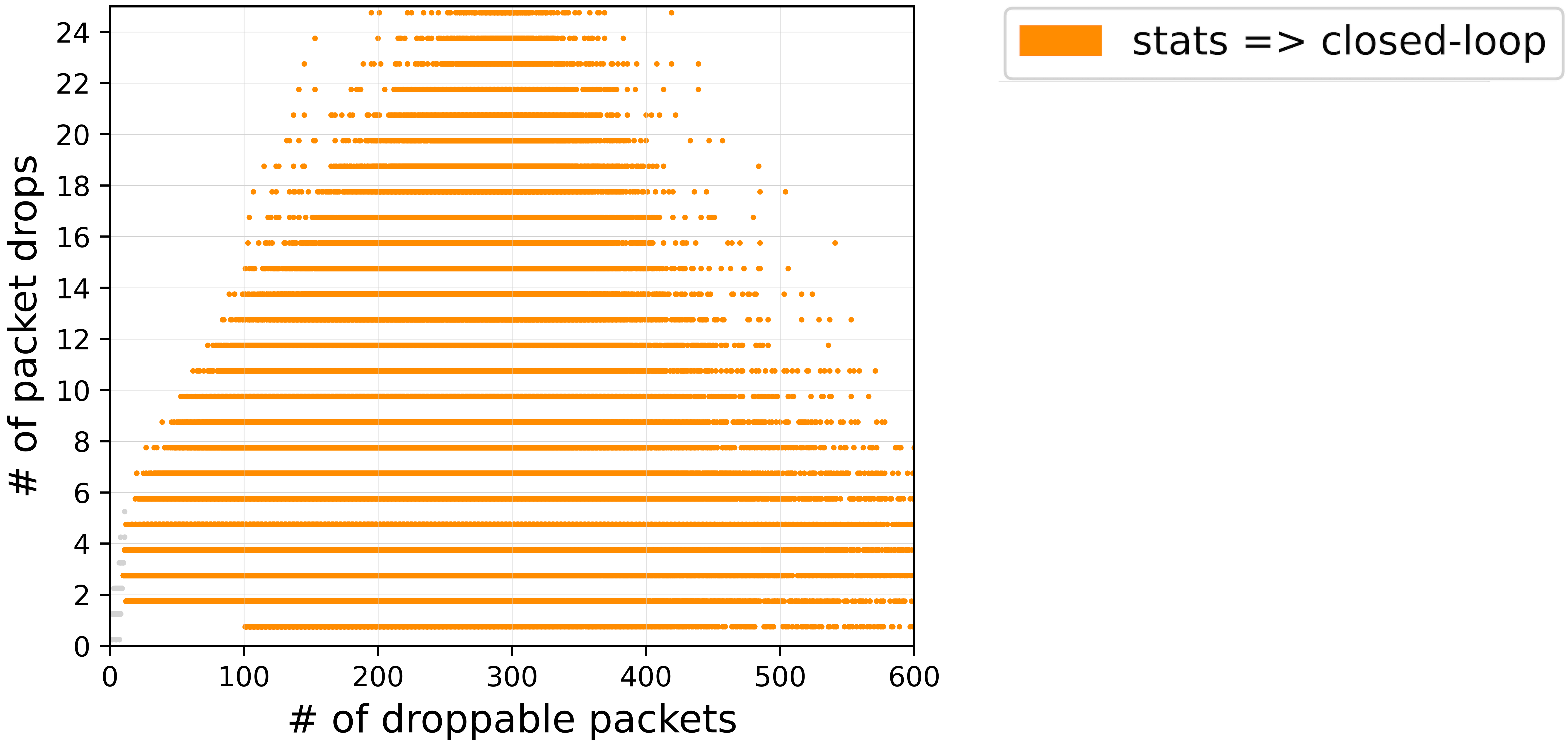
Dealing with duplicates



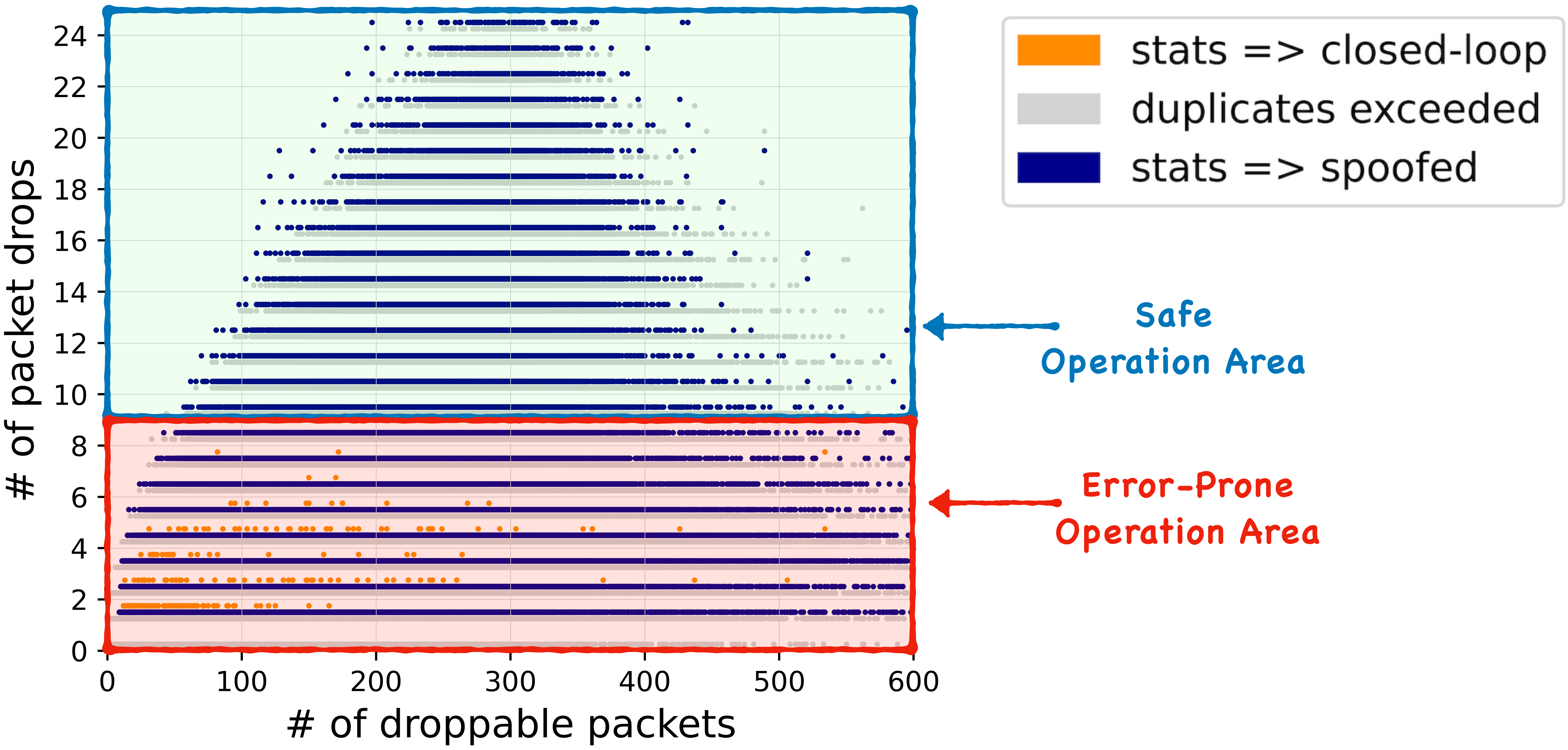
- We treat flows with 15% loss as suspicious.
- Rely on stats to cope with < 15% dups.

