Wireshark Packet Dissectors for the Latest V2X Message Protocol

Automotive Linux Summit 2017

May 31-June 2, 2017, Tokyo, Japan

Steve Kwon | CEO

Contents & Today's Goal

- **V2X (Vehicle to Everything)**
- **V2X Protocol Standards**
  - Protocol Overview
  - ASN.1 and Protocol Encapsulation
- **Wireshark Patches for the Latest V2X Protocols**
  - Logical Link Control
  - IEEE 1609.3 - WAVE Short Message
  - IEEE 1609.2 - WAVE Security Service
  - IEEE 1609.3 - WAVE Service Advertisement
  - SAE J2735 Mar. 2016 - DSRC Message Set Dictionary
- **Wireshark V2X Message Dissector Demo**
- **Conclusion & Summary**

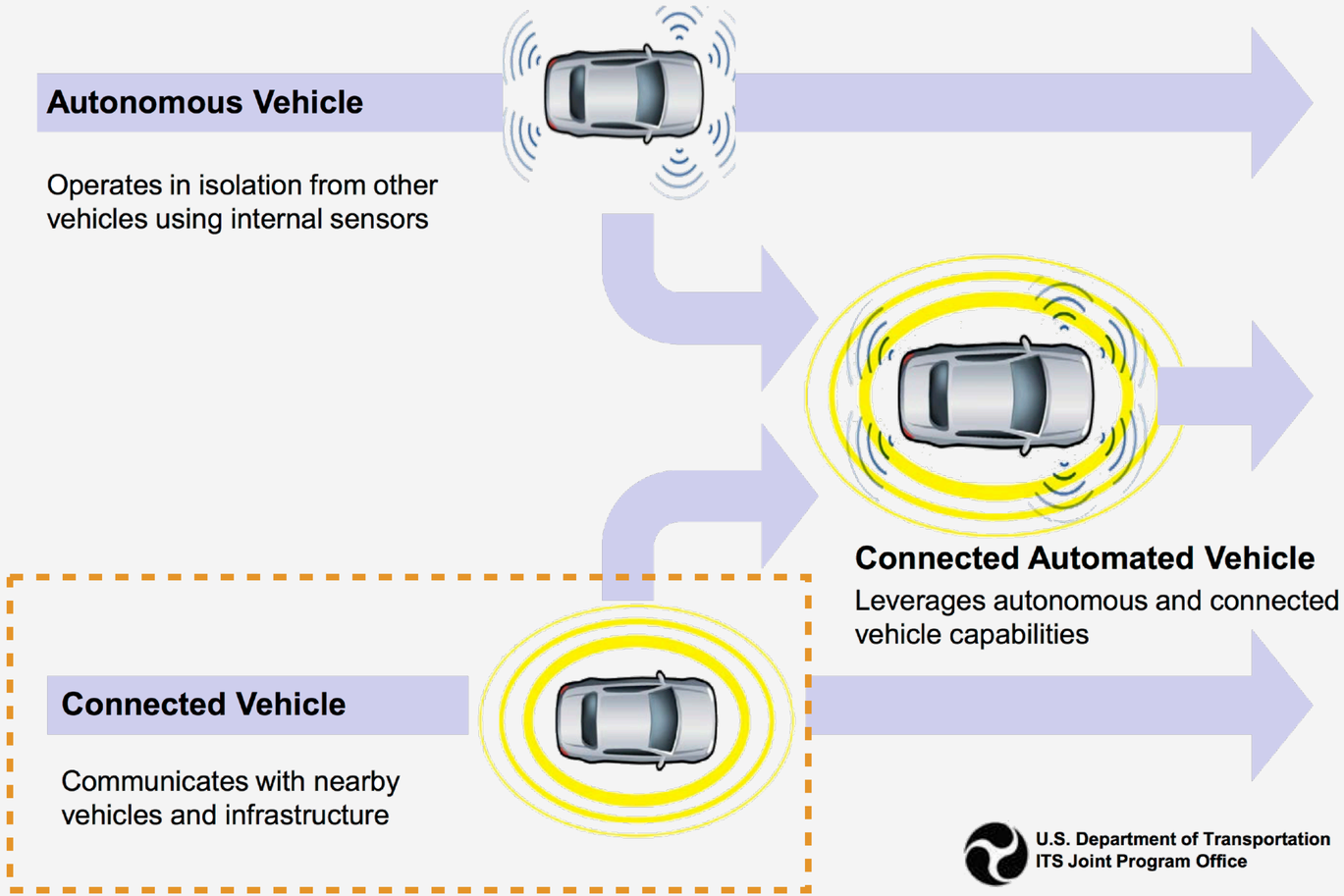
Today's Goal

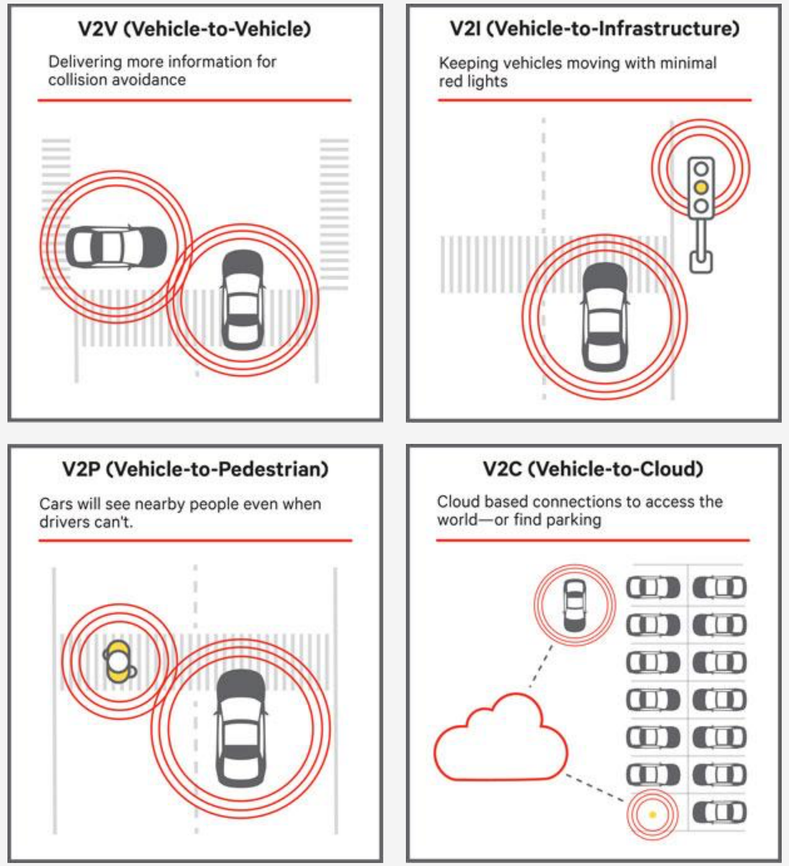
Understand the LatestV2X Protocols

and How to Build own Analyzer

V2X (Vehicle to Everything)

