



BLACK HOLE DETECTION IN PACKET NETWORKS

Julian Lucek

JUNIPER
NETWORKS

Driven by
Experience™

Background

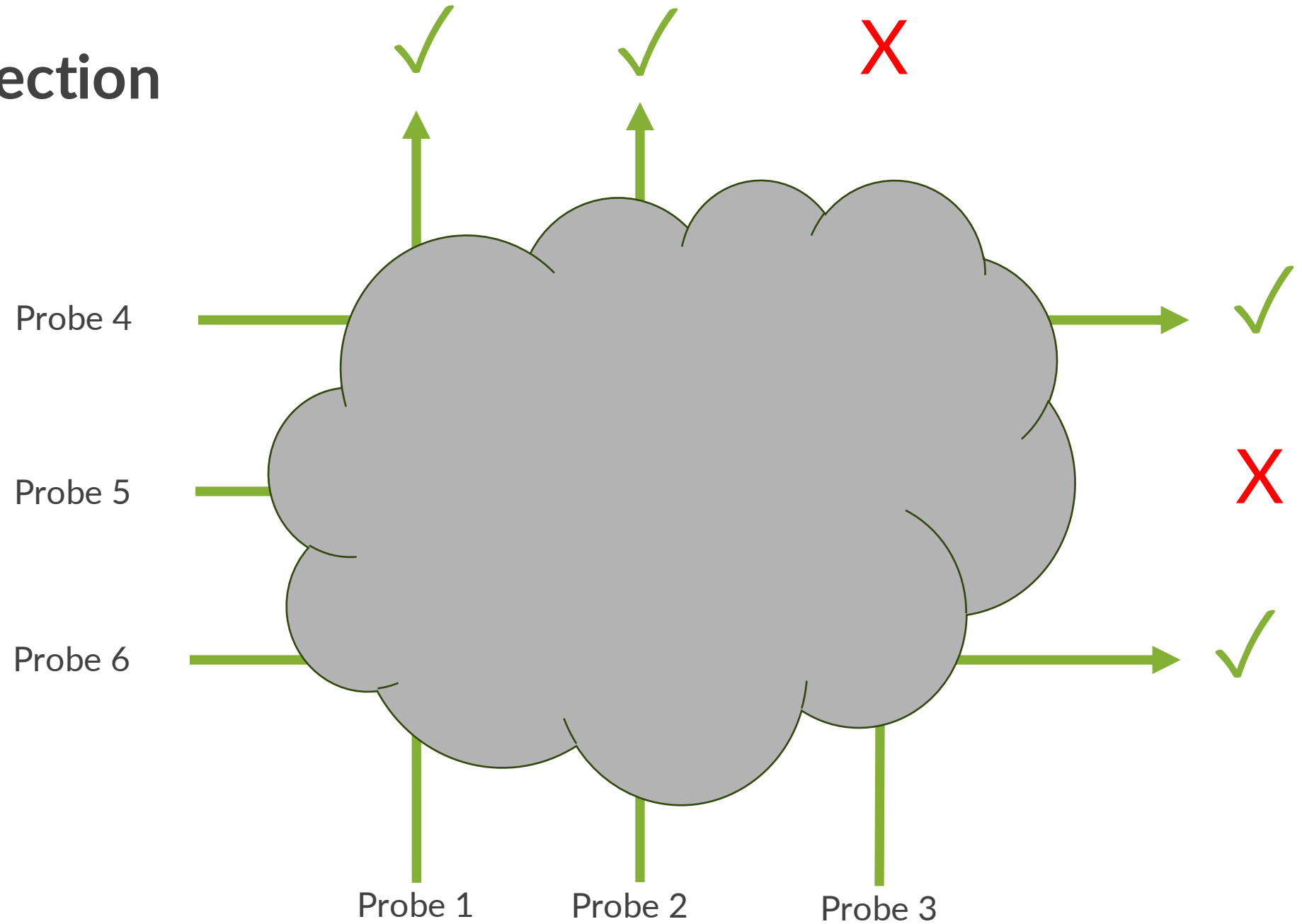
Traditionally, identifying the location and the cause of packet drops (“black holes”) in a network is quite time-consuming: “Needle in a haystack”.

