A Fuzzing-Based Test-Creation Approach for Evaluating Digital TV Receivers via Transport Streams

Fabricio Izumi, **Eddie Filho**, Lucas Cordeiro, Orlewilson Maia,
Romulo Fabricio, Bruno Farias, Aguinaldo Silva

eddie.filho@tpv-tech.com
TPV Technology

12th Aug 2024

TPV

Overview

Challenges on Digital TV systems

Misconfigured headend **equipment**

Incorrect data structures and protocols formats

Receiver malfunctions and field problems caused by incorrect information in Transport Streams

Proposed approach

Robustness evaluation using **grammar-based guided fuzzing**

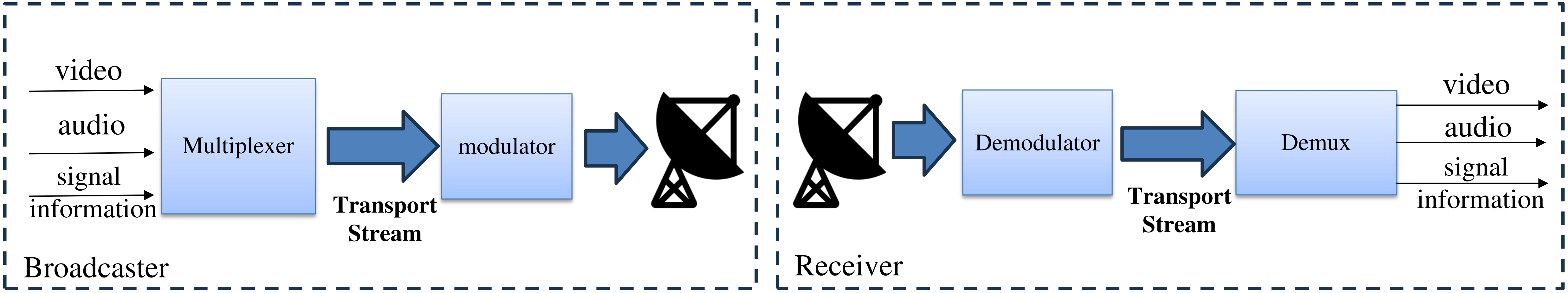
Goals

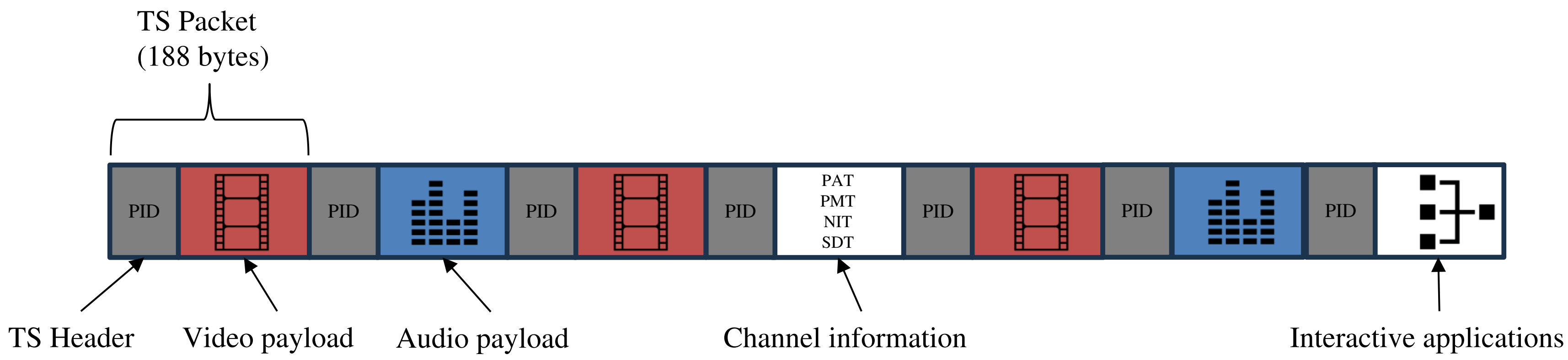
Test receivers under unforeseen conditions

Enhance operational reliability and robustness in commercial DTV platforms

TPV

Terrestrial DTV System Architecture





What happens?

