

asn1.io/
CDR Inspector

zooming into call records

by OSS NOKALVA, INC.

Overview

Purpose

- Troubleshoots Call/Charging Data Records (CDRs)
- Various formats supported (GSM TAP, 3GPP TS 32 205, 3GPP TS 32 298, and more)
- Improves the accuracy of billing records
- Allows scripting over decoded results (exported as JSON)

Scenarios

- Find errors across multiple records
- Find data mismatches with a schema
- Detect and recover malformed TLV data
- Handle large encodings and/or large number of records
- Detect the record type
- Detect CDR file headers
- Programmatically process and interpret the decoding results



Focusing on data

With CDR Inspector the schema becomes “secondary”, while a variety of file formats, including large files, can be decoded and/or errors detected.

- The **schema is optional**. Decoding raw BER can be done without the schema. This helps to catch low level encoding errors, and make general sense of data, e.g. search for text fragments, etc.
- If the **data mismatch the schema type**, the closest matching type can be auto-detected.
- CDR files can contain **one or more records**.
- **Large files** with thousands of records can be examined in chunks.
- CDR files with headers are supported, including **auto detection of headers**.

Example 1

1. Specify the data by either uploading a file or pasting as hex text.
2. Optionally pick the corresponding schema and type.
3. Note, CDR Inspector can help you with detecting the type by scanning the schema for the best match for your data.
4. Use options to control the output, to search for text, and more.
5. Navigate records, nodes, and low level encoding details.

The screenshot shows the 'INPUT' tab of the CDR Inspector tool. It is divided into two main sections: 'Step 1' and 'Step 2'.
Step 1: 'Provide the data for the schema / type below. Usually it's a file containing multiple ASN.1 BER-encoded records (may also include non-ASN.1 headers)'. A dropdown menu is set to 'Enter manually (HEX)'. Below it, a text area contains hex data: `30 30 80 11 00 FF FF FF FF FF FF FF FF FF FF FF 81 06 46 61 6C 63 6F 6E 04 80 02 46 50 A5 0A 0C 03 43 61 72 0C 03 47 50 53`. A red circle with the number '1' highlights this area.
Step 2: 'Please provide your schema (asn file) and the type that matches the records in your data file. Schema provides structure and semantic information for the data.' A dropdown menu is set to 'Enter schema manually'. Below it, a text area contains the schema: `World-Schema DEFINITIONS ::= BEGIN Rocket ::= SEQUENCE`. A red circle with the number '2' highlights this area.
Type: A dropdown menu is set to 'World-Schema.Rocket'. A red circle with the number '3' highlights this area.
Options: Located on the right side. A dropdown menu is set to '- unspecified'. A red circle with the number '4' highlights this area. Below it, there are checkboxes for 'Match word' and 'Match whole word'. A 'Search for the provided text (matches marked with ✓)' field is present. There is a 'Max characters to display for values' field set to '100'. There are radio buttons for 'content-dependent (TBDC, ASCII, HEX)', 'ASCII', 'hexadecimal', 'length-dependent', 'binary', and 'hexadecimal'. There is a checkbox for 'Show missing OPTIONAL'. A dropdown menu for 'Advanced options' is visible at the bottom right.

The screenshot shows the 'OUTPUT' tab of the CDR Inspector tool, displaying the 'export results'. It features a hex dump on the left and a detailed view of a record on the right.
Hex Dump: `30 30 80 11 00 FF FF FF FF FF FF FF FF FF FF 81 06 46 61 6C 63 6F 6E 83 FF FF FF FF FF FF FF 81 06 46 61 6C 63 6F 6E 83 01 00 A4 04 80 02 46 50 A5 0A 0C 03 43 61 72 0C 03 47 50 53`. A red circle with the number '5' highlights the first few bytes.
Record View: Shows 'Record #0/1' with a navigation bar. Below it, a 'Data summary' box states: 'total 1 record covered, max record size 50 bytes. See the details of specific record(s) below.' Below the summary, the record details are shown:
Record #0: offset 0, length 48 bytes (0% errors) SEQUENCE
SEQUENCE (4 case)
range : 348282366920938463463374607431768211455
name : Falcon
message : "Hello World"
Note: TLV is absent, DEFAULT value is assumed.
Schema: message UTF8STRING
Offset: n/a
Tag: n/a
Len: n/a
Val: n/a

Advanced features

Advanced features	Generic CDR decoders	CDR Inspector
TLV errors	✗ skipping, breaking	✓ detecting, revealing, recovering
Schema mismatch	✗ breaking	✓ detecting, revealing, recovering
Decoding DEFAULTs and OPTIONALs	✗ excluded from the output	✓ shown in the output
TBCD strings	✗ undetected	✓ detected
Search decoding results	✗ none	✓ all
TLVs embedded inside OCTET/BIT STRINGS	✗ undetected	✓ detected
Merging constructed encodings of multi-fragmented strings	✗ unsupported	✓ supported

Example 2

1. Low level errors in catching bad **TLV encodings**
2. High level errors in catching **schema mismatches**
3. Showing **DEFAULTS** and **OPTIONALS**
4. **Search** for "ENUMERATED" - the record and the field containing the searched text is marked with ✓
5. Detecting **TLVs embedded** in OCTET/BIT STRINGS

The image displays two screenshots of a network analysis tool's output window, showing TLV parsing results. The top screenshot shows a single record with a schema mismatch error. The bottom screenshot shows a record with an embedded TLV.

Top Screenshot (Record 0-0/1):

- 3. DEFAULT value:** A callout points to the 'SEQUENCE (6 comp)' field.
- 2. Schema mismatch:** A callout points to the error message: "Error: Bad value of a BOOLEAN. Expected tags: [CONTEXT 3]. Schema: fuel ENUMERATED {solid, liquid, gas}".
- 1. TLV error:** A callout points to the 'fuel : True' field.
- 4. Searched for ENUMERATED:** A callout points to the 'fuel : True' field, which is highlighted in yellow.

Bottom Screenshot (Record 0-3/4):

- 5. Embedded TLV:** A callout points to a yellow note: "Note: TLV embedded into the BIT/OCTET string above. No schema available".

Power Tips

- **Re-inspect parts** of your data by selecting the HEX portion of a record, and decoding it with CDR Inspector again, potentially specifying a different schema type.
- Use **“Sort by errors”** to find bad records quickly.
- **Export** the inspector's output in **JSON format** for programmatic processing and analysis of decoding results.

DIFFERENTIATORS

OSS offers more

ACCESSIBILITY

The online platform requires no installation and provides access from any browser on any platform.

SUPPORT

OSS offers comprehensive technical support with a long list of customers and decades of history.

EXPERTISE

OSS employed the industry experts for all aspects of using ASN.1 in a right way

EASE OF USE

OSS' web tools are user-friendly and produce intuitive results. The UI offers a wealth of options to fit your scenario.