Transient drops are particularly difficult to detect.

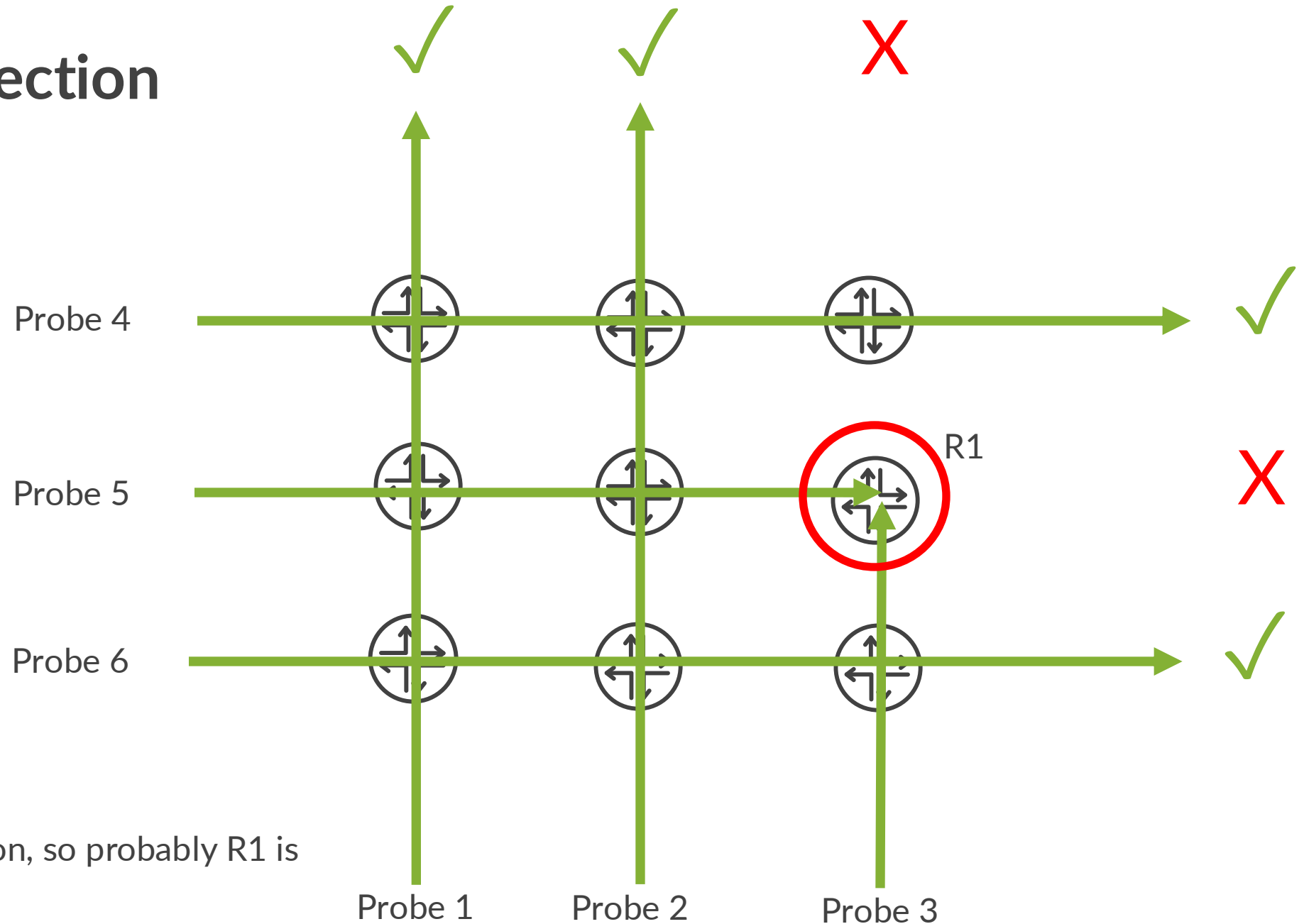
Two main approaches to detection:

