Differential Testing of Bundle Protocol v7 Implementations: A Preliminary Report

Stephan Havermans, Lars Baumgärtner, Marcus Wallum & Juan Caballero

November 5, 2025

Solar System Internet

ESA MOONLIGHT

CONNECTIVITY AND NAVIGATION ON THE MOON

The European Space Agency's (ESA) Moonlight programme aims to lead Europe in enabling connectivity from the lunar surface and to the Earth, serving the needs of future missions by creating a constellation of satellites for satellite communications around the Moon. The Moonlight programme works with European and Canadian industry to create a shared service in telecommunications and navigation, which is predicted to contribute to the activation of a €100 billion lunar economy.

NASA News & Events

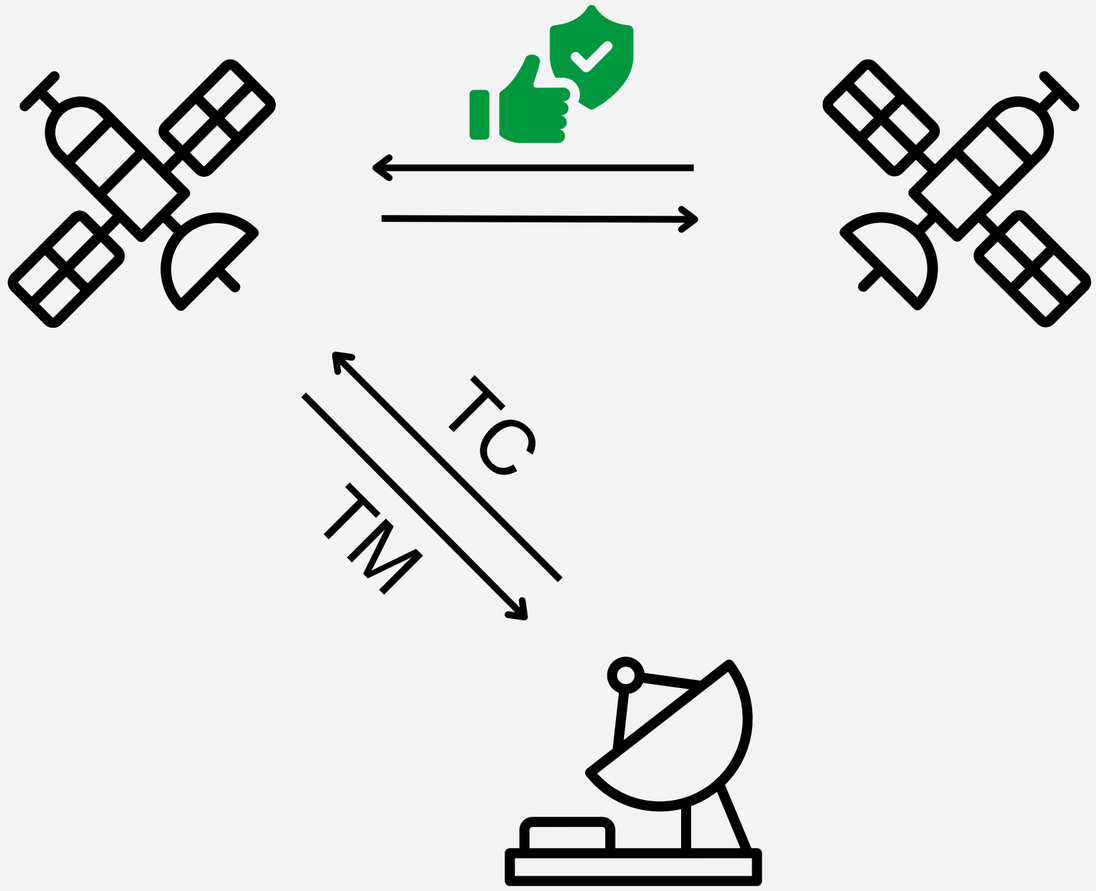
Disruption Tolerant Networking to Demonstrate Internet in Space