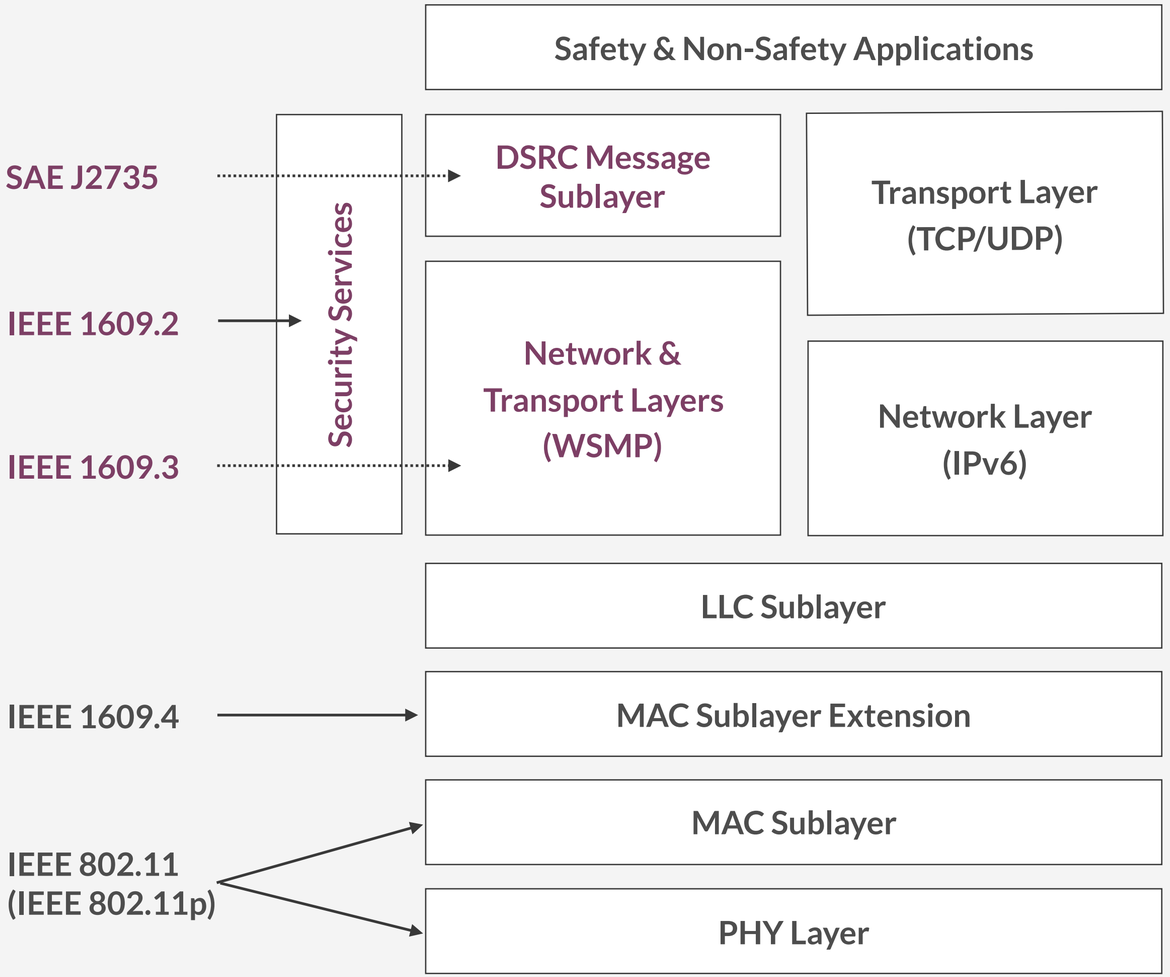
- **Radio Communication (DSRC) + Positioning System (GNSS/GPS) + Vehicle Status (transmission, steering angle, brake, …)**
- **U.S. is progressing Vehicle-to-Vehicle (V2V) communications Mandate for New Light Vehicles**
  - Notice of Proposed Rulemaking (NPRM) published at 01/12/2017, Comments Close at 04/12/2017





https://www.qualcomm.com/news/snapdragon/2015/06/04/snapdragon-automotive-solutions-connected-car-platforms-all-types-vehicle

V2X Protocol Standards - Overview



Standard	Latest	Usage
SAE J2945/9	2017	Vulnerable Road User Safety Message Minimum Performance Requirements
SAE J2945/1	2016	On-Board System Requirements for V2V Safety Communications
SAE J2735	**2016**	**DSRC Message Set Dictionary**
IEEE 1609.2	**2016**	**Security Services for Applications and Management Messages**
IEEE 1609.3	**2016**	**Networking Services**
IEEE 1609.4	2016	Multi-Channel Operation
IEEE 1609.12	2016	Identifier Allocations
IEEE 802.11 (IEEE 802.11p)	2016	WAVE PHY and MAC

