RfidResearchGroup / **proxmark3** Public

Code Issues 84 Pull requests 12 Actions Projects Wik

master proxmark3 / doc / **trace_notes.md** Go to file ···

[redacted]

[redacted]

Notes about the tracelog

Table of Contents

Trace command

^Top

The `trace` command lists the data exchange by the proxmark3 and a tag or a reader in human readable form.

With `trace list` a table is shown which gives timing information, the src of the data bytes, the transmitted/received bytes itself, a check if the CRC was correct and some decoding of the command.

Timing

^Top

The Start and the End column lists timestamps when the transmission of the shown data started (time of first bit) and when it ended (end of last modulation).

The unit for this time information depends on the protocol in use:

- ISO14443A and Thinfilm: all times are in carrier periods (1/13.56MHz)
- For Legic timing information depends also on direction:
- Reader Mode: Timings are in ticks (1us == 1.5ticks)
- Tag Mode: Timings are in sub carrier periods (1/212 kHz == 4.7us)
- Hitag1 / Hitag2 / HitagS: Elementary Time Unit (ETU) is 8µs
- iCLASS, ISO15693, ISO18092 and FeliCa have no accurate timing information at the moment
- For others timing is not available

By specifying the option `f` (e.g. `trace list -t 14a -f`) the frame delay times are shown. (So you don't have to do the math by your own).

Sources

^Top

If the data is marked as a response the source is shown as Tag. Otherwise it is marked as Reader (Rdr).

Data

^Top

This column shows the raw bytes transmitted over the air. With option `c` CRC bytes are marked in square brackets.

CRC

^Top

Marks if the transmitted CRC matches with the calculated CRC.

Annotation

^Top

Annotations provide a rough decoding of the transmitted data. For ISO14443A a more detailed decoding is available with Wireshark (s. next chapter)

Tracelog format

^Top

The binary format for the dynamic tracelog is as following.

```
/*
  Traceformat:
  32 bits timestamp (little endian)
  16 bits duration (little endian)
  15 bits data length (little endian) (0x7FFF)
  1 bit isResponse (0=reader to tag, 1=tag to reader)
  data length Bytes data
  x Bytes parity,  where x == ceil(data length/8)
```

```
*/

typedef struct {
    uint32_t timestamp;
    uint16_t duration;
    uint16_t data_len : 15;
    bool isResponse : 1;
    uint8_t frame[];
    // data_len         bytes of data
    // ceil(data_len/8) bytes of parity
} PACKED tracelog_hdr_t;

#define TRACELOG_HDR_LEN        sizeof(tracelog_hdr_t)
#define TRACELOG_PARITY_LEN(x)  (((x)->data_len - 1) / 8 + 1)
```

Trace and Wireshark

^Top

To get a more detailed explanation of the transmitted data for ISO14443A traces the output can be converted to a pcapng file to read it with Wireshark.

- copy the output (starting with the timestamp) into a textfile
- run `text2pcap -t "%S." -l 264 -n <input-text-file> <output-pcapng-file>`
- now open your pcapng file in Wireshark or read it with the CLI version `tshark`

An example frame

with `trace list -t 14a`:

```
19072 |      29536 | Rdr |93  70  88  04  cf  ff  bc  7f  bb
|  ok | SELECT_UID
```

the same data with `tshark -r foo.pcapng -V -x`:

```
Frame 5: 13 bytes on wire (104 bits), 13 bytes captured (104
bits) on interface 0
    Interface id: 0 (unknown)
        Interface name: unknown
    Encapsulation type: ISO 14443 contactless smartcard
standards (177)
    Arrival Time: Aug 17, 2019 23:17:00.000002606 CEST
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1566076620.000002606 seconds
    [Time delta from previous captured frame: 0.000000840
seconds]
    [Time delta from previous displayed frame: 0.000000840
seconds]
    [Time since reference or first frame: 0.000001907
seconds]
    Frame Number: 5
    Frame Length: 13 bytes (104 bits)
    Capture Length: 13 bytes (104 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: iso14443]
ISO 14443
    Pseudo header
        Version: 0x00
        Event: Data transfer PCD -> PICC (0xfe)
        Length field: 9
    Message: Select
        SEL: 0x93
        NVB: 0x70
        CT: 0x88
        UID_CLn: 04cfff
        BCC: 0xbc
        CRC: 0xbb7f [correct]
        [CRC Status: Good]
```

```
0000  00 fe 00 09 93 70 88 04 cf ff bc 7f bb
.....p.......
```

If the Wireshark ISO14443a dissector is missing some commands or needs some other rework please file a bug.