- Indirect Detection
- Direct Detection

Indirect Detection



Indirect Detection



- Probe 3 is not arriving
- Probe 5 is not arriving
- They have R1 in common, so probably R1 is black-holing

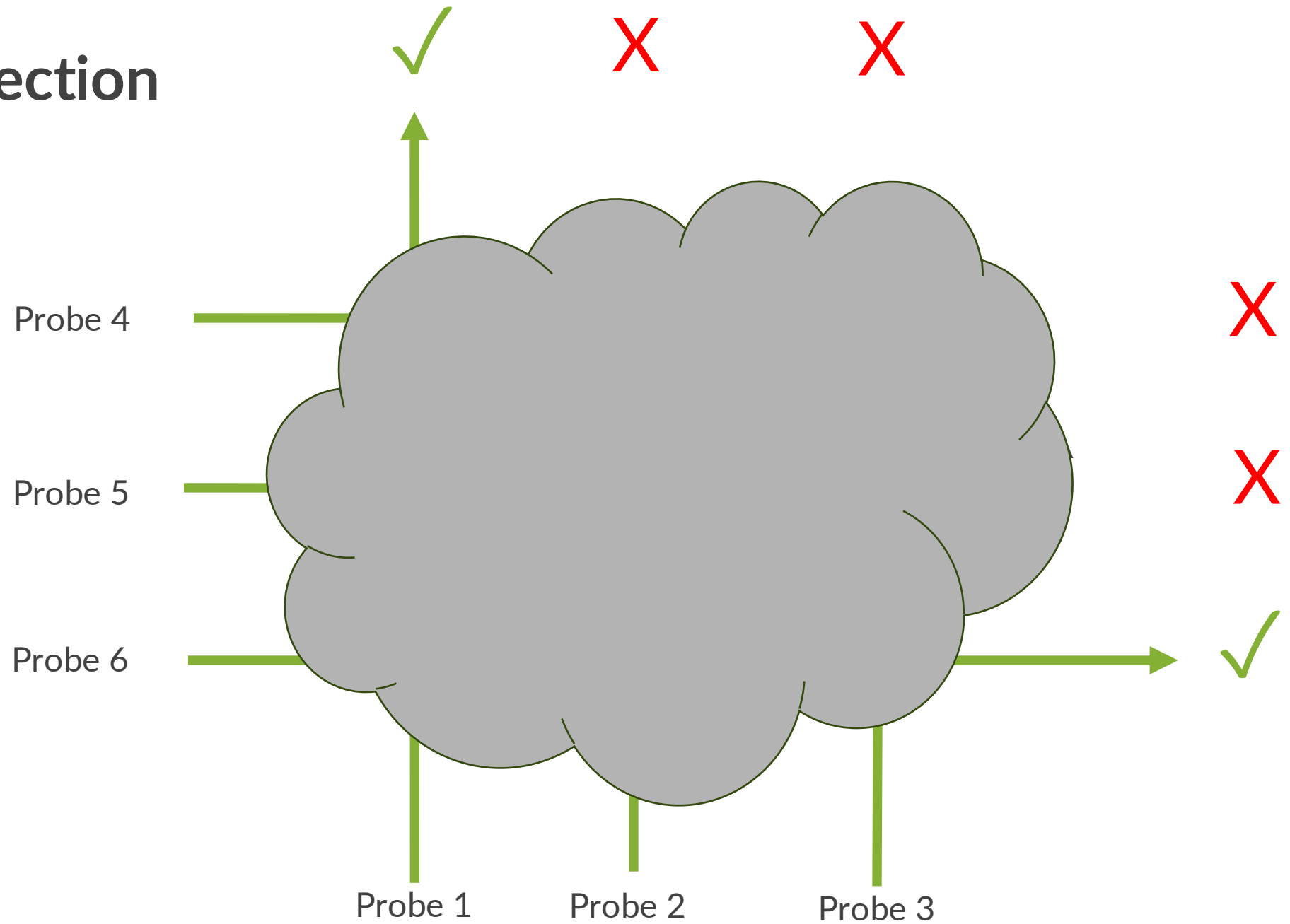
Issues with Indirect Detection

A black hole might only affect packets with particular characteristics, and the probe packets might not have those characteristics.