Broadcasters send erroneous data

- Wrong data on transport level:
  - Wrong data in PSI/SI tables;
  - Wrong data in compressed media;
  - Wrong data in interactive applications.
- Who is responsible?
  - Broadcasters are the source;
  - However, ultimately, it is a receiver manufacturers' problem;
  - The solution regards enhanced robustness.
- Causes
  - They look like random events, combining incorrect information and the way software is developed;
  - As an insight, it resembles (guided) fuzzing;
  - Inconsistent encoding of audio and video streams.

Fidings

- Standards check if the structure is ok but not the associated data.
- The is no known methodology in literature to prepare receivers for real error scenarios
- Consequently, our proposal targets robustness testing, based on fuzzing, during development phases.

Field-problem Analysis

Error Sources

- Media-related encoding data:
  - Wrong size information in **H.264 packet headers;**
  - Wrong audio format announced in tables.
- System-related:
  - **Wrong clock references** affecting media synchronization;
  - Intervals between tables (configuration) larger than recommended.
- Data-related:
  - Conditional access information transmitted in free-to-air channels;
  - Non-existent services;
  - Inconsistent encoding of audio and video streams;
  - Incorrect info for interactive applications (compressed?).

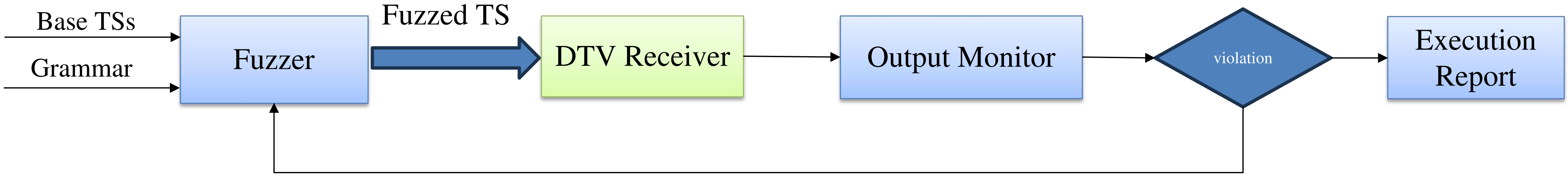
Symptons of failing receivers

- Video freezing or flickering.
- Frame skipping.

Image source: Adobe (https://t.ly/6LtUf)

- Abscence of audio.
- System crash.

DTV-oriented Smart Fuzzer



- Our fuzzer should be:
  - Generation-based using system specifications;
  - Gray-box as analysis and usual implementations are known;
  - Coverage-based targeting entire subsystems;
  - Smart due to the use of known information.
- Additional aspects:
  - Inputs come from known field problems, fragile parts, and DTV standards;
  - Test cases are based on usual implementations and known processing chains;
  - TS processing problems appear on the standard outputs, tha is, audio and video.

Fuzzing DTV-Signal Fields

- The field *stream_type* could be fuzzed to introduce disagreement between content and signalled encoding.
- Incorrect data can be captured by monitoring audio and video outputs.

Syntax	Bitwidth
TS_program_map_section() {	
table_id	8
section_syntax_indicator	1
'0'	1
reserved	2
section_length	12
program_number	16
reserved	2
version_number	5
current_next_indicator	1
section_number	8
last_section_number	8
reserved	3
PCR_PID	13
reserved	4
program_info_length	12
for (i = 0; i < N; i++) {	
descriptor()	
}	
for (i = 0; i < N1; i++) {	
stream_type	8
reserved	3
elementary_PID	13
reserved	4
ES_info_length	12

Grammar Based on the MPEG-2 TS Format

```
program_number = ‘original_network_id’,
service_type,
service_number;
service_type = ‘01’|‘10’|‘11’;
service_number = ‘001’ |‘010’|‘011’|‘100’
                 |‘101’|‘110’|‘111’;
```

