Analyzing Bluetooth Snoop Log of Android Infocar App for Vehicle Diagnostics

2022.10.07

Author : Hojun Seong, Jiheun Jung, Sangchul Han, Minkyu Park, Seong-je Cho

Presenter : Jiheun Jung

Affiliation : Dankook University

Email : wlgjsjames@dankook.ac.kr

INDEX

01 Introduction

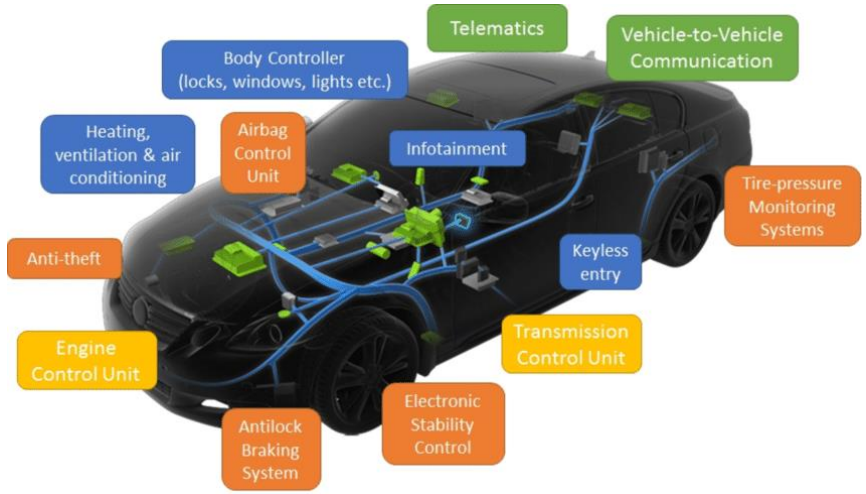
Introduction

❖ The complexity of vehicle's internal systems increases

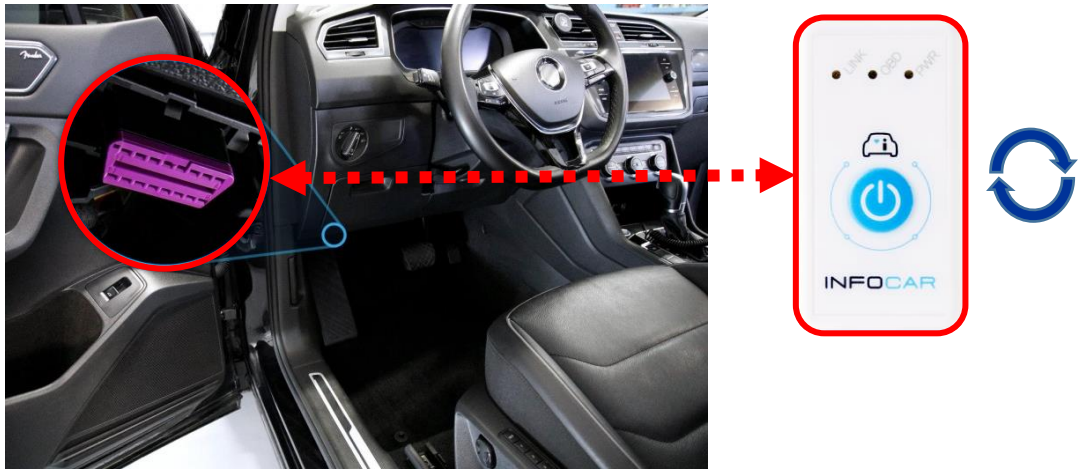
- Vehicle electrification

❖ Diagnostic app services for drivers

- OBD-II scanner
  - Connect to a mobile app through Bluetooth or WiFi
- Data exchange
  - Packet data : mobile device and the OBD scanner connected with Bluetooth



[Source : "Security concerns in co-operative intelligent transportation systems" (2017)]



※ OBD(On-Board Diagnostics) : the standard used to self-diagnose and report the internal status of a vehicle

[Source : https://infocarmobility.com]

Introduction

❖ Vehicle accident perspective

- Help to Investigate and identify the cause

❖ In this paper,

- Pre-planned scenario-based packet collection and analysis
  - OBD-II scanner is connected to your mobile device through Bluetooth
  - Collection : Bluetooth HCI snoop log function
  - Analysis based on specific driver's behavior
  - Discuss how to use the data in actual vehicle accidents

02 Background & Related Work

Background & Related Work

1. Bluetooth HCI Snoop Log function

   - Records of all HCI(Host Controller Interface) processes performed on the device
   - Android version 4.4(Kitkat) or later devices

2. Bluetooth Profile

   - Specification that defines dependencies on the protocol stack for applying Bluetooth technology and other interfaces that require interaction
     - AVRCP(A/V Remote Control Profile), HFP(Hands-Free Profile), SPP(Serial Port Profile), etc.