Evaluation

For aggregates with only closed-loop traffic, Penny's **stats** always work



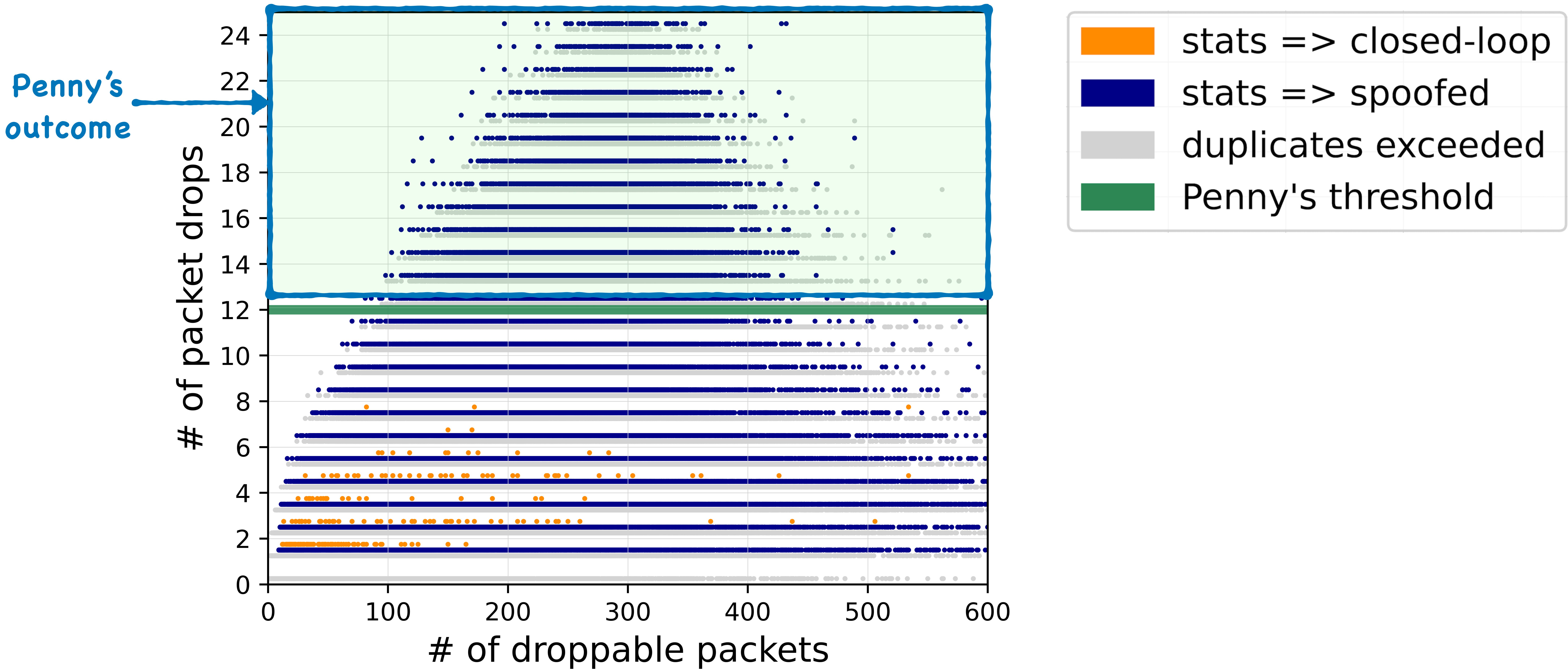
For malicious traffic, Penny's **stats** work whenever we drop enough packets