V2X Protocol Standards - ASN.1* and Protocol Encapsulation

*http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html

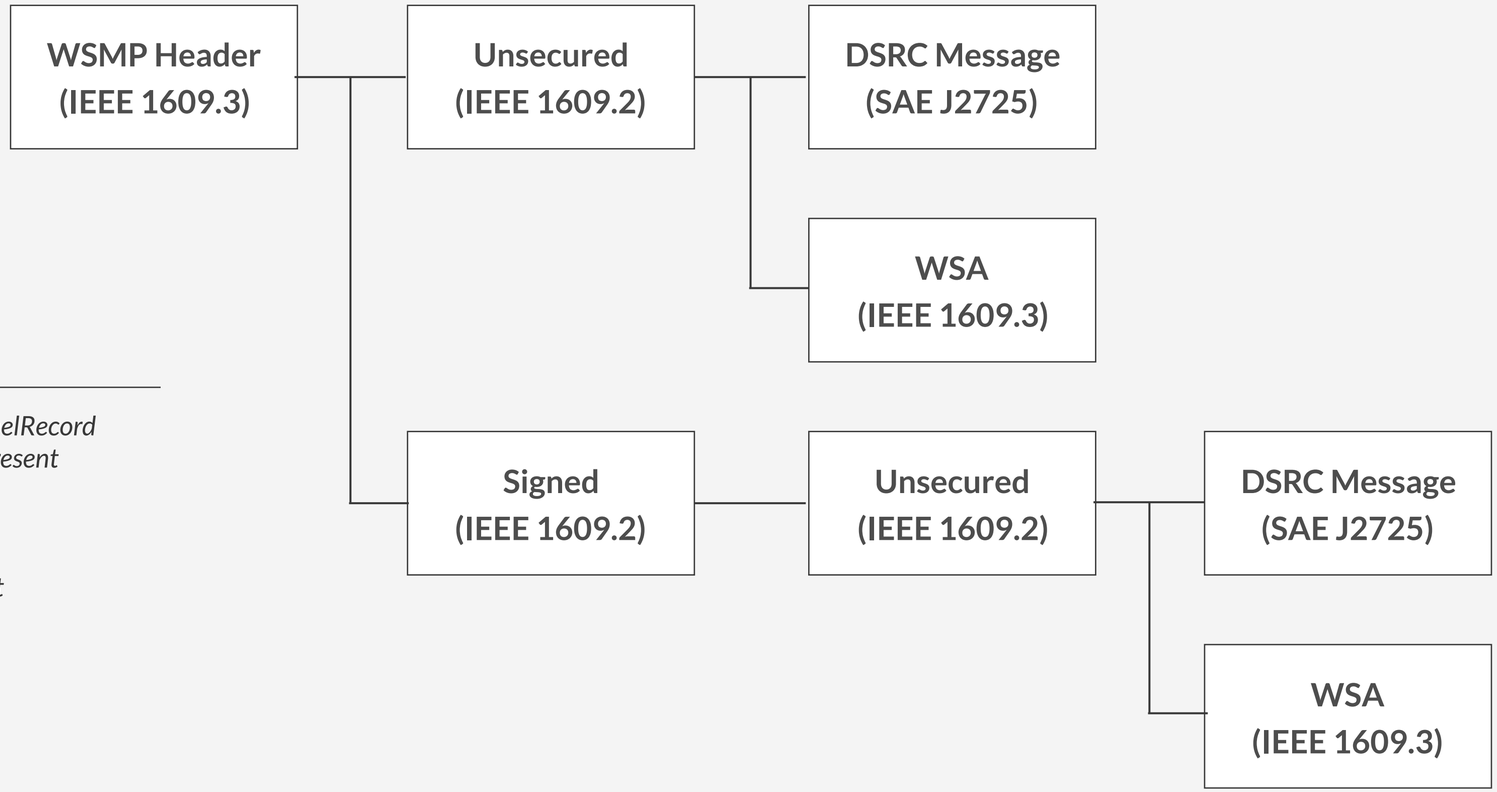
- IEEE 1609.3 and SAE J2735 use UPER (Unaligned Packet Encoding Rule)
- IEEE 1609.2 uses COER (Canonical Octet Encoding Rule)

  => It is almost impossible to read or understand without a packet analyzer

40CBAA3A 5108A512 …

0	*No extension values present in PersonnelRecord*
1	*Bitmap bit = 1 indicates "children" is present*
0	*No extension values present in "name"*
0	*Length is within range of extension root*
0000.11	*Length of name.givenName = 4*
001011 .101010 10.0011 1010.01	*name.givenName = "John"*
010001	*name.initial = "P"*
….	*…*