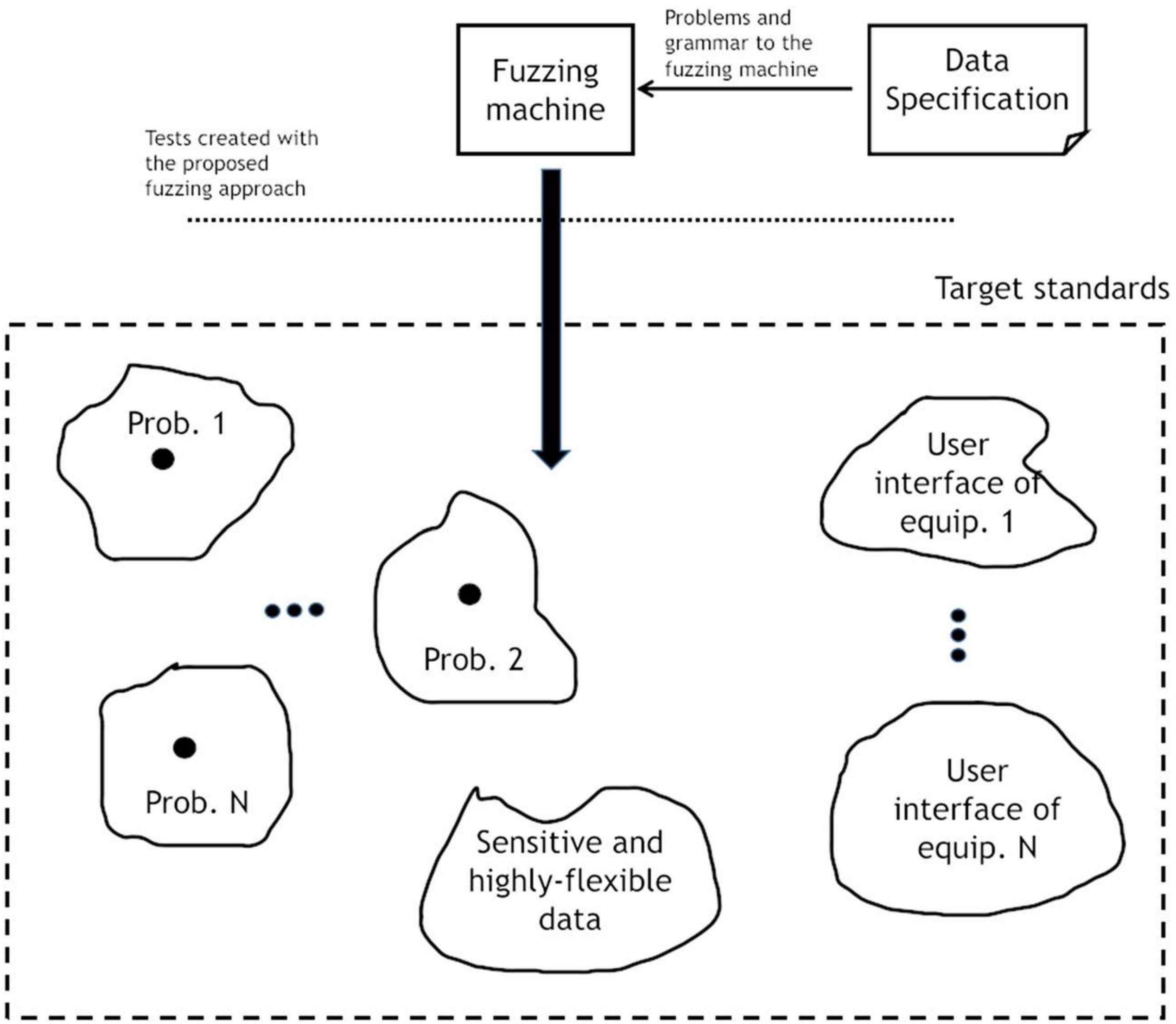
Grammar for *program_number* field

```
component_descriptor = ‘01010000’,
    ‘00000110’,
    stream_content_ext,
    stream_content_and_component_type,
    component_tag,
    ISO_639_language_code;
stream_content_ext = 4 * binary_digit;
stream_content_and_component_type = ‘000100000000’
| (‘0000’, component_type);
component_type = 8 * binary_digit;
binary_digit = ‘0’|‘1’
```

