Automotive Ethernet Fuzzing:
From Purchasing ECU to SOME/IP Fuzzing

Jonghyuk Song, Soohwan Oh, Woongjo Choi

AUTOCRYPT
Who are we?

• Jonghyuk Song (jhsong@autocrypt.io)
  • AUTOCRYPT Red Team leader, automotive cyber security researcher
  • Offensive security researcher, CTF player (Defcon CTF Finalist 2009, 2011, 2017)

• Soohwan Oh
  • Automotive engineer and security tester at AUTOCRYPT
  • Experts for fuzzing test and issue analysis on the in-vehicle networks

• Woongjo Choi
  • AUTOCRYPT Blue Team leader, automotive security tester
  • Software test expert : vehicle, mobile, application processor, ultrasound system, etc.
This talk is for

• How to setup test environment with real ECUs?

• How to do Automotive Ethernet fuzzing on the ECU?

→ A practical guide to SOME/IP fuzzing with the real ECUs.
Advent of Automotive Ethernet

Higher data bandwidth is required for ADAS, infotainment, ...

CAN is not enough to handle it.

Automakers have started to adopt Automotive Ethernet.
  BMW X3, X4, X5, i3, i8, ...
  Jaguar XJ, XF, ...
  Volkswagen Passat

In-Vehicle Infotainment

Diagnostics

Camera System for ADAS
(Advanced Driver Assistance System)
### Automotive Ethernet Protocols

<table>
<thead>
<tr>
<th>Layer 1</th>
<th>Layer 2</th>
<th>Layer 3</th>
<th>Layer 4</th>
<th>Layer 5-7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automotive Ethernet Physical Layer</td>
<td>IEEE Ethernet MAC + VLAN (802.1Q)</td>
<td>AVB</td>
<td>AVB</td>
<td>IEEE 802.1AS</td>
</tr>
</tbody>
</table>

#### Protocols

- **SOME/IP** (Scalable service-Oriented MiddlewarE over IP)
  - Control communication protocol between ECUs. (RPC, event notifications)

- **DoIP** (Diagnostics Over IP)
  - Automotive diagnostic protocol based on IP

---

*Dr. Lars Völker, BMW Group, Communication Protocols for Ethernet in the vehicle, 2013*
Automotive Ethernet Fuzzing?

Automotive Ethernet has not yet been tested enough.

→ Let’s do SOME/IP fuzzing!

→ Is it essential buying a car?
Let’s test on ECU

It’s expensive to buy a whole car for every single test.
→ Let’s hack an ECU first!

How?
  Where can we buy?
  Which ECU is proper for test?
  Then how to setup?

* ECU: Electric Control Unit
Procedures

1. Select ECU
2. Buy ECU
3. Setup test environment
4. Network configuration for SOME/IP
5. Discovery SOME/IP services
6. Fuzzing
Select ECU
ECU Survey & Selection

• We chose a BMW iX head unit ECU.
  • Most head unit ECU support Automotive Ethernet.

• Why BMW?
  • Automotive Ethernet industry leader
    • First automaker to use Ethernet in a production vehicle.
      (X5 model uses Ethernet in camera system at 2013)
  • BMW provides ECU info on TIS (Technical Information System)
    • wiring diagrams, pin map, connector, basic info, ...
BMW Technical Information System

Paid service

VIN number is required to search

www.bmwtis.com
Google serves various VIN numbers

2022 BMW iX, new, $97,570
See the new 2022 BMW iX priced at $97,570. The iX with a VIN of [redacted] located in Chandler, AZ, has 0 miles, is Storm Bay Metallic with a ... [redacted]

Where is the VIN number on a BMW? - YOUCANIC
2022. 6. 2. — BMW Vehicle Identification Number (VIN) can be found printed in several locations on the vehicle.

BMW - Vehicle Identification Number
Additionally, you can find the VIN at various locations on your car. Frequently you can find your VIN number in one of the following locations: on the driver’s ...

New 2023 BMW iX M60 [redacted]
See the details of the new BMW iX M60 with VIN [redacted].

2022 BMW iX
$91,245 • new

Basics
Condition: New
Interior Color: Black
Exterior Color: Sophisto
Drivetrain: All-Wheel Drive
Transmission: Automatic
Fuel: Electric
Engine: Electric
VIN: [redacted]
Stock Number: CK57882
Miles: 0
Wiring Diagram

Wiring diagram shows the information for setup the ECU. power, ground, ETH, ...

We can find out that the head unit supports Ethernet!
Buy ECU
Gain ECUs from real automakers

• In South Korea,
  • BMW
    • Some service centers sell ECUs.
    • We searched the BMW service centers from BMW Korea homepage ➔ Phone call ➔ Buy
  • Tesla
    • Service centers sell ECUs but not as an individual unit.
    • The ECUs should be installed in the car.
  • Hyundai
    • Hyundai Mobis agency sells ECUs.
Used ECUs are able to obtain from eBay.
Setup Test Environment
Conversion between Automotive Ethernet and standard Gigabit Ethernet
Laptop

Media Converter

ECU

Power Supply

Display
Connect power to ECU

- Red (+)
  - Power (12V)
  - Ethernet wake up (12V)

- Black (-)
  - Ground

You can check the pin map from TIS.
Setup display
Power On!
ECU – Media Converter – Laptop

SFP port

Rj45 port for control page
Network Configuration for SOME/IP
Virtual LAN (VLAN) for Automotive Ethernet

- Automotive Ethernet uses VLAN to separate in-vehicle network logically.
Virtual LAN (VLAN) for Automotive Ethernet

Traditional Ethernet frame

<table>
<thead>
<tr>
<th>Dst addr</th>
<th>Src addr</th>
<th>Length/Type</th>
<th>Data</th>
<th>CRC/FCS</th>
</tr>
</thead>
</table>

Ethernet frame with VLAN tag

<table>
<thead>
<tr>
<th>Dst addr</th>
<th>Src addr</th>
<th>VLAN Tag</th>
<th>Length/Type</th>
<th>Data</th>
<th>CRC/FCS</th>
</tr>
</thead>
</table>

- **TPID**
- **PCP**
- **DEI**
- **VID**
How to know IP and VLAN ID?