NASA

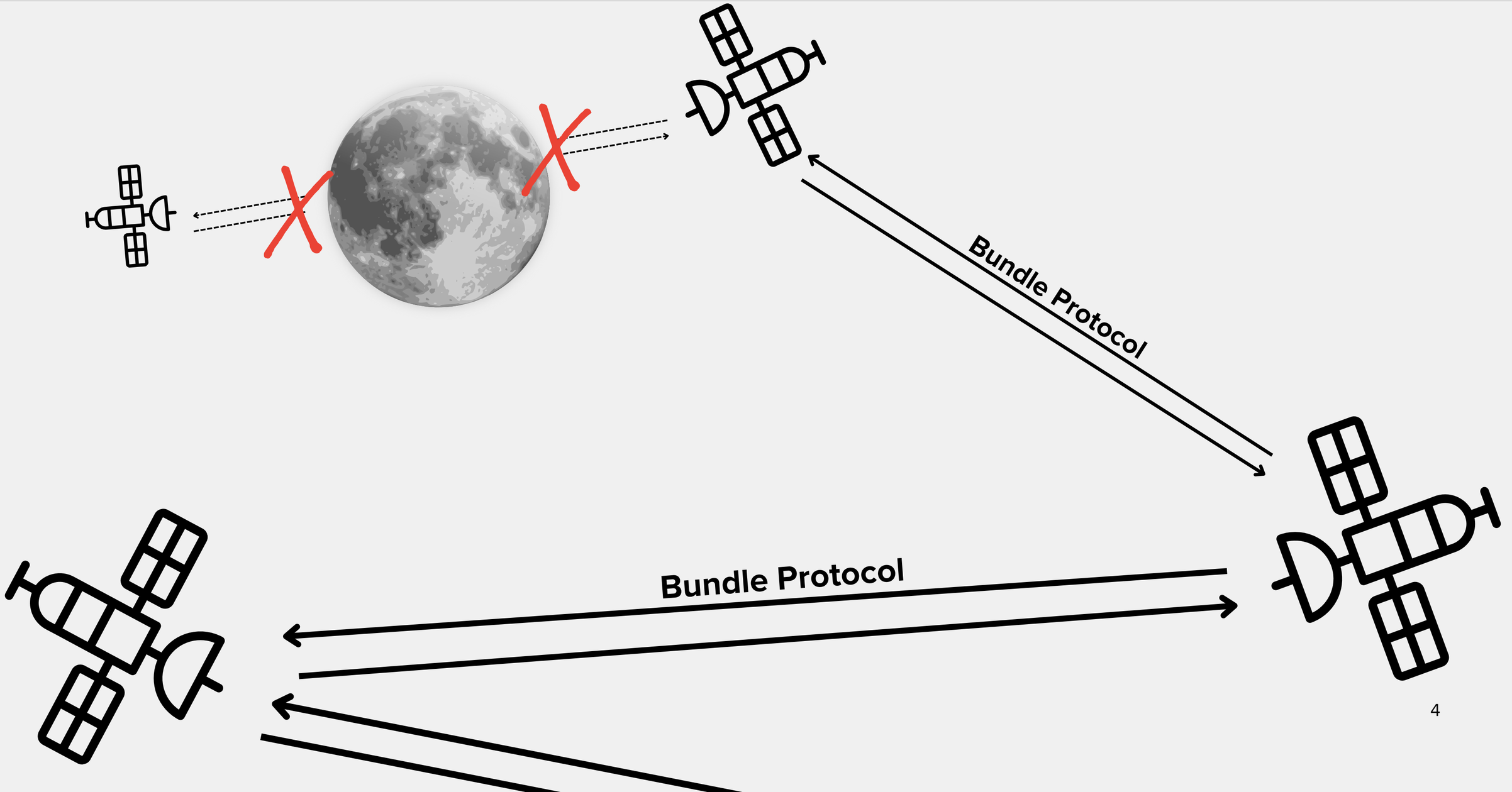
LunaNet: Empowering Artemis with Communications and Navigation Interoperability

Space Communications

- **Robust communication** is critical for space missions



Space Communications



Protocol Testing

Security

- Detect crashes & vulnerabilities

Conformance

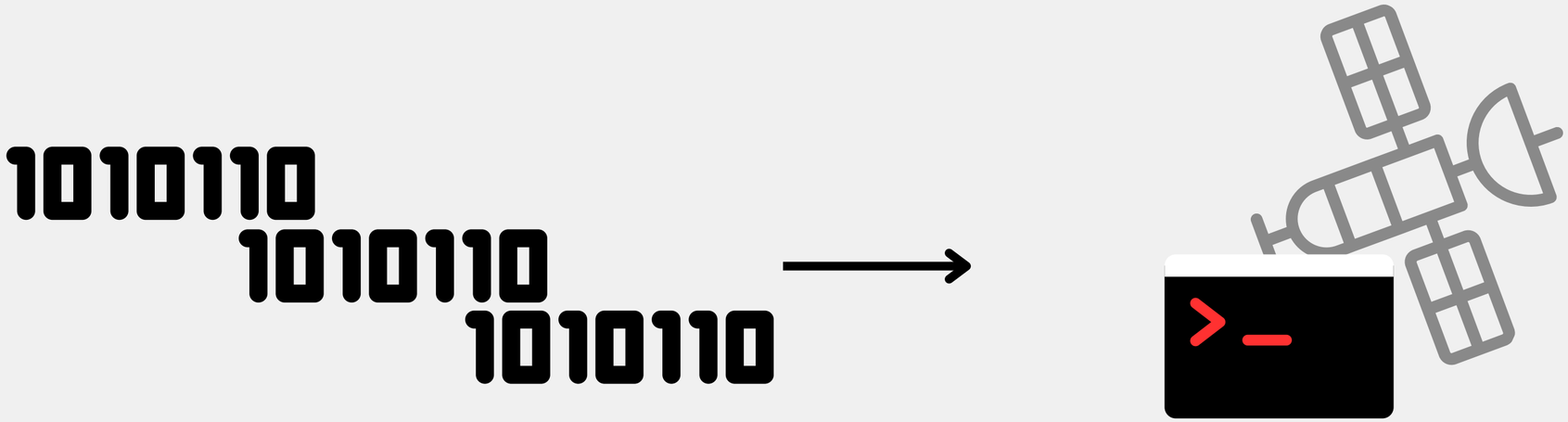
- Check against specification

Security Testing (Fuzzing)

1. Generate many inputs

2. Feed inputs to implementation

3. Observe crashes



Conformance Testing

Specification ⟶ </> Implementation

- **Today: mostly manual (e.g. PICS list)**
  - Time-consuming, requirements missed

A differential testing approach for conformance testing

Related Work

- **Formal methods (Dahbura et al., 2002; Bishop et al., 2005)**
  - Create formal models of protocol
- **Model-based differential testing (Song et al., 2015; Sivakorn et al., 2017)**
  - Symbolic execution of target implementations
- **Differential Fuzzing (Chen et al., 2016; Petsios et al., 2017; Yang et al., 2021)**
  - Generate concrete inputs