Grammar for *component_descriptor* field

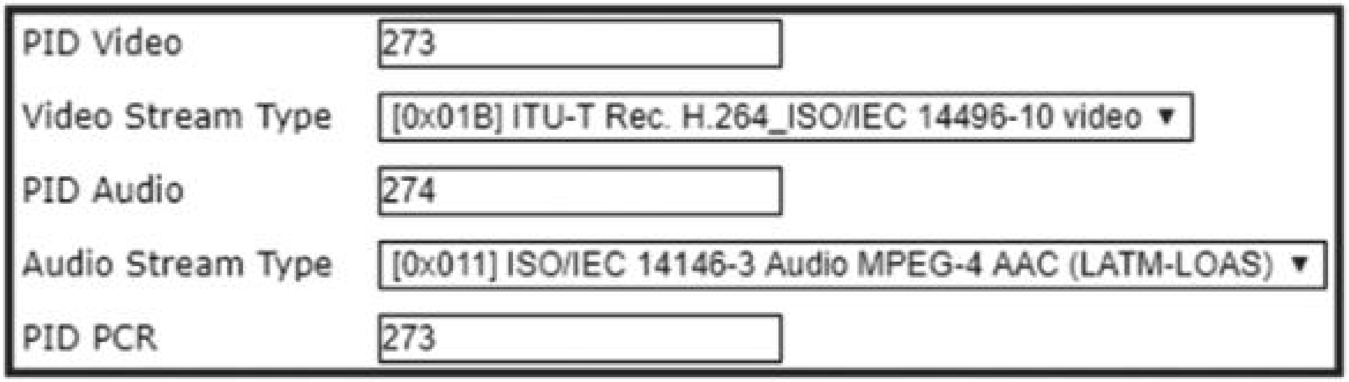
Fuzzing Strategy

- To make the whole approach practical, we defined a test creation strategy.
- Error creation based on areas around field problems, sensitive data, and parameters configured in GUIs is performed with fuzzing.
- Error regions could be continuously expanded if the related random process continues.
- Such an evaluation system can be enhanced over time with new field problems, GUIs, and DTV enhancements.



Example of Possible Configuration Error

- When evaluating the GUI of a commercial multiplexer, we can easily identify fragile spots.