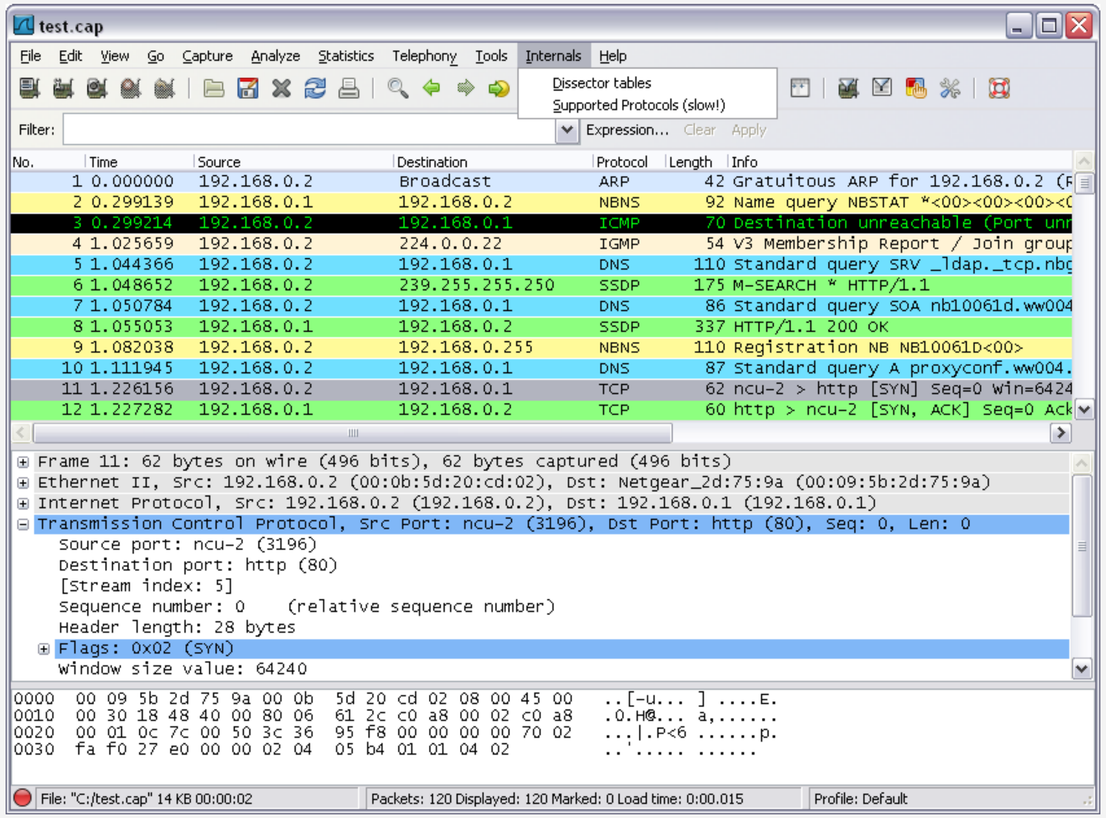
< Example of UPER decoding from ISO/IEC 8825-2: 2015 PER >



< V2X Protocol Encapsulation >

Wireshark

Wireshark is the world's foremost and widely-used network protocol analyzer. It lets you see what's happening on your network at a microscopic level and is the de facto (and often de jure) standard across many commercial and non-profit enterprises, government agencies, and educational institutions. Wireshark development thrives thanks to the volunteer contributions of networking experts around the globe and is the continuation of a project started by Gerald Combs in 1998. - https://www.wireshark.org



WIRESHARK

- *Deep inspection of hundreds of protocols*
- *Live capture and offline analysis*
- *Multi-platform : Windows, Linux, OSX, …*
- *Captured network data can be browsed via GUI or console*
- *The most powerful display filters*
- *Live data can be read from Ethernet, IEEE 802.11, …*
- *Decryption support: IPsec, SSL/TLS, WPA/WPA2, …*
- *Coloring rules can be applied to the packet lis*
- *Output can be exported to XML, CSV, plain text, …*
- ***Dissector from ASN.1 (BER/PER, Aligned/Unaligned, asn2wrs.py*)***