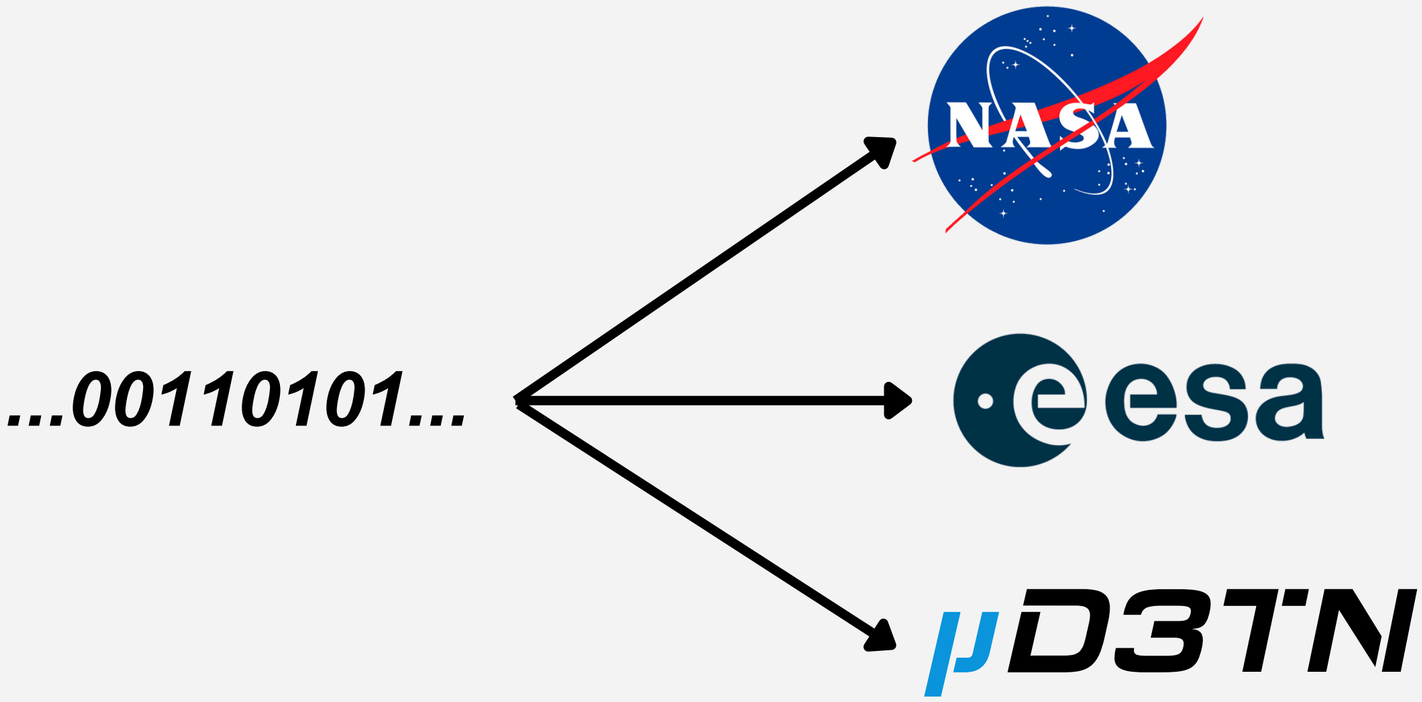
Differential Fuzzing

Deviations = Inputs accepted by some, rejected by others

...00110101...