3. AT Command

   - Commands used to control the modem(ELM327)

4. OBD-II PID(Parameter ID)

   - Code used to request diagnostic data from the vehicle

※ ELM327 : Microcontrollers designed to act as bridges between OBD ports and standard RS232 serial interfaces

Background & Related Work

※ **DoIP :** *A protocol that makes it possible to diagnose a vehicle in a remote location online*
※ **UDS :** *Protocol used by the diagnostic system to communicate with the vehicle's ECU (ISO 14229)*

❖ [6] : Buquerin et al. “A generalized approach to automotive forensics.”

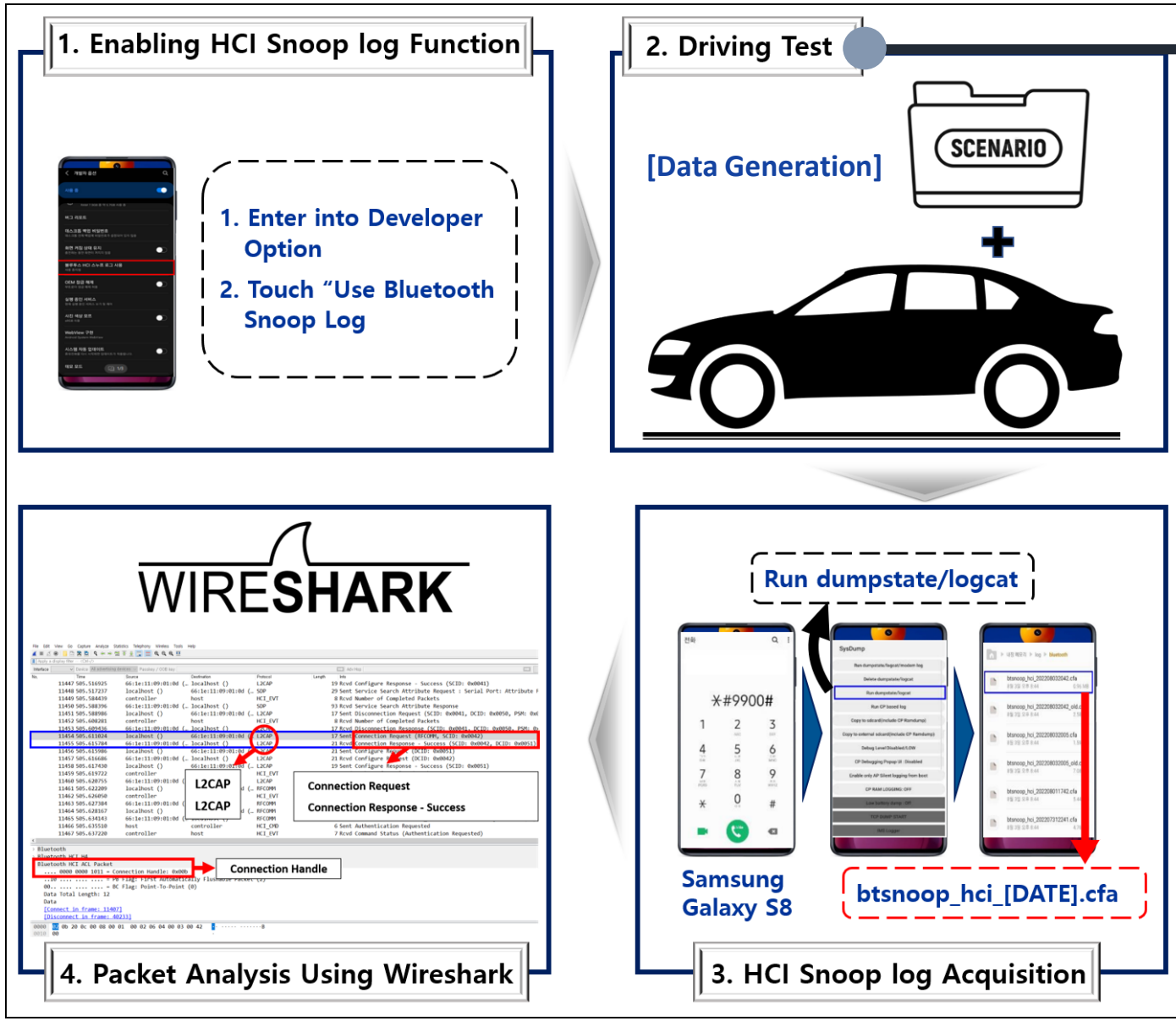
- Presented the process of performing digital forensics
  - On the state-of-art vehicle system
- Performed a case study to demonstrate the automotive forensics process
  - Used OBD scanners and self-developed applications
    - DoIP(diagnostic over IP) & UDS(Unified Diagnostic Service)
  - Monitored and collected network traffic, and analyzed data using Wireshark
    - Result : Manufacturer-Specific ID, TCP handshake, connection setting data