*https://wiki.wireshark.org/Asn2wrs

Wireshark Patches for the Latest V2X Protocols

GitHub - https://github.com/wayties/wireshark - “v2x” branch, based on 2.2.6 tag

packet-llc.c: tweak for IEEE Std. 802, EPD

IEEE Std. 1609.3 uses IEEE Std.802 EPD (EtherType Protocol Discrimination), so it needs a little tweak to call a dissector by ethertype without SNAP header

packet-wsa.c: add support for WAVE Service Advertisement

WAVE Service Advertisement (WSA) dissector generated from ASN.1
Ref. IEEE Std. 1609.3 - 2016

packet-wsm.c: add support for WAVE Short Message

WAVE Short Message (WSM) dissector generated from ASN.1 files
Ref. IEEE Std. 1609.3 - 2016

packet-ieee1609.c: call WSA dissector from IEEE 1609.2

When WSA header version is 0x3X, it will call WSA dissector
Ref. IEEE Std. 1609.3/2 - 2016

packet-ieee1609.c: add support IEEE 1609.3 / 2

IEEE 1609.3 WSM dissector and IEEE 1609.2 UnsecuredData and SignedData(Partial) by direct implementation
Ref. IEEE Std. 1609.3/2/12 - 2016

packet-j2735.c: add support for SAE J2735 DSRC Message Set

packet-j2735.c: add support for SAE J2735 DSRC Message Set

SAE J2735 dissector generated from ASN.1 file and it will be called by IEEE 1609.2 Unsecured/SignedData
Caution: It does't not include any SAE J2735 related contents,
Ref. SAE J2735 DSRC Message Set Dictionary - Mar. 2016

Logical Link Control

IEEE 802.11 QoS	DSAP= 0xAA	SSAP= 0xAA	Control= 0x03	Protocol ID=	Ethertype

LPD (LLC Protocol Discrimination)

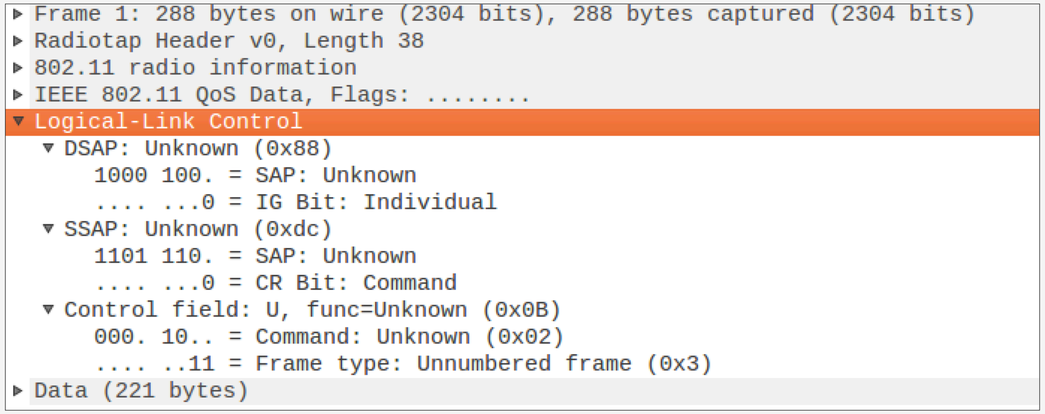
Ethertype =
0x88DC : WSMP, 0x86DD : IPv6

