OASIS Wireshark Dissector (oasis_wireshark_dissector.lua)

This Wireshark Lua plugin dissects the OASIS File Transfer Protocol. It is designed to work with PCAP files generated by `oasis_send` and `oasis_recv` utilities, which use `LINKTYPE_USER2 (149)`.

Features

- Dissects OASIS protocol packets, identifying commands like OPEN, WRITE, and CLOSE.
- Parses and displays fields from the Directory Entry Block (DEB) in OPEN packets.
- Calculates and verifies the Longitudinal Redundancy Check (LRC).
- Decodes DLE stuffing and Run-Length Encoding (RLE) in packet payloads.
- Identifies ACK, ENQ, and EOT signals.
- Provides expert information for potential issues like LRC mismatches or malformed packets.

Installation

To use this dissector, you need to place the `oasis_wireshark_dissector.lua` file into your Wireshark plugin directory.

Finding your Wireshark Plugin Directory:

1. Open Wireshark.
2. Go to `Help -> About Wireshark`.
3. In the pop-up window, go to the `Folders` tab.
4. Locate the path next to `Personal Lua Plugins` or `Global Lua Plugins`. This is where you should place the Lua script. If the `Personal Lua`

`Plugins` directory does not exist, you may need to create it.

Installation Steps:

1. Download the `oasis_wireshark_dissector.lua` script.

2. Copy the `oasis_wireshark_dissector.lua` file to your Wireshark Personal Lua Plugins directory.

    - **Windows:** Typically `C:\Users\<YourUsername>\AppData\Roaming\Wireshark\plugins` or `%APPDATA%\Wireshark\plugins`. If `AppData` is hidden, you might need to show hidden files in File Explorer. Alternatively, you can use the path shown in Wireshark's About dialog.

    - **Linux:** Typically `~/.local/lib/wireshark/plugins` or `~/.config/wireshark/plugins`. The global plugin directory might be `/usr/lib/x86_64-linux-gnu/wireshark/plugins/` or similar, but personal plugins are recommended.

    - **macOS:** Typically `~/.local/lib/wireshark/plugins` or `~/Library/Application Support/Wireshark/plugins`. The global plugin directory might be `/Applications/Wireshark.app/Contents/Resources/share/wireshark/plugins/` but personal plugins are recommended.

3. Restart Wireshark, or reload Lua plugins by going to `Analyze -> Reload Lua Plugins` (or press `Ctrl+Shift+L`).

How to Use

1. **Capture OASIS Traffic:** Use the `oasis_send` or `oasis_recv` utilities with the `--pcap <filename.pcap>` option to generate a PCAP file containing the OASIS serial communication. For example:

    ```
    ./oasis_send COM1 MYFILE.TXT_S --pcap transfer.pcap
    ./oasis_recv /dev/ttyS0 --pcap received.pcap
    ```

2. **Open PCAP in Wireshark:** Open the generated `.pcap` file in Wireshark.

3. **View Dissected Packets:** Wireshark should automatically recognize the OASIS protocol packets (if they use `LINKTYPE_USER2`

(149)) and display the dissected information in the packet details pane.

- The Direction field will show whether the data was Transmitted (TX) or Received (RX) from the perspective of the capturing utility (oasis_send/oasis_recv).
- Message Type will identify the type of communication (e.g., ENQ, ACK, Data Packet).
- For Data Packets:
  - Command: Will show OPEN, WRITE, or CLOSE.
  - Payload (Raw/Stuffed): Shows the payload before DLE/RLE decoding.
  - Payload (Decoded Bytes): Shows the actual data after decoding.
  - If the command is OPEN, a "Directory Entry Block (DEB)" section will appear, showing the parsed fields of the file being opened.
  - LRC (Received) and LRC (Calculated) will be shown. Expert info will highlight mismatches.

4. **Filtering:** You can filter for OASIS protocol packets using the display filter oasis.

Dissector Fields Overview

The dissector provides the following main fields:

- oasis.direction: Indicates if the packet was RX or TX.
- oasis.type: General type of the message (ENQ, ACK, Data Packet, etc.).
- **ACK Packets:**
  - oasis.ack.toggle: The toggle bit ('0' or '1') of the ACK.
- **Data Packets (oasis.packet.*):**
  - oasis.packet.cmd: The command type (OPEN, WRITE, CLOSE).
  - oasis.packet.payload_raw_stuffed: The payload before DLE/RLE decoding.

    - `oasis.packet.payload_decoded.bytes`: The decoded payload data.
    - `oasis.packet.payload.seq_link`: For WRITE packets (potentially sequential), the next sector link.
    - `oasis.packet.lrc.received`: The LRC byte received in the packet.
    - `oasis.packet.lrc.calculated`: The LRC calculated by the dissector.
- **DEB Fields (`oasis.deb.*`) - for OPEN packets:**
    - `oasis.deb.file_format.type`: The type of file (Sequential, Direct, etc.).
    - `oasis.deb.file_format.attributes_str`: File attributes (Read, Write, Delete protected).
    - `oasis.deb.file_name_str`: File name.
    - `oasis.deb.file_type_str`: File type/extension.
    - `oasis.deb.record_count_val`: Number of records.
    - `oasis.deb.block_count_val`: Number of 1K blocks.
    - `oasis.deb.start_sector_val`: Starting logical sector address.
    - `oasis.deb.ffd1.*`: Format Dependent Field 1, broken down by file type (e.g., record length, key length).
    - `oasis.deb.timestamp.decoded_str`: Decoded file timestamp.
    - `oasis.deb.owner_id_val`: Owner ID.
    - `oasis.deb.ffd2.*`: Format Dependent Field 2, broken down by file type (e.g., last sector, load address, program length).

Troubleshooting

- **Dissector Not Loading:**
    - Ensure Wireshark was restarted or Lua plugins were reloaded after placing the script.
    - Check `Help -> About Wireshark -> Folders` to confirm you used the correct plugin directory.
    - Check `Analyze -> Enabled Protocols...` and make sure "OASIS" is checked.

    - Look for Lua errors in Wireshark's console output (if available/enabled) or in Help -> Internals -> Lua Errors.

- **Packets Not Dissected as OASIS:**
    - Verify that the PCAP file was generated with LINKTYPE_USER2 (149). The oasis_send/oasis_recv utilities are designed to do this. If using other capture methods, the link-layer type might be different, and this dissector won't apply directly.
    - The first byte of each frame's data (after the libpcap pseudo-header for LINKTYPE_USERN) *must* be the direction byte (0x00 for RX, 0x01 for TX) that oasis_pcap.c prepends.

- **LRC Mismatch:**
    - Indicates potential corruption in the captured data or an issue with the LRC calculation in the sending/receiving utility or the dissector.

- **Malformed Packet Errors:**
    - The dissector tries to identify issues like missing trailers or incorrect DLE stuffing. This could point to problems in the communication or capture.