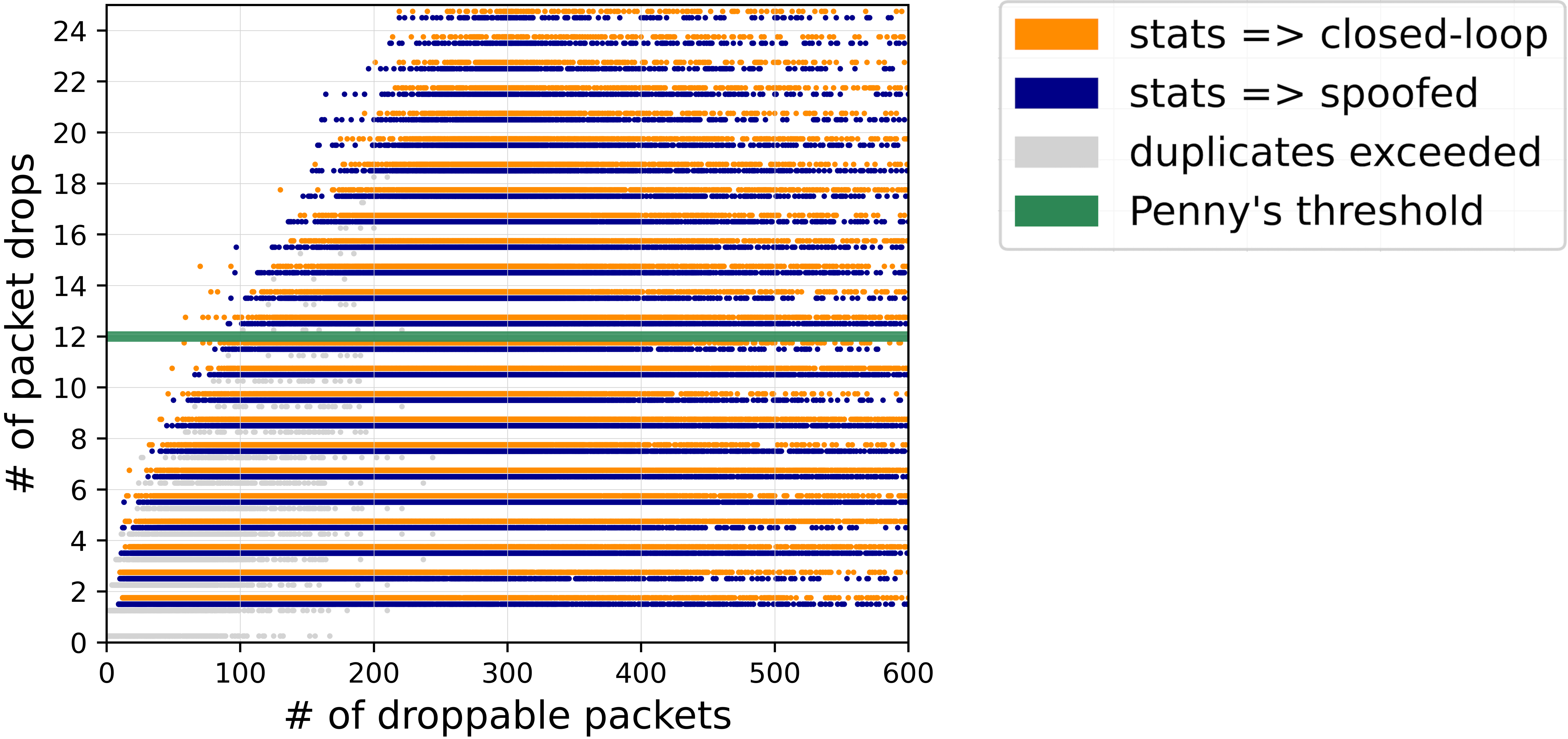
Safe Operation Area

Error-Prone Operation Area

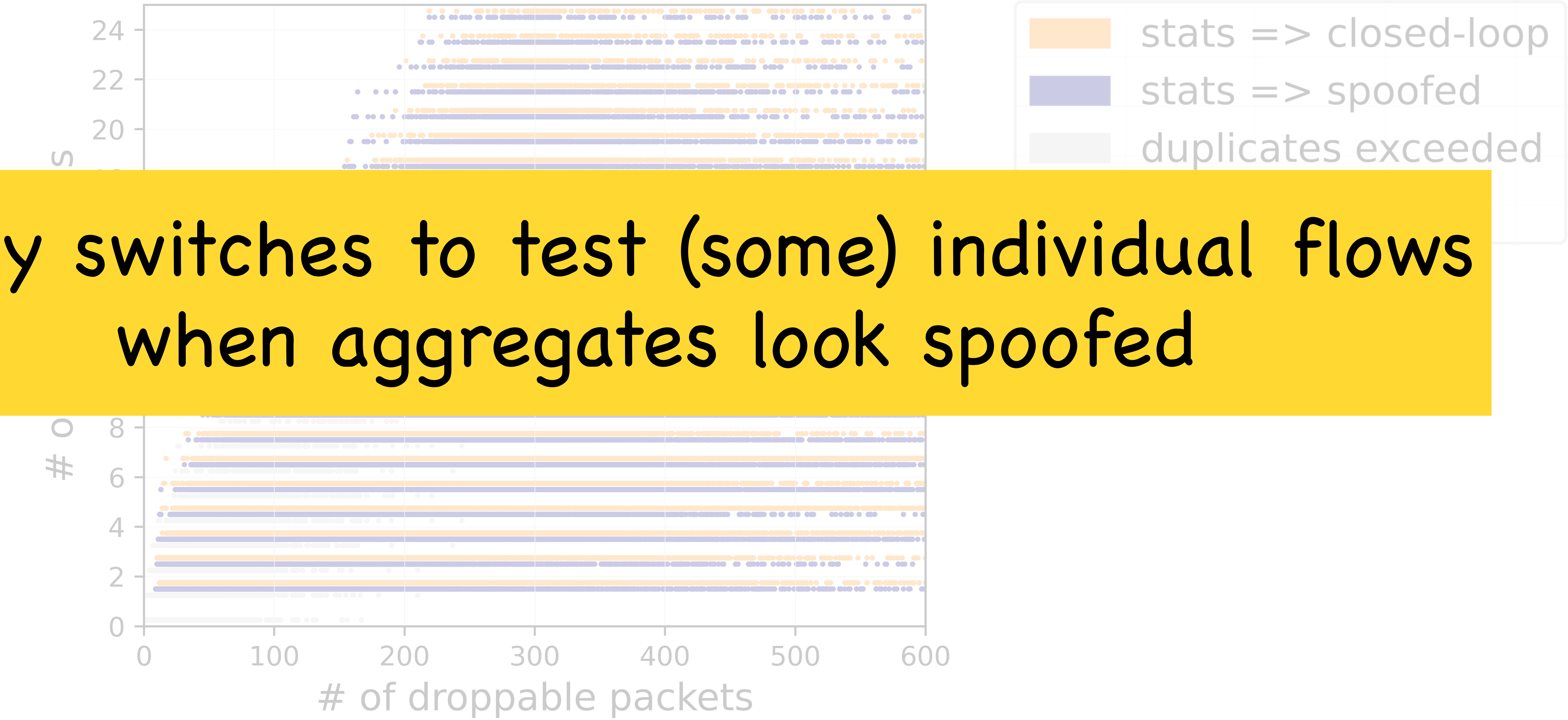
For malicious traffic, Penny is always correct as each drops at least 12 packets