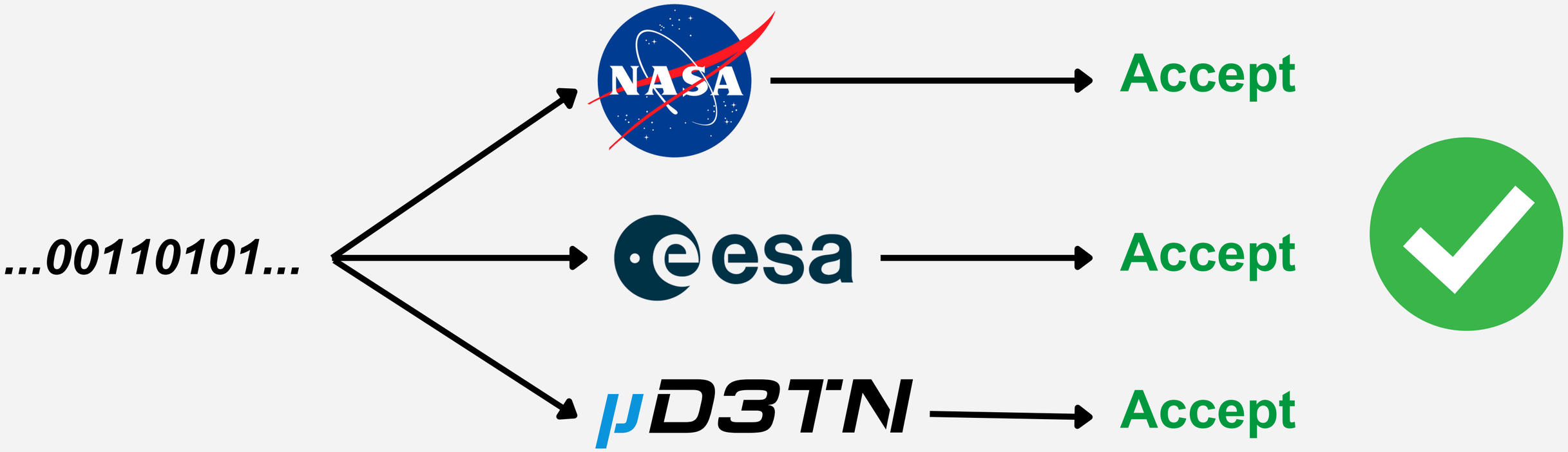
Differential Fuzzing

Deviations = Inputs accepted by some, rejected by others



Differential Fuzzing

Deviations = Inputs accepted by some, rejected by others



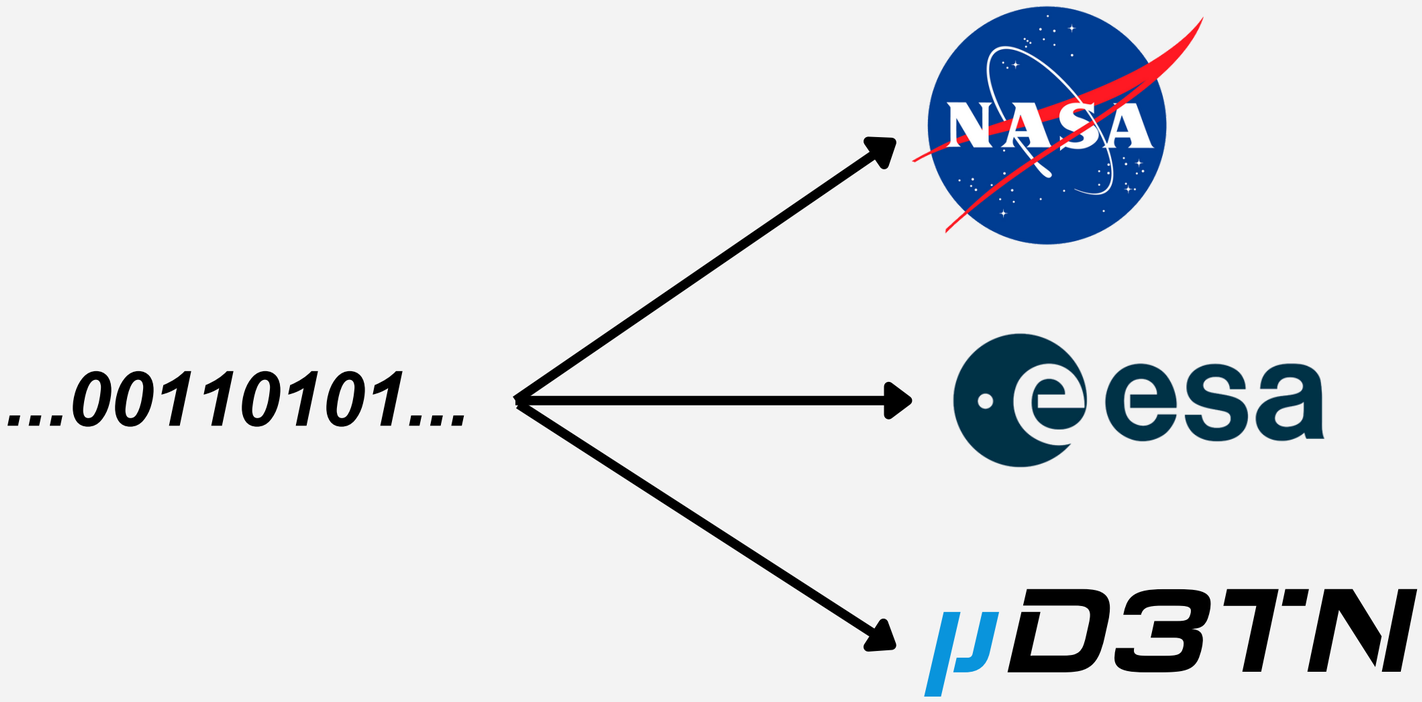
Differential Fuzzing

Deviations = Inputs accepted by some, rejected by others

...00110101...