❖ [7] : Lee et al. “Identifying User Behavior for Vehicle Forensic Analysis from Bluetooth HCI Snoop Log of a Mobile Device Paired with an Android-based Car AVN.”

- Collected Bluetooth HCI Snoop Log
  - Connected the Android-based AVN to mobile devices
- Analyzed packet after performing a specific user’s behavior
  - Searching the contacts, searching the recent call list, and playing music
  - Reconstruction of driver’s behavior
    - Useful in the investigation of vehicle accidents and crimes

03 Process of Bluetooth HCI Snoop Log Analysis

Process of Bluetooth HCI Snoop Log Analysis



Process of Bluetooth HCI Snoop Log Analysis

TABLE 1. SCENARIO INCLUDING SPECIFIC USER BEHAVIOR

Time	Scenario	Description
	Departure	
pm 7:26	Starting the engine	-
7:29	Connecting Required Equipment	Plugging in OBD scanner to OBD port and connecting with the diagnosis app
7:30	Departing for the stopover	-
7:43	Arriving at the stopover	Stopping the car
7:44	Stopping the engine	Stopping the engine without unplugging the OBD scanner
7:48	Starting the engine	-
7:49	Reconnecting the equipment	Reconnecting the diagnosis app
7:50	Departing for the destination	-
7:59	Arriving at the destination	Stopping the car
8:00	Stopping the engine	Stopping the engine and disconnecting all equipment and the app

TABLE 2. DRIVING TEST ENVIRONMENT

Vehicle	
Manufacturer	Kia Motors
Model	K3
Year	2018
Mobile device	
Model	Samsung Galaxy S8
Android Version	Android 9.0(Pie)
OBD-II based vehicle diagnostic platform	
OBD-II Scanner	Infocar IO180-OH
Mobile Application	Infocar 2.24.35

04

Bluetooth HCI Snoop Log Analysis based on Specific User Behavior

HCI Snoop Log Analysis based on Specific User Behavior

❖ Packet Analysis for Specific User Behavior

- Pattern, Profile/Protocol, Data

Three Specific User Behaviors

Index	Description
1	Plug and connect OBD scanner
2	Drive
3	Stop and turn off engine

1. Plug and connect OBD Scanner

TABLE 3. EXAMPLE OF PACKETS GENERATED WHEN PAIRING BETWEEN SCANNER AND VEHICLE DIAGNOSTICS APP

Time	Source	Destination	Profile/Protocol	Info
pm 07:29 :13.27	Galaxy S8	InfocarOH	L2CAP(Protocol)	Connection Request
07:29 :13.30	InfocarOH	Galaxy S8	L2CAP(Protocol)	Connection Response-Pending
07:29 :13.35	InfocarOH	Galaxy S8	L2CAP(Protocol)	Connection Response-Success
07:29 :13.42	Galaxy S8/ InfocarOH	InfocarOH/ Galaxy S8	SPP(Profile)	Initializing Using AT Command

Pattern

TABLE 1. SCENARIO INCLUDING SPECIFIC USER BEHAVIOR

Time	Scenario	Description
Departure		
pm 7:26	Starting the engine	-
7:29	Connecting Required Equipment	Plugging in OBD scanner to OBD port and connecting with the diagnosis app
7:30	Departing for the stopover	-
7:43	Arriving at the stopover	Stopping the car
7:44	Stopping the engine	Stopping the engine without unplugging the OBD scanner
7:48	Starting the engine	-
7:49	Reconnecting the equipment	Reconnecting the diagnosis app
7:50	Departing for the destination	-
7:59	Arriving at the destination	Stopping the car
8:00	Stopping the engine	Stopping the engine and disconnecting all equipment and the app

※ L2CAP(Logical Link Control and Adaption Protocol) : *protocol used in the Bluetooth standard that provides adaption between higher layers and the baseband layer of the Bluetooth stack*

HCI Snoop Log Analysis based on Specific User Behavior

❖ Packet Analysis for Specific User Behavior(Cont.)