For mixed traffic, Penny's **stats** do not always work



For mixed traffic, Penny's **stats** do not always work, but **Penny** does



Penny switches to test (some) individual flows when aggregates look spoofed

Other results

- Flow performance degradation is negligible.
  - We only drop 12 carefully selected packets per test.

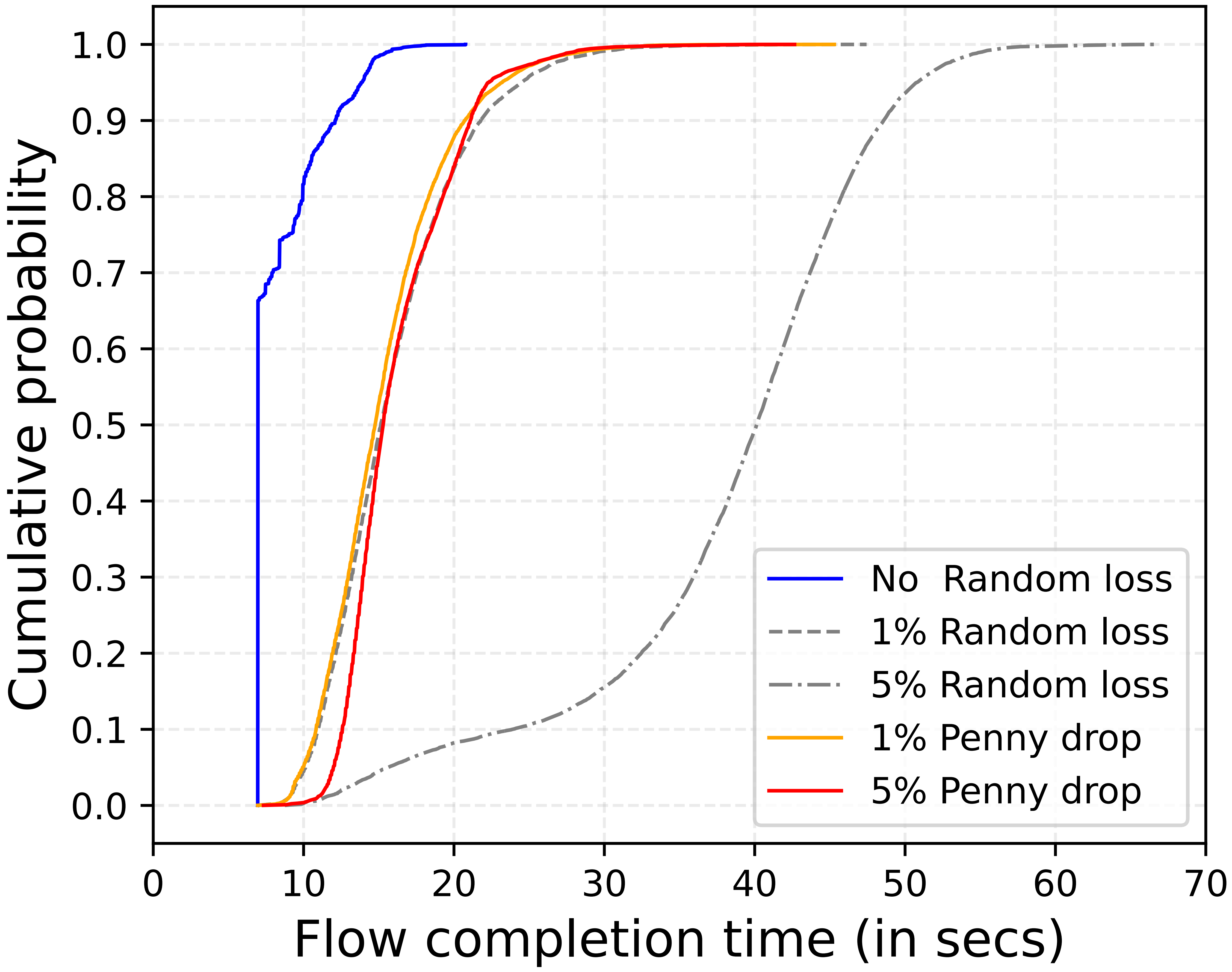
- Feasibility of system implementation.
  - Low processing requirements
  - Low memory requirements.