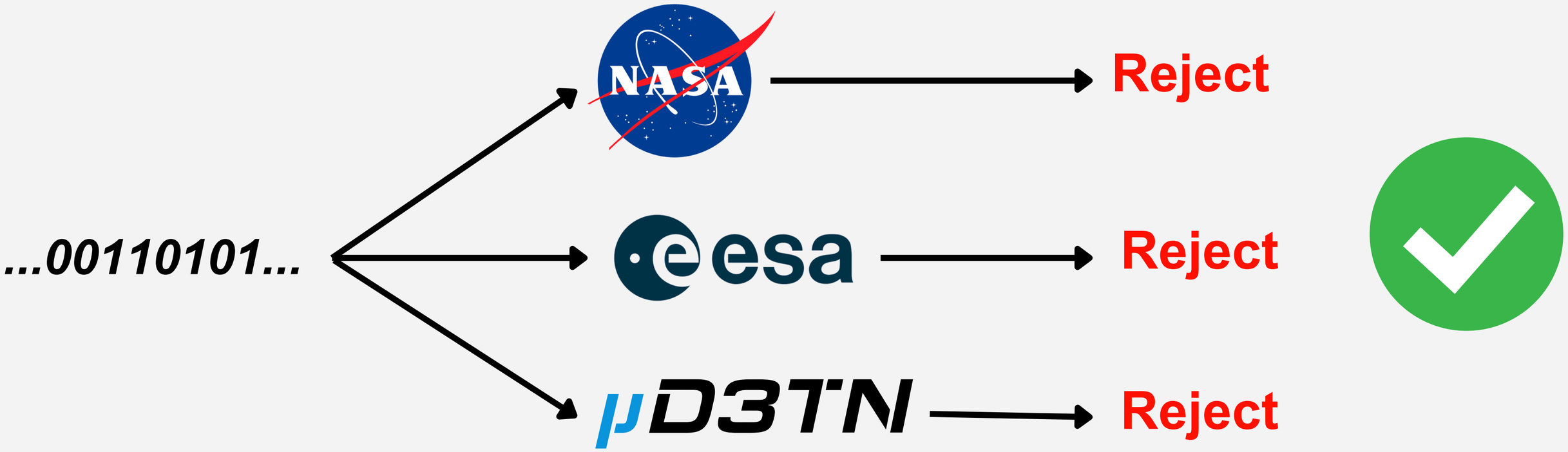
Differential Fuzzing

Deviations = Inputs accepted by some, rejected by others



Differential Fuzzing

Deviations = Inputs accepted by some, rejected by others



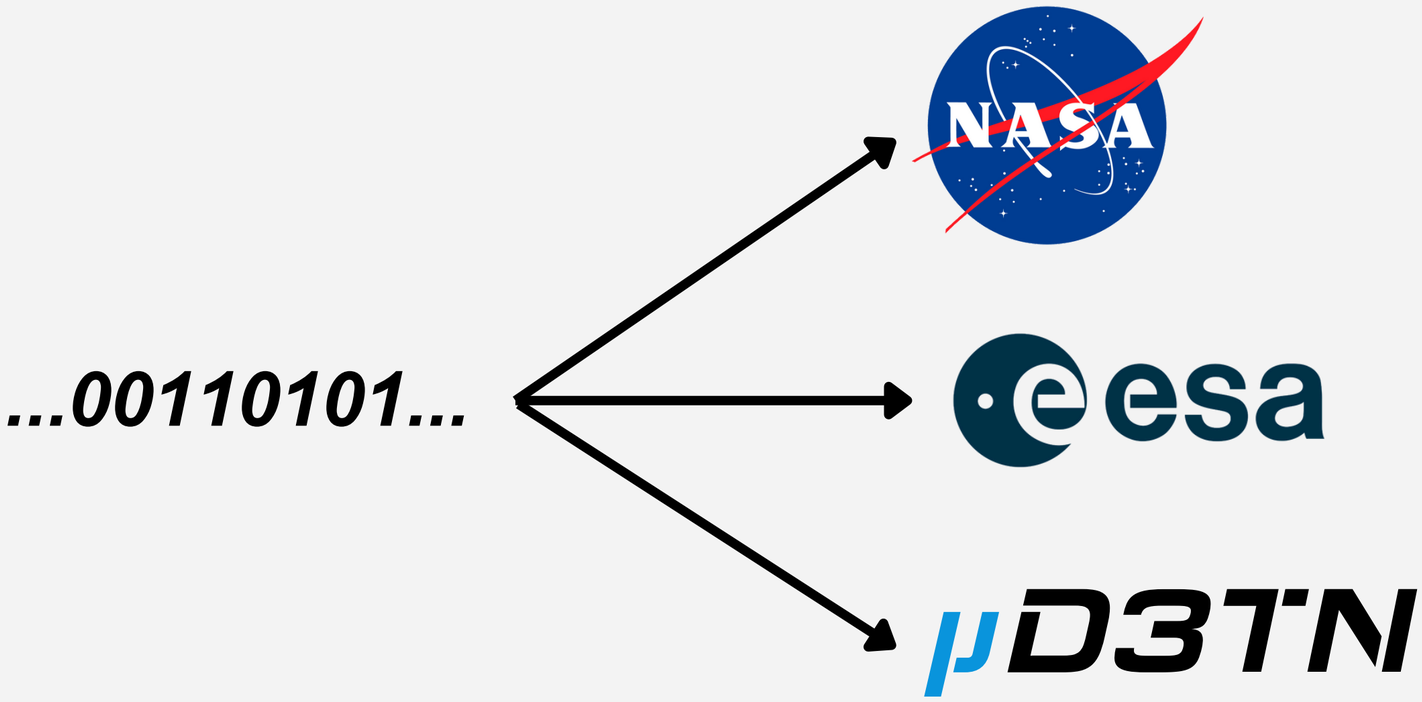
Differential Fuzzing

Deviations = Inputs accepted by some, rejected by others

...00110101...