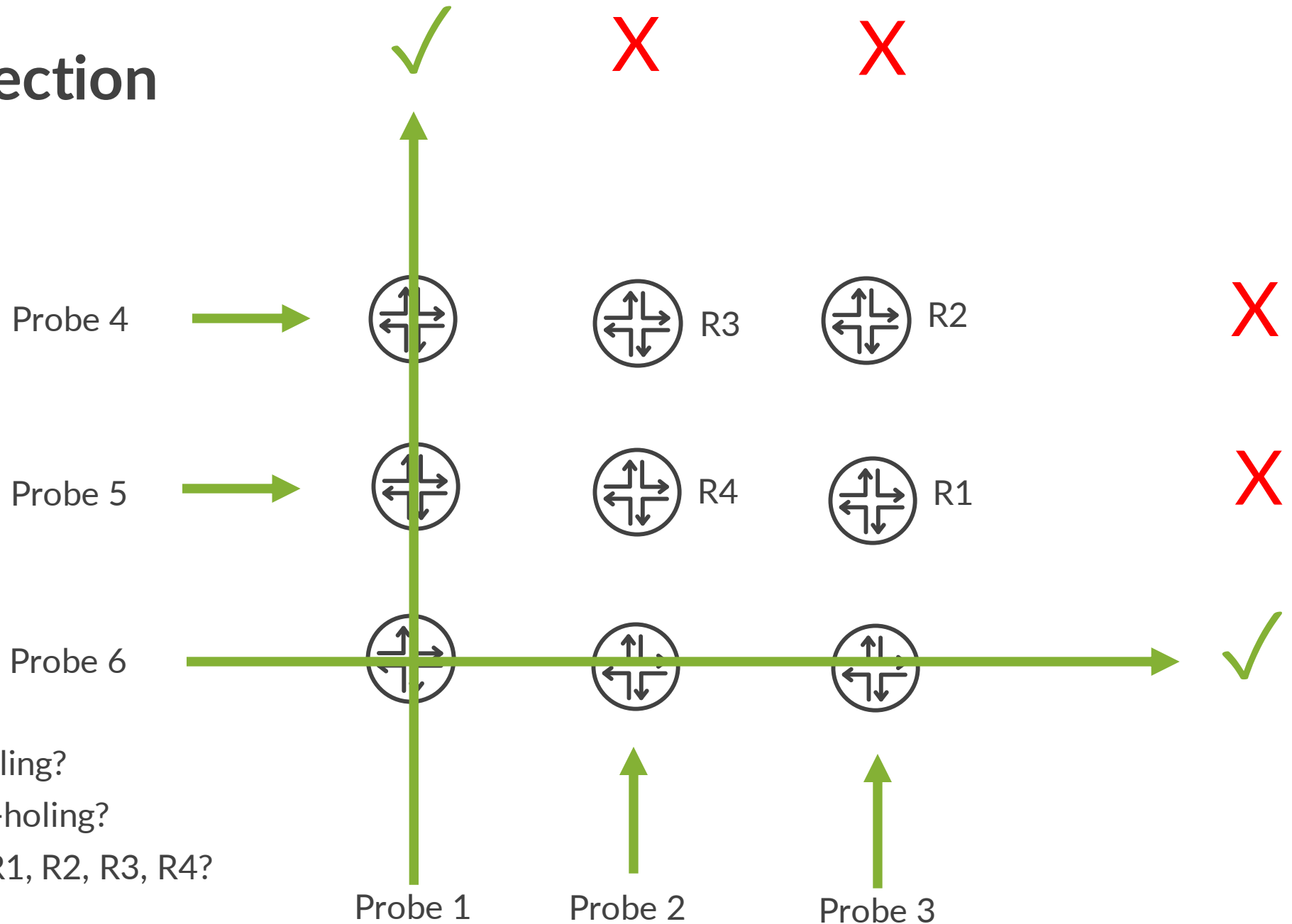
Triangulation becomes increasingly difficult in a large network

- For example, if there is more than one blackhole present...