2. Drive

- SAE 1979 : E/E Diagnostic Test Modes

TABLE 4. EXAMPLE OF PACKETS GENERATED BETWEEN DEPARTURE/ARRIVAL TRIP TO A WAYPOINT

Time	source	Destination	Protocol/Profile	Info
pm 07:30 :33.61	Galaxy S8	InfocarOH	SPP(Profile)	"010D\r"
07:30 :33.60	Infocar OH	Galaxy S8	RFCOMM (Protocol)	UIH Channel=1 UID
07:30 :33.64	Infocar OH	Galaxy S8	SPP(Profile)	"7E803410D03 \r"
07:30 :33.68	Infocar OH	Galaxy S8	SPP(Profile)	"\r>"

Pattern

TABLE 1. SCENARIO INCLUDING SPECIFIC USER BEHAVIOR

Time	Scenario	Description
Departure		
pm 7:26	Starting the engine	-
7:29	Connecting Required Equipment	Plugging in OBD scanner to OBD port and connecting with the diagnosis app
7:30	Departing for the stopover	-
7:43	Arriving at the stopover	Stopping the car
7:44	Stopping the engine	Stopping the engine without unplugging the OBD scanner
7:48	Starting the engine	-
7:49	Reconnecting the equipment	Reconnecting the diagnosis app
7:50	Departing for the destination	-
7:59	Arriving at the destination	Stopping the car
8:00	Stopping the engine	Stopping the engine and disconnecting all equipment and the app

※ RFCOMM(Radio Frequency Communications) : *protocol is a simple set of transport protocols, made on top of the L2CAP protocol, providing emulated RS-232 serial ports*

HCI Snoop Log Analysis based on Specific User Behavior

❖ Packet Analysis for Specific User Behavior(Cont.)

2. Drive

PID Type	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
SAE Standard 7E8h, 7E9h, Etc.	Data Length (3~6byte)	Request Service Code +0x40	PID Code	Data 0	Data 1	Data 2	Data 3	Not used (may be 00h or 55h)

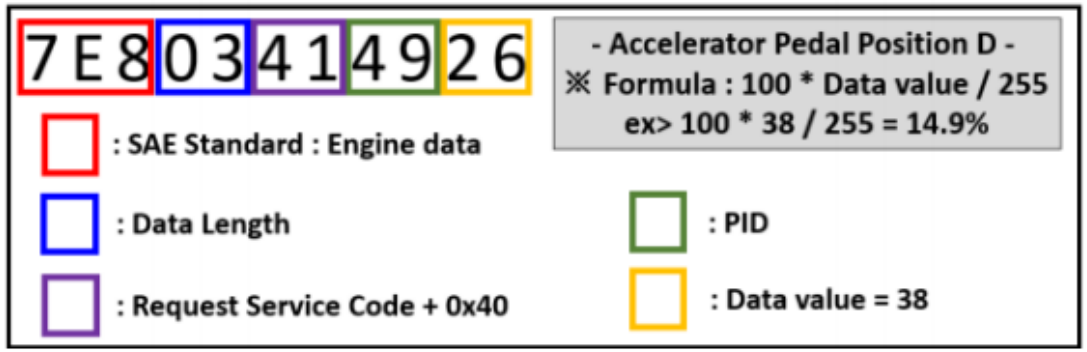


Fig. 2. Raw Data Structure and Example of Raw Data Analysis

- PID Type -> 7E8(Engine Data), 7E9(Transmission Data)
- Data Length : 3 ~ 6 byte
- Request Service Code : 01(Show current Data)

TABLE 5. EXAMPLES OF VEHICLE STATUS INFORMATION

OBD-II PID	Description of PID	Raw Data	Description of Raw Data(data type/Data value)
010D	Vehicle Speed Sensor	7E803410D03	Engine data/3km
0104	Calculated LOAD Value	7E80341043A	Engine data/ 22.7 %
010F	Intake air Temperature	7E803410F56	Engine data/46℃
0149	Accelerator Pedal Position D	7E803414926	Engine data/14.9%
0111	Absolute Throttle Position	7E80341113F	Engine data/20%
010B	Intake Manifold Absolute Pressure	7E803410B22	Engine data/34kPa
010C	Engine RPM	7E904410C0F90/ 7E804410C0F90	Transmission data/996min-1

HCI Snoop Log Analysis based on Specific User Behavior

❖ Packet Analysis for Specific User Behavior(Cont.)