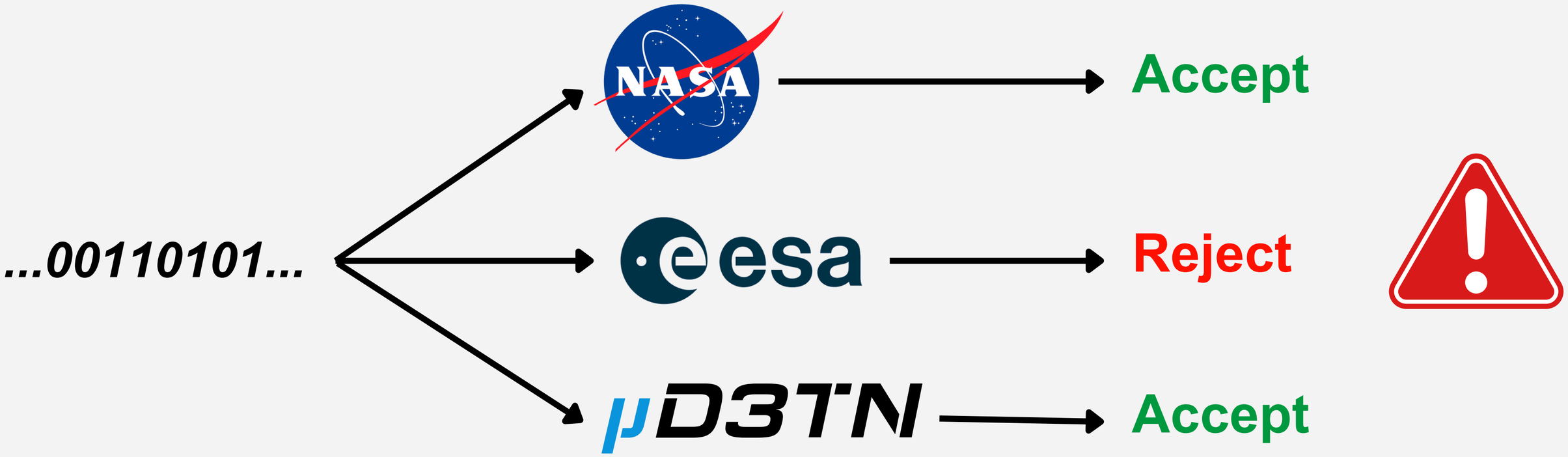
Differential Fuzzing

Deviations = Inputs accepted by some, rejected by others



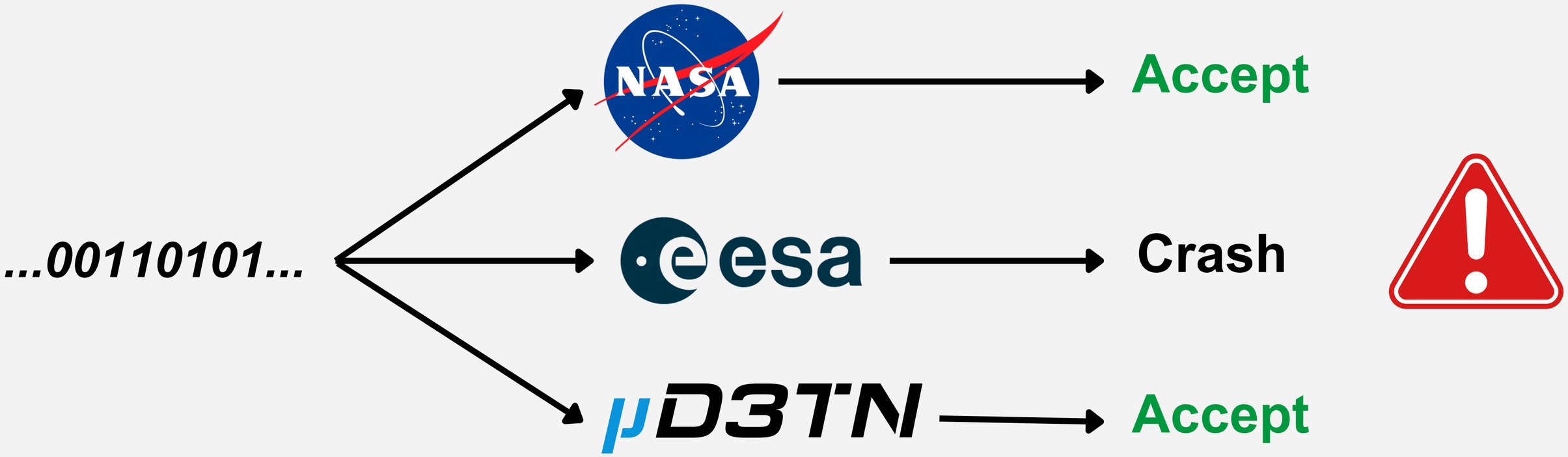
Differential Fuzzing

Deviations = Inputs accepted by some, rejected by others



Differential Fuzzing

Deviations = Inputs accepted by some, rejected by others



Bundle Protocol v7

Binary, **CBOR** encoding, **checksums**, **no** default **response**

PDU of Bundle Protocol is a ‘**Bundle**’

Part of recent CCSDS **interoperability testing**

Two specifications: **CCSDS & RFC**

Now is the time to test: **before real deployment**

Target Selection

Program	Org.	Lang.	Public	Version
ESA BP	ESA	Java	✗	448f180
µD3TN	D3TN	C	✓	0.14.2
bp7-rs	OSS	Rust	✓	0.10.7

- Three real-world BPv7 implementations – diverse in origin and programming language

Scope

- No default response available
- Stability

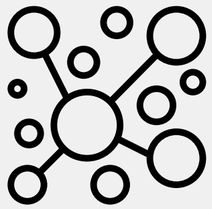
Our Approach

At least **2 implementations** of the **same protocol**

Differential Testing: Same inputs, compare outputs

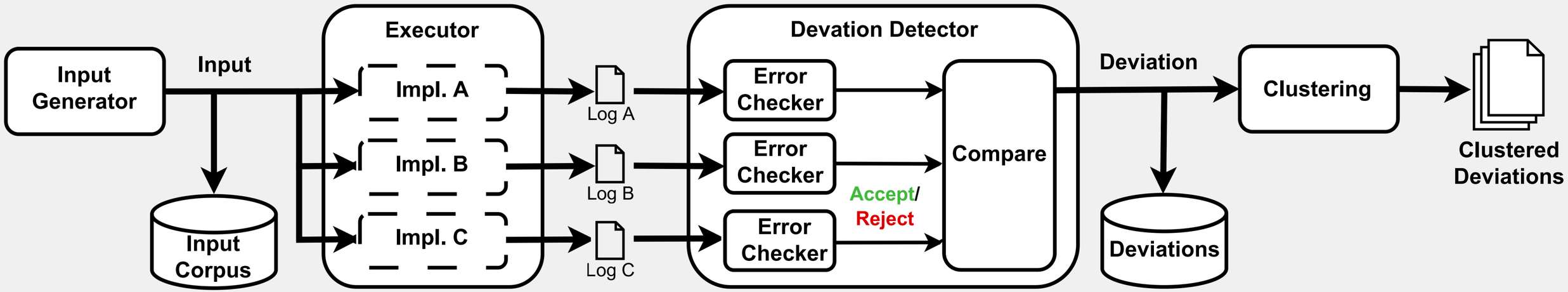
Deviations: Inputs accepted by some, rejected by others