Indirect Detection



Indirect Detection



- Are R3 and R1 black-holing?
- Or are R2 and R4 black-holing?
- Or more than 2 out of R1, R2, R3, R4?

Direct detection

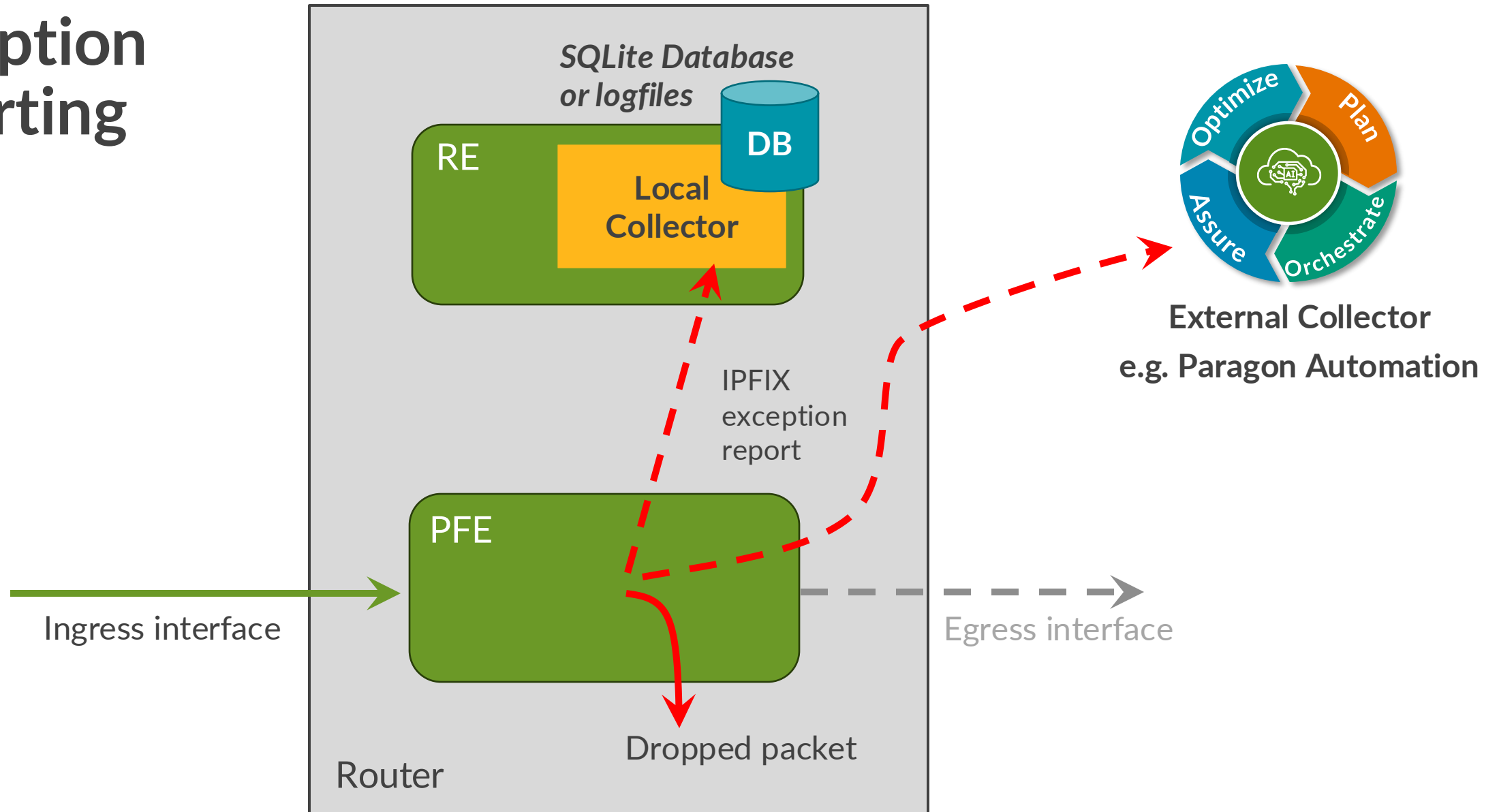
Key principle:

A router *knows* why it is dropping a packet.