3. Stop and turn off engine

- Case 1) After parking, turn off its engine
  - ACC mode switching (with vehicle spare power)
- Case 2) After parking, unplug the OBD scanner and turn off

TABLE 1. SCENARIO INCLUDING SPECIFIC USER BEHAVIOR

Time	Scenario	Description
	Departure	
pm 7:26	Starting the engine	-
7:29	Connecting Required Equipment	Plugging in OBD scanner to OBD port and connecting with the diagnosis app
7:30	Departing for the stopover	-
7:43	g at the over	Stopping the car
7:44	Stopping the engine	Stopping the engine without unplugging the OBD scanner
7:48	Starting the engine	-
7:49	Reconnecting the equipment	Reconnecting the diagnosis app
7:50	Departing for the destination	-
7:59	g at the nation	Stopping the car
8:00	Stopping the engine	Stopping the engine and disconnecting all equipment and the app

Case 1

Case 2

Pattern

Pattern

TABLE 6. EXAMPLE OF PACKETS GENERATED WHEN THE VEHICLE IS TURNED OFF

Case 1 : Turn off the vehicle without unplugging the scanner				
Time	*Source*	*Destination*	*Protocol/Profile*	*Info*
pm 07:44 :13.21	Galaxy S8	InfocarOH	L2CAP(Protocol)	Disconnection Request
07:44 :13.34	InfocarOH	Galaxy S8	L2CAP(Protocol)	Disconnection Response
07:44 :13.52	Galaxy S8	InfocarOH	L2CAP(Protocol)	Connection Request
…… (Same as Table 3)				
07:44 :14.62	Galaxy S8/ InfocarOH	InfocarOH / Galaxy S8	SPP(Profile)	Initializing Using AT Command

Case 2 : Unplug the scanner before turning off the vehicle				
Time	*Source*	*Destination*	*Protocol/Profile*	*Info*
pm 08:00 :41.69	Galaxy S8	InfocarOH	L2CAP(Protocol)	Disconnection Request
…… (Same as Case 1)				
08:00 :43.09	Galaxy S8/ InfocarOH	InfocarOH / Galaxy S8	SPP(Profile)	Initializing Using AT Command
08:00 :50.07	InfocarOH	Galaxy S8	SPP(Profile)	"UNABLE TO CONNECT\r\r>"

05 Discussion

Discussion

❖ Result of Analysis

User Behavior	Pattern of Packets	Profile/Protocol	Data	Practical Use
Plug and connect OBD Scanner	1. Connection Request 2. Connection Response(Pending) 3. Connection Response(Success) 4. Initializing Using AT Command	L2CAP(Protocol) SPP(Profile)	--	Determine whether or not the vehicle driver uses the OBD scanner and the approximate boarding time
Drive	1. “010D\r” [PID] 2. UIH Channel=1 UID 3. “7E803410D03\r” [Raw Data] 4. “\r>”	SPP(Profile) RFCOMM(Protocol)	Real time Vehicle Status Info(Speed, Accel Pedal Position, Engine RPM, etc.)	Vehicle sensor data can provide information about the state of the vehicle when a vehicle accident occurs Ex) Vehicle speed, Sudden acceleration
Stop and turn off Engine	Case1) 1. Disconnection Request 2. Disconnection Response 3. Connection Request 4. Connection Response(Pending) 5. Connection Response(Success) 6. Initializing Using AT Command	L2CAP(Protocol) SPP(Profile)	--	Determine the time to get off the vehicle & shut off time due to a vehicle accident
	Case2) 1. Disconnection Request 2. Initializing Using AT Command 3. “UNABLE TO CONNECT\r\r>”		--	

06 Conclusion & Future Work

Conclusion & Future Work

❖ Generate, collect and analyze packets

- Data Generation : pre-planned scenario includes specific driver's behavior
  - Connecting OBD scanner to mobile car diagnosis app
- Data Collection : Bluetooth HCI Snoop Log
- Data analysis
  - Specific driver's behavior
    - Pattern of sent/received packets
    - used protocols and profiles
    - Packet's data field
- Useful for investigating real vehicle accident and identifying its cause

Conclusion & Future Work

❖ Future Work

- Differences from the actual vehicle accident environment
  - Collecting real-world vehicle accident cases and common car accident scenarios
- Plan to analyze Bluetooth HCI Snoop logs
  - Collected accident cases and scenarios

Thank you

- Q&A -