Penny's impact on individual flows

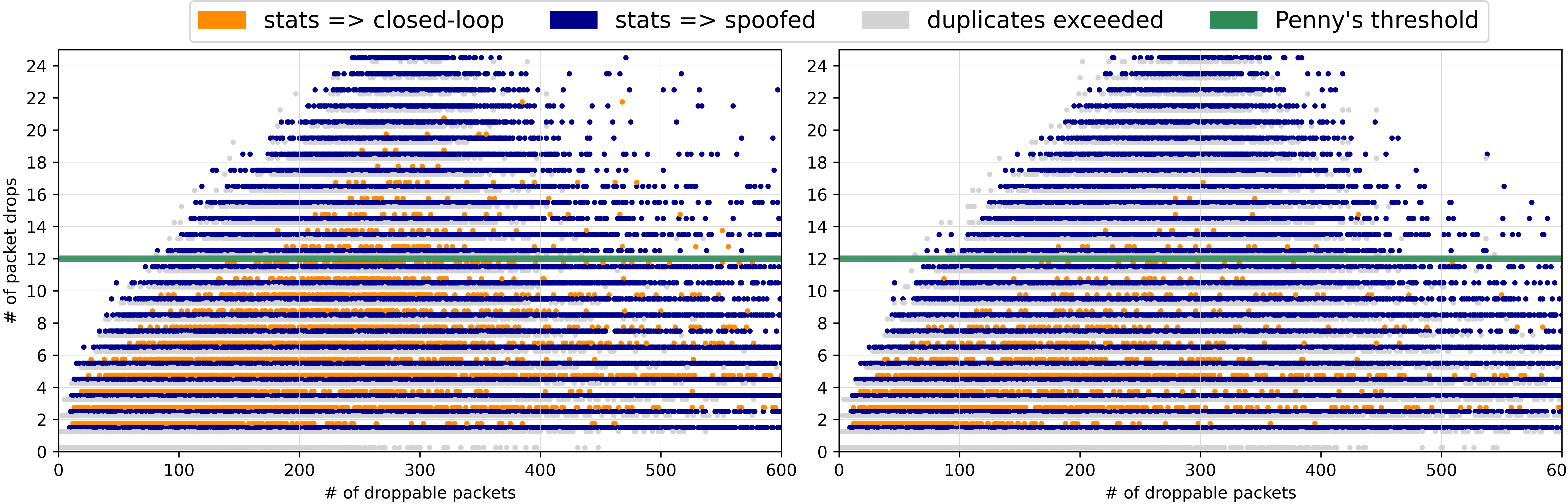
- Experiment setup:
  - TCP background traffic
  - 1 MB-long Cubic flows

- Dropping with a 5% probability (12 drops) leads to a faster conclusion and has the same impact as a 1% random loss.

- Similar results for other TCP variants.



Accuracy of **Penny's** statistical model



(a) 20% closed-loop - 80% spoofed

(b) 10% closed-loop - 90% spoofed