Logs: Leverage generated log files



Clustering: Group deviation with the same root cause

Architecture



Input Generation

Mutational

- AFL++
- Seeds
- Language-agnostic

Coverage-guided

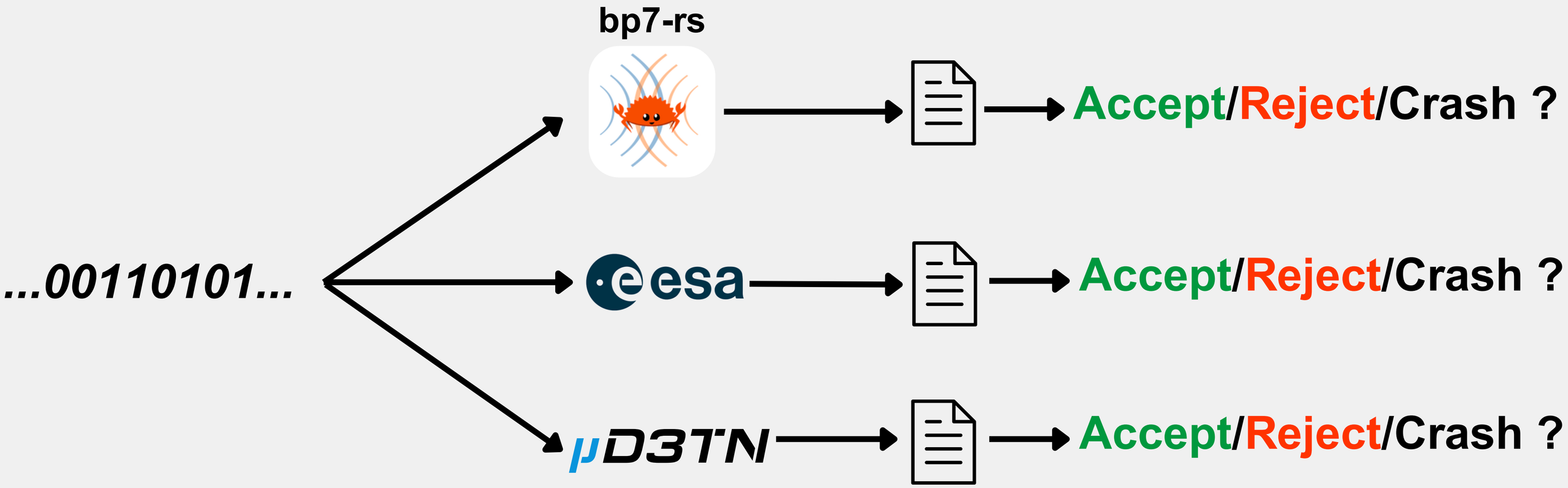
- AFL++
- Tracks novel paths
- Single impl. (μD3TN)

Grammar-based

- Peach Fuzzer
- Requires grammar
- Uses protocol structure

All feed into same executor pipeline

Deviation Detection

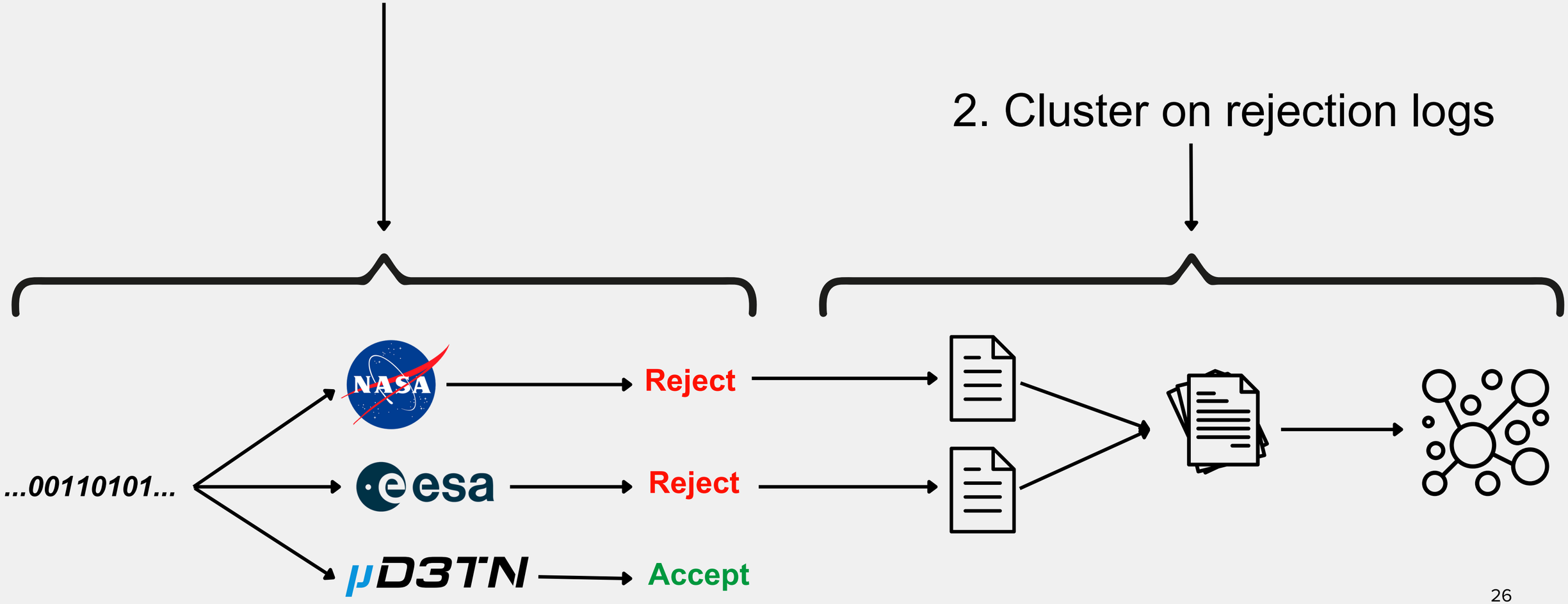


Why logs? → Almost always available!