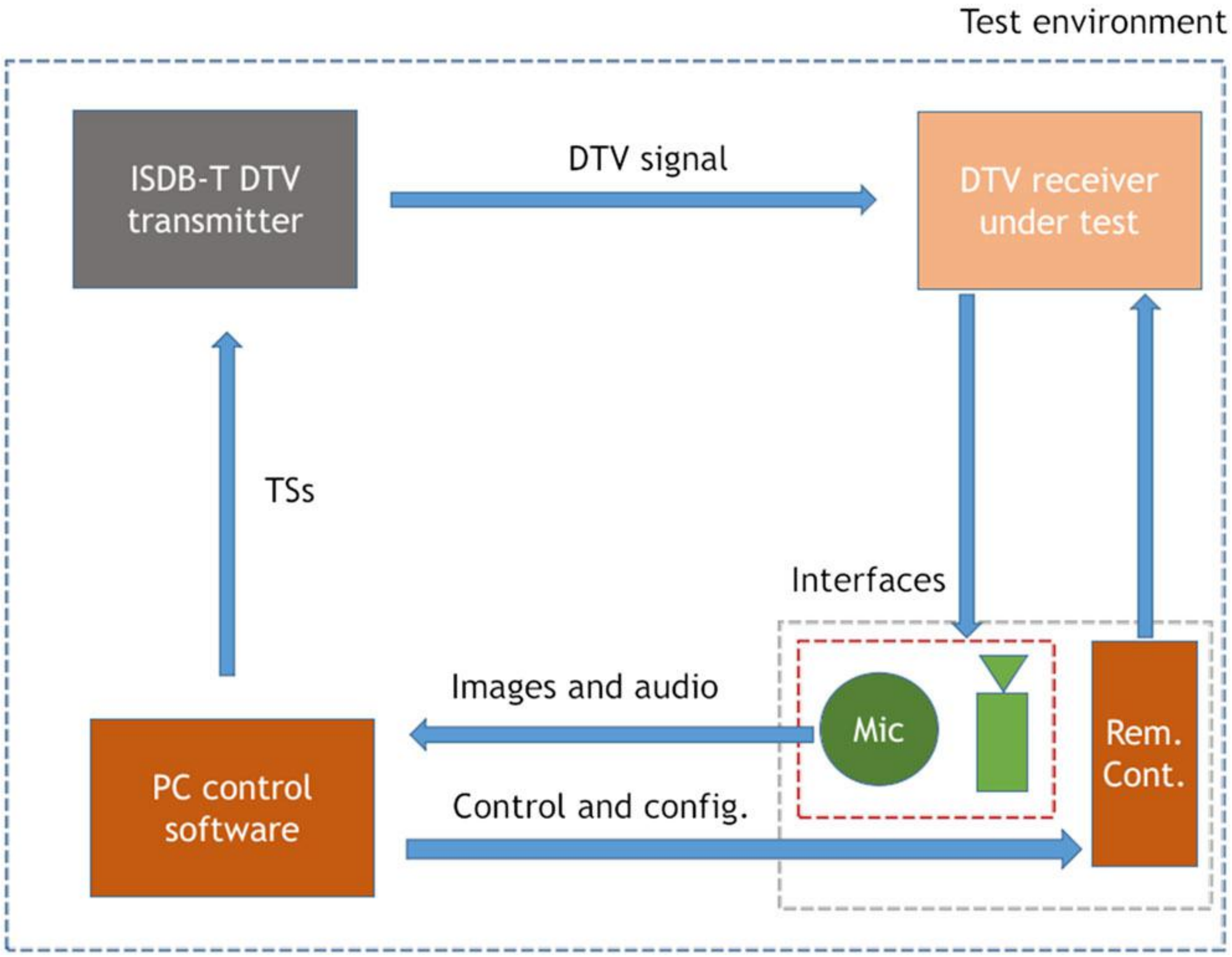
TPV Fuzzing Tool

Image processing module

- Screen detection algorithm.
- Freezing and flickering detection:
  - Histograms;
  - Structural Similarity Index;
  - OpenCV framework.