Wireshark shows IP and VLAN ID.
(Some USB to Ethernet adapter doesn’t show VLAN ID. If you can’t see VLAN ID, it will be a good idea to change the adapter.)
Setup IP and VLAN

(Windows) Ethernet properties → Networking → Configure → Advanced
- “PacketPriority & VLAN” → “Packet Priority & VLAN Enable”
- “VLAN ID” → “Value” → vlan id

Select any IP in same subnetwork
Discovery SOME/IP Services
SOME/IP Service Discovery (SD)

Several SOME/IP services are running on the different ports at ECU. To fuzz a SOME/IP service, we should know the port and the service ID.

- Check “OFFER” message
ECU periodically broadcast ”OFFER” containing the service IDs and port numbers.

- Send “FIND” message
ECU will respond with ”OFFER”.

“OFFER” message includes service ID and port.
Let’s Do Fuzzing!
Fuzz Data Generation

Service ID [16 Bit]  0 [1 Bit]  Method ID [last 15 Bit]

Client ID [16 Bits]  Session ID [16 Bits]

Message ID (Service ID / Method ID) [32 bit]

Length [32 bit]

Request ID (Client ID / Session ID) [32 bit]


Payload [variable size]

<table>
<thead>
<tr>
<th>Number</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x00</td>
<td>REQUEST</td>
<td>A request expecting a response (even void)</td>
</tr>
<tr>
<td>0x01</td>
<td>REQUEST_NO_RETURN</td>
<td>A fixed request expecting no response</td>
</tr>
<tr>
<td>0x02</td>
<td>NOTIFICATION</td>
<td>A notification/event callback expecting no response</td>
</tr>
<tr>
<td>0x03</td>
<td>RESPONSE</td>
<td>The response message</td>
</tr>
<tr>
<td>0x04</td>
<td>ERROR</td>
<td>The response containing an error</td>
</tr>
<tr>
<td>0x05</td>
<td>TP_REQUEST</td>
<td>A TP request expecting a response (even void)</td>
</tr>
<tr>
<td>0x06</td>
<td>TP_REQUEST_NO_RETURN</td>
<td>A TP fixed request expecting no response</td>
</tr>
<tr>
<td>0x07</td>
<td>TP_NOTIFICATION</td>
<td>A TP notification/event callback expecting no response</td>
</tr>
<tr>
<td>0x08</td>
<td>TP_RESPONSE</td>
<td>The TP response message</td>
</tr>
<tr>
<td>0x09</td>
<td>TP_ERROR</td>
<td>The TP response containing an error</td>
</tr>
</tbody>
</table>

How to know ECU’s state?

- **Return codes**
  - We can guess the ECU’s state and error handling logics.

- Check response on the valid SOME/IP-SD message
  - no response from ECU → SOME/IP service is dead?

- Check periodic SOME/IP-SD messages from the ECU
  - no longer “OFFER” messages transmission from the ECU → SOME/IP service is dead?

<table>
<thead>
<tr>
<th>Return Codes</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x00</td>
<td>E_OK</td>
</tr>
<tr>
<td>0x01</td>
<td>E_NOT_OK</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>0x06</td>
<td>E_TIMEOUT</td>
</tr>
<tr>
<td>0x09</td>
<td>E_MALFORMED_MESSAGE</td>
</tr>
</tbody>
</table>
Implementation: Scapy supports SOME/IP!

**SOME/IP and SOME/IP SD messages**

Creating a SOME/IP message

This example shows a SOME/IP message which requests a service 0x1234 with the method 0x421. Different types of SOME/IP messages follow the same procedure and their specifications can be seen here: [http://www.some-ip.com/papers/cache/AUTOSAR_TR_SomeIpExample_4.2.1.pdf](http://www.some-ip.com/papers/cache/AUTOSAR_TR_SomeIpExample_4.2.1.pdf).

Load the contribution:

```python
load_contrib('automotive.someip')
```

Create UDP package:

```python
u = UDP(sport=30509, dport=30509)
```

Create IP package:

```python
i = IP(src="192.168.0.13", dst="192.168.0.10")
```

Create SOME/IP package:

```python
sip = SOMEIP()
sip.iface_ver = 0
sip.proto_ver = 1
sip.msg_type = "REQUEST"
sip.recode = "E_OK"
sip.srv_id = 0x1234
sip.method_id = 0x421
```

Add the payload:

```python
sip.add_payload(Raw ("Hello"))
```

Stack it and send it:

```python
p = i/u/sip
send(p)
```

You can insert fuzz data here
Results

• BMW
  • Still, we do fuzzing ... 
  • Haven’t found anything yet

• But we found some crashes of other OEM’s ECUs.
  • we can’t disclose it because of NDA.
Conclusion

• You can do car hacking without a whole car itself.
  • You can buy, setup and test ECUs.

• Let’s do fuzzing Automotive Ethernet.
  • It’s an early stage. There’s still much to fuzz.
  • We will try DoIP.