Clustering

1. Initial clusters from Accept/Reject pattern

2. Cluster on rejection logs



Results: Deviation Detection

Strategy	Tool	All Rejected	Deviation	All Accept	Deviation Inputs	Clusters
Mutational	AFL++	96.46%	1.10%	2.44%	19,516	118
Coverage-guided	AFL++	98.06%	0.85%	1.09%	28,615	313
Grammar-based	Peach	98.11%	0.71%	1.18%	4,364	149

- **Mutational** → Highest deviation rate
- **Coverage-guided** → Most unique clusters
- **Grammar-based** → Lowest deviation rate, but not clusters

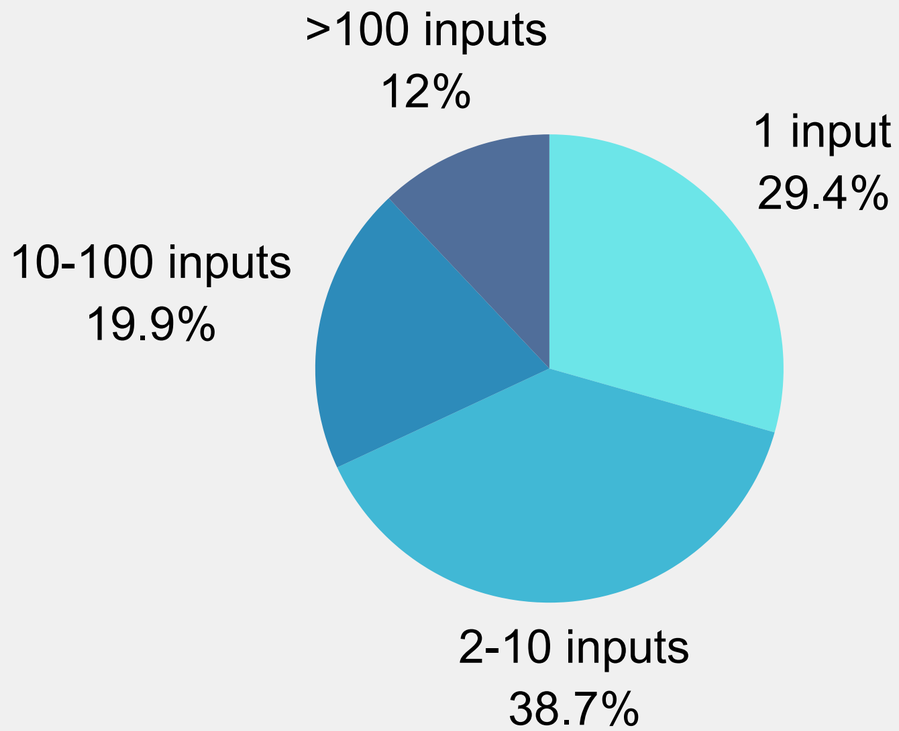
Results: Deviation Clusters

51.373
Deviation inputs

432
Deviation Clusters

- Largest cluster: **5113 inputs**
- Mean cluster size: **~119 inputs**

# of deviation inputs per cluster



Results BPv7: Cluster Analysis

Cluster: ESA_BP+bp7-rs REJECT, uD3TN ACCEPT

Normalized Message:

[ESA BP]: Decoding failed: esa.egos.bp.protocol.coding.api.exceptions.BundleBlockException: Error decoding extension block. Found payload Block Number instead **||**

[bp7-rs]: Error validating bundle: [BundleError("Block numbers occurred multiple times")]

Occurrences: 220

[ud3tn_v0.14.2] OK:

BPv7 bundle

- source: ipn:1.1
- destination: ipn:2.1
- report to: ipn:1.0
- creation ts.: 803395908842
- sequence no.: 0
- expires at: 803396509

- proc. flags: 0x0000
- **block no. 1 of type = 7 (bundle age block)**
- flags: 0x0001
- length: 1
- **block no. 1** of type = 1 (payload block)
- flags: 0x0000
- length: 13

Reported Bugs

10 Requirements violations found

14 Bugs reported

5 Bugs found
- 4 Conformance
- 1 Vulnerability

bp7-rs

4 bugs found
- Conformance

5 Bugs found
- Conformance

11 Bugs already fixed

Limitations

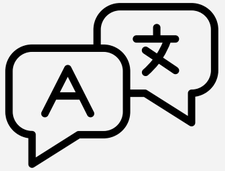
Only testing a **single protocol**

Limited coverage → parser logic only, not full node

Bug goes **undetected** if all implementations under test **share it**

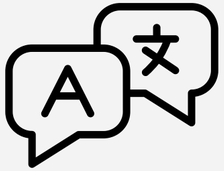
Conclusions

Differential testing is practical and effective in real-world implementations

- Language-agnostic 
- Scalable 
- Finds a variety of conformance issues 

Conclusions

Differential testing is practical and effective in real-world implementations

- Language-agnostic 
- Scalable 
- Finds a variety of conformance issues 

Questions ?

stephan.havermans@imdea.org

Bugs Reported

Impl.	Issue	Reason	Status
µD3TN	Accepts invalid BP version numbers	Too lenient receiver	Fixed
µD3TN	Accepts set status report flags with Source ID none	Too lenient receiver	Fixed
µD3TN	Accepts invalid payload block number	Too lenient receiver	Fixed
µD3TN	Accepts bundle w/ zero timestamp and no Bundle Age block	Too lenient receiver	Fixed
µD3TN	Accepts duplicate block numbers	Too lenient receiver	Fixed
bp7-rs	Accepts invalid block ordering	Too lenient receiver	Fixed
bp7-rs	Accepts malformed `dtn:none` EIDs	Too lenient receiver	Fixed
bp7-rs	Rejects zero timestamp w/ bundle age block present	Incorrect validation	Fixed
bp7-rs	Rejects legal processing flag combination (fragmentation flags)	Incorrect validation	Fixed
ESA BP	Accepts malformed `dtn:none` EIDs	Too lenient receiver	Reported
ESA BP	Accepts invalid CRC type value and wrong CBOR type	Too lenient receiver	Reported
ESA BP	Accepts duplicate block numbers	Too lenient receiver	Reported
ESA BP	Accepts bundle w/ zero timestamp and no Bundle Age block	Too lenient receiver	Reported
ESA BP	IndexOutOfBoundsException kills parsing thread	Uncaught exception	Fixed