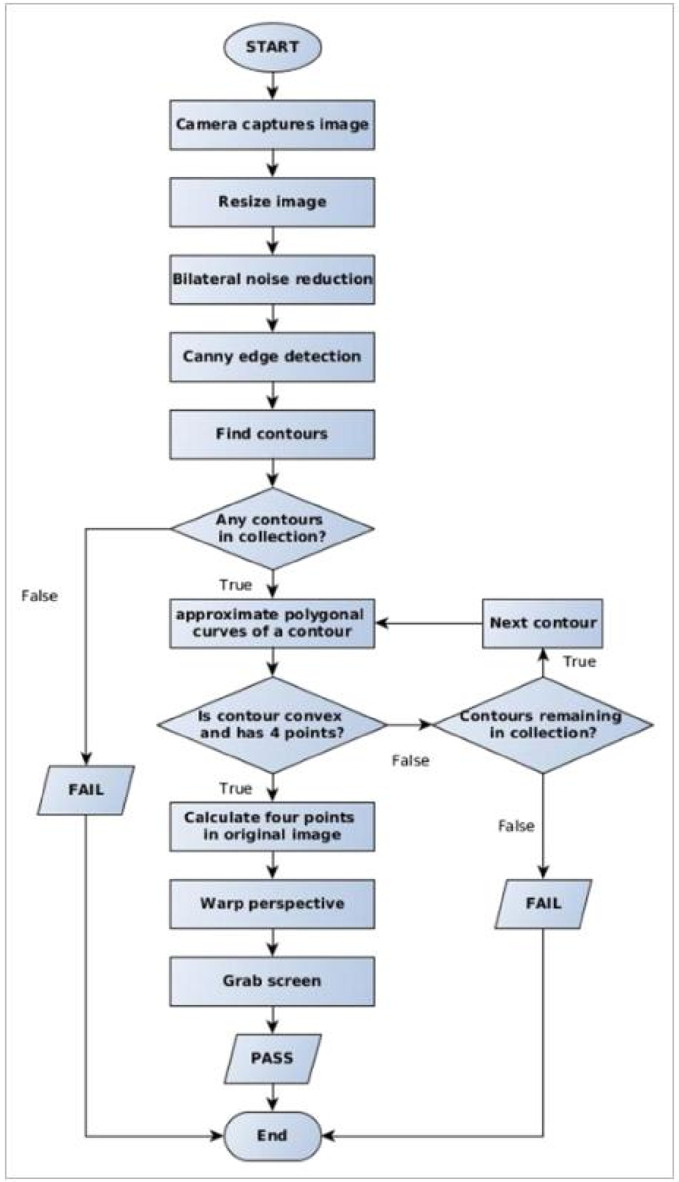
Audio analysis module

- Amplitude verification.
- Frequency verification.
- ALSA library.