So it can proactively send a report about it! (an *exception report*)

Works with any kind of packet e.g. IPv4, IPv6, MPLS, Layer 2 etc

Exception reporting



Available in Junos since 2021: “Juniper Resiliency Interface (JRI)”

Deployed in multiple production networks

Examples of forwarding exception categories

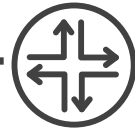
- Unknown address family
- No matching IP prefix in FIB
- No matching MPLS label in FIB
- TTL expired
- Unknown VLAN tag
- MAC learn limit exceeded
- GRE mismatch
- MTU exceeded
- Various packet errors e.g. header errors, checksum errors, length not matching declared length...

Easy to setup test

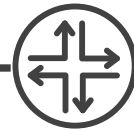
ping 192.0.2.1 **ttl 1**



R1



R2
(DUT)



R3
(192.0.2.1)

Example exception report (local collector)

***** IPFIX *****

Version: 10

Length: 106

Export time: 1695116517

Seq no: 3397

ObservationDomainID: 65536

setID: 1024

setLength: 90

field_type: exceptionCode, field_length: 2

value: TTL Expired

field_type: nhIndex, field_length: 4

value: 707

field_type: oifIndex, field_length: 4

value: 0

field_type: underlyingiifIndex, field_length: 4

value: 0

field_type: iifIndex, field_length: 4

value: 362

field_type: flowDirection, field_length: 1

value: 00

field_type: dataLinkFrameSize, field_length: 2

value: 1446

field_type: dataLinkFrameSection, field_length: 64

value: 2c6bf561 482a2c6b f5e6d72a 88470002

31014500 05948050 00000101 b4480b65

69010b00 006b0800 d64b023a 00006509

6ce50007 b6370809 0a0b0c0d 0e0f1011

↻ First N bytes of the packet (64 in this example)

IETF work: IPFIX extensions

IP Flow Information Export (IPFIX) Information Elements Extension for Forwarding Exceptions

<https://datatracker.ietf.org/doc/draft-mvmd-opsawg-ipfix-fwd-exceptions/>

```
IP Flow Information Export
Internet-Draft
Intended status: Standards Track
Expires: 22 March 2024

C. Munukutla
S. Vaid
Juniper Networks, Inc.
A. Mahale
D. Patel
Google, Inc.
19 September 2023
```

```
IP Flow Information Export (IPFIX) Information Elements Extension for
Forwarding Exceptions
draft-mvmd-opsawg-ipfix-fwd-exceptions-08
```

- Proposes new IPFIX Information Elements for reporting exceptions
- Proposes putting Forwarding Exception reason codes in the IANA registry (e.g. 2 = TTL_EXPIRY)

New IPFIX Information Elements defined in the IETF draft (highlighted in green)