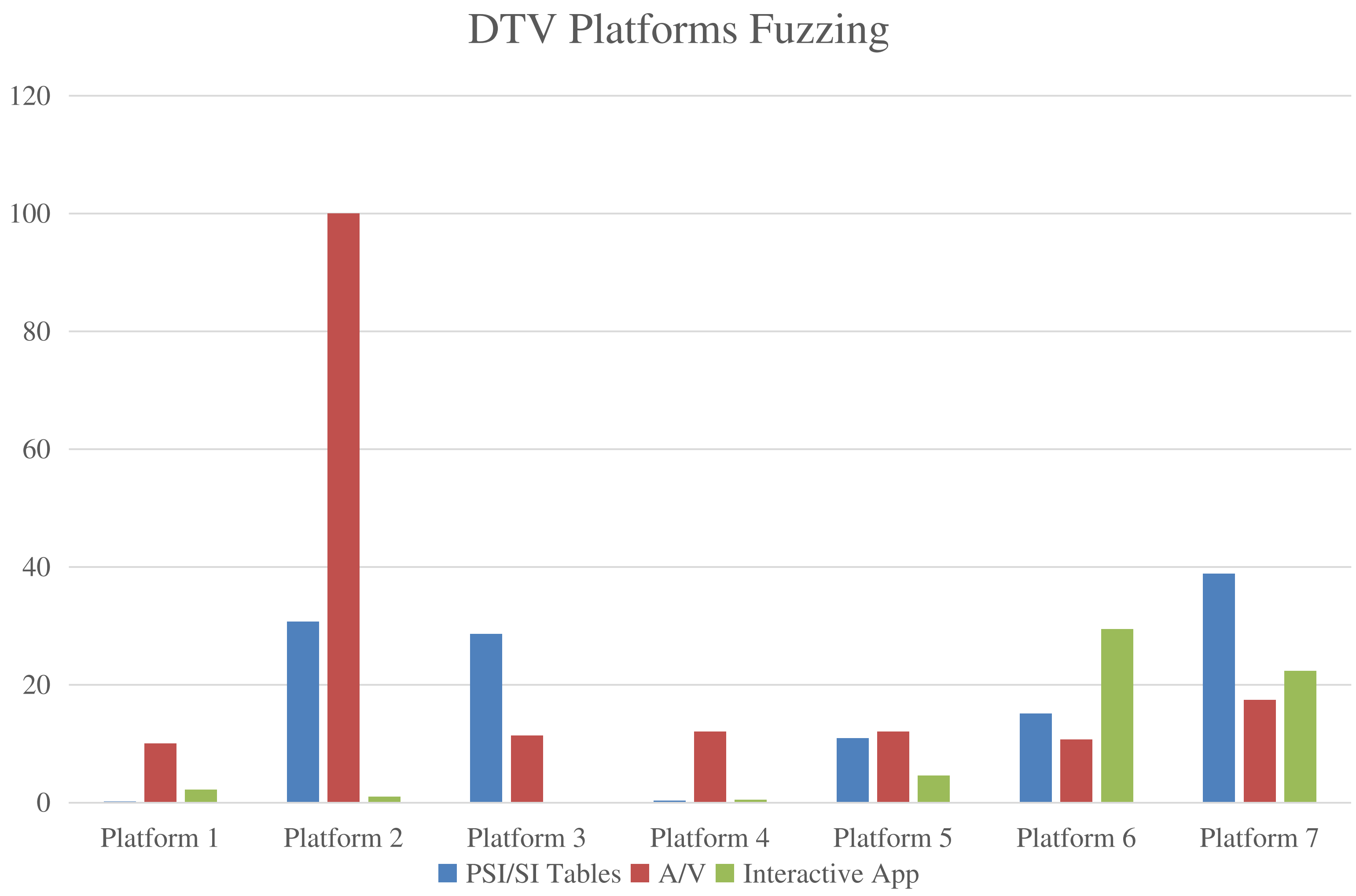


Screen Detection

- Traditional image processing techniques.
- Simple screen detection for segmenting the analysis area.



Experimental Results



Evaluations on 7 commercial platforms.

Platform 5 was under development.

The other platforms are off-the-shelf ones.

The manufacturers represent 80% of the Brazilian DTV market.

Most issues are concentrated in PSI/SI and A/V.

Bug fixes in DTV receiver software impacting millions of users.

Enhancements to devices and transmission setups.

TPV

Important Aspects

- Platforms 2, 3, 6 and 7 surprisingly presented fragile code, even being manufactured by companies with a long history in DTV.
- Platform 2 is a model from 2016, while platforms 6 and 7 were released in 2017, and, finally, Platform 3 was released in 2013.
- DTV receivers usually present a lifespan of at least 10 years.
- The lowest average failure rates regard interactive applications, which indicates a lot of development effort.

TPV Test Groups in Each Category

- PSI/SI:
  - Tables PAT, PMT, NIT, SDT, CAT and EIT, together with their respective descriptors;
  - Table repeat periods;
  - Correlated fields;
  - Services;
  - Media encoding declarations;
  - PID declarations;
  - Table section control data;
  - Virtual channels;
  - Synchronization data.

Test Groups in Each Category

- A/V:
  - Video stream syntax;
  - Video stream syntax;
  - AAC stream elements;
  - LATM stream elements;
  - H.264 profiles and levels;
  - H.264 headers and parameter sets;
  - Audio specific elements (e.g. number of channels and sampling frequency);
  - H.264 SEI messages;
  - H.264 frame information;
  - Video specific elements (e.g. frame rate).

Test Groups in Each Category

- Ginga:
  - DSM-CC syntax;
  - DSM-CC descriptors;
  - DSM-CC compression;
  - DSM-CC section control data;
  - Ginga application syntax;
  - Ginga APIs.

Additional Comparison

- We have also compard only fuzzing engines: ours and Peach.
- We have built the Peach's input format for performing evalution regarding the program map table (PMT), with 95 sections of it.
- We have also created 95 TSs with our approach.
- Platforms 2 and 3 were evaluated: they are popular models from multinational manufacturers, presented many problems, and are provided by market leaders.

	The proposed methodology				Peach [39]			
	Total	Success	Fail	P. Failed (%)	Total	Success	Fail	P. Failed (%)
Platform 2	95	59	36	37.89%	95	75	20	21.05%
Platform 3	95	55	40	42.11%	95	82	13	13.68%

Conclusion and Future Work

- Our work presents a **collection of real field problems** identified in DTV networks and outlines **a scheme for non-compliance insertion** that performs **grammar-based guided fuzzing**.

- The experimental results showed that our methodology is **effective on finding real problems** on comercial Digital TV platforms.

- In terms of fuzzing technique, we envision future work on applying machine learning algorithms that provide adaptability toward known fragile parts.

Thank you!

- Izumi, Fabrício; de Lima Filho, Eddie B.; Cordeiro, Lucas C.; Maia, Orlewilson; Fabrício, Rômulo; Farias, Bruno; Silva, Aguinaldo. A fuzzing-based test-creation approach for evaluating digital TV receivers via transport streams. Software Testing, Verification and Reliability, 2022.