***** Netflow/IPFIX *****

Version: 10

Length: 106

Export time: 1695116517

Seq no: 3397

ObservationDomainID: 65536

setID: 1024

setLength: 90

field_type: exceptionCode, field_length: 2

value: TTL Expired

field_type: nhIndex, field_length: 4

value: 707

field_type: oifIndex, field_length: 4

value: 0

field_type: underlyingiifIndex, field_length: 4

value: 0

field_type: iifIndex, field_length: 4

value: 362

field_type: flowDirection, field_length: 1

value: 00

field_type: dataLinkFrameSize, field_length: 2

value: 1446

field_type: dataLinkFrameSection, field_length: 64

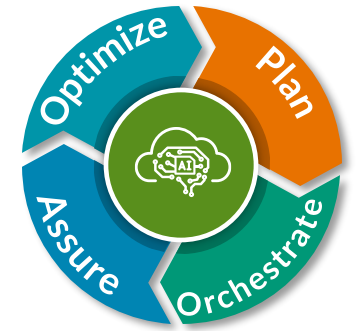
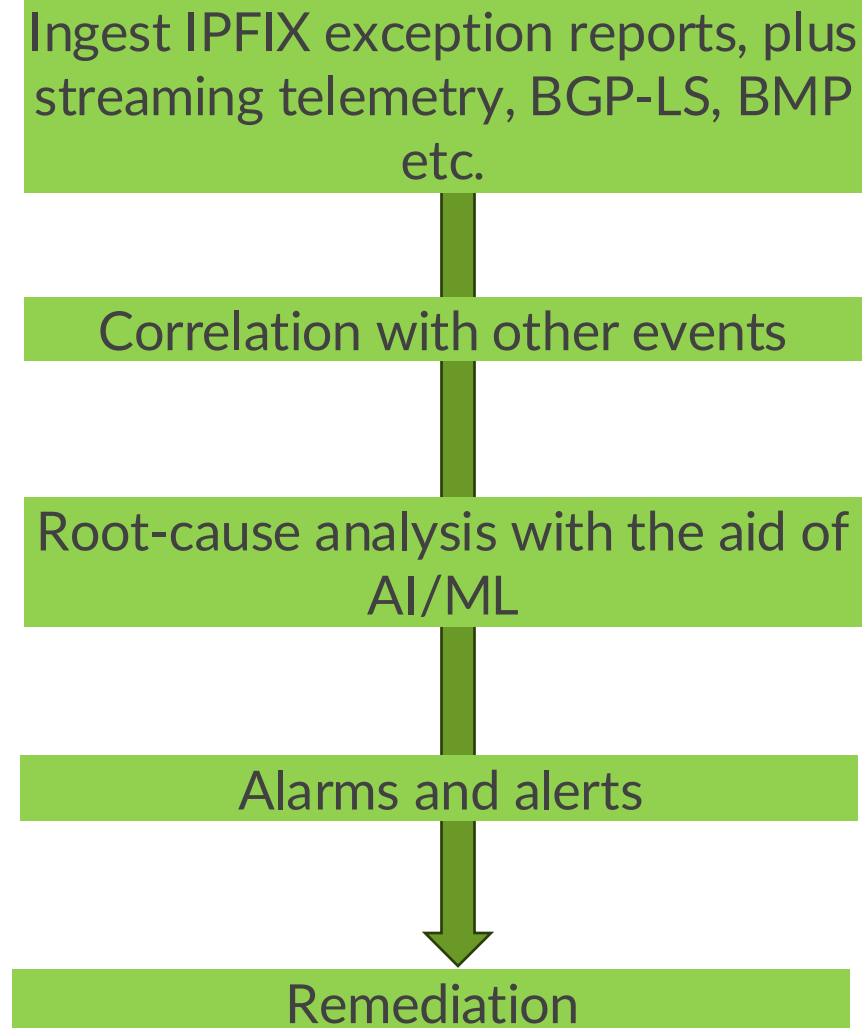
value: 2c6bf561 482a2c6b f5e6d72a 88470002

31014500 05948050 00000101 b4480b65

69010b00 006b0800 d64b023a 00006509

6ce50007 b6370809 0a0b0c0d 0e0f1011

The role of the external collector



Paragon Automation

Further information

- Juniper Techpubs link:
 - <https://www.juniper.net/documentation/us/en/software/junos/flow-monitoring/topics/topic-map/resiliency-exception-reporting.html>
- Blogs:
 - <https://community.juniper.net/blogs/julian-lucek/2024/01/02/detection-of-blackholes-in-networks-using-jri>
 - <https://community.juniper.net/blogs/anton-elita/2024/01/08/packets-lost-in-transit>



THANK YOU

JUNIPER
NETWORKS

Driven by
Experience™