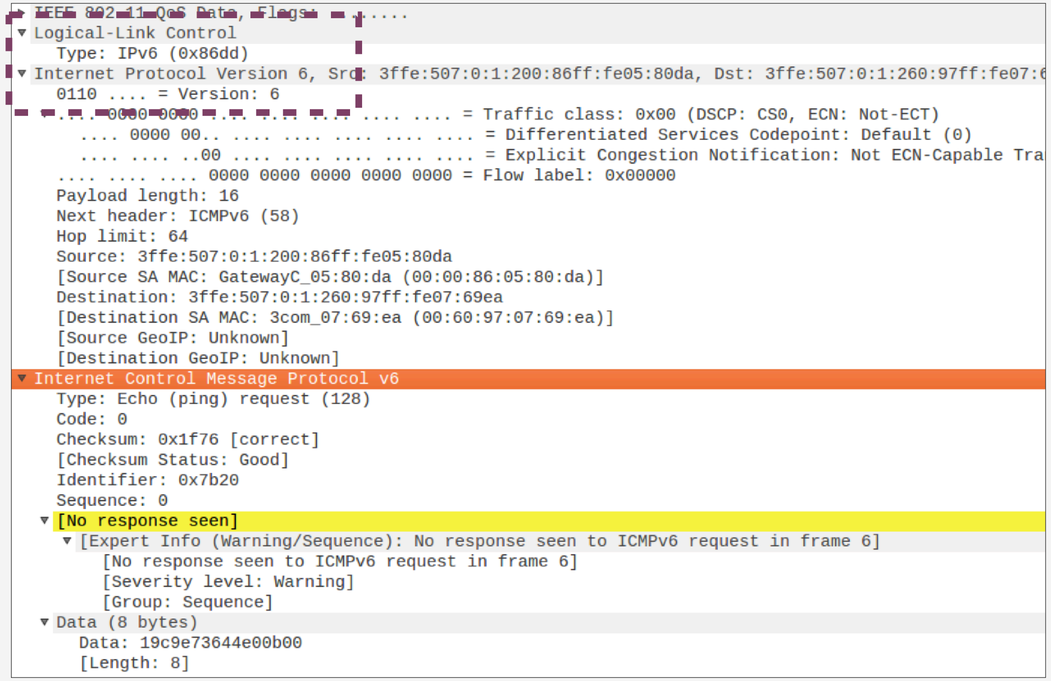
IEEE 802.11 QoS	Ethertype

EPD (EtherType protocol discrimination)

```
is_snap = (dsap == SAP_SNAP) && (ssap == SAP_SNAP);
...
/* tweak for IEEE Std. 802, EPD(EtherType Protocol Discrimination) */
if (!is_snap) {
  etype = tvb_get_ntohs(tvb, 0);
  next_tvb = tvb_new_subset_remaining(tvb, 2);
    if (dissector_try_uint(ethertype_subdissector_table, etype, next_tvb, pinfo, tree)) {
      proto_tree_add_uint(llc_tree, hf_llc_type, tvb, 0, 2, etype);
      return tvb_captured_length(tvb);
    }
...
```

epan/dissectors/packet-llc.c
- Try ethertype dissector without SNAP header





IEEE 1609.3 - WAVE Short Message (1/2)

Version	PSID	Extension Fields	WSMP WAVE Element ID	Length	WSM Data
1 byte	4 bytes	variable	variable	2 bytes	variable

WSMP-N-Header					WSMP-T-Header*		
Subtype	WSMP-N-Header Option Indicator	WSMP Version	WAVE Information Element Extension	TPID	PSID	WSM Length	WSM Data
4 bits	1 bits	3 bits	variable	1 byte	variable	variable	variable

*WSMP-T-Header format for TPID = 0

```
▼ Wave Short Message Protocol(IEEE P1609.3)
    Version: 11
    PSID: 0x00000003
    Transmit Power: 148
    Channel: 172
    Data Rate: 12
    WAVE element id: WSMP (128)
    WSM Length: 52995
▼ [Malformed Packet: WSMP]
  ▼ [Expert Info (Error/Malformed): Malformed Packet (Exception occurred)]
      [Malformed Packet (Exception occurred)]
      [Severity level: Error]
      [Group: Malformed]
```

configure.ac
epan/dissectors/CMakeLists.txt
epan/dissectors/Makefile.am ***// Add "wsm" to Config & Makefiles***
epan/dissectors/asn1/CMakeLists.txt
epan/dissectors/asn1/Makefile.am

*epan/dissectors/asn1/wsm/CITSapplMgmtIDs.asn**
epan/dissectors/asn1/wsm/CMakeLists.txt
epan/dissectors/asn1/wsm/Makefile.am
epan/dissectors/asn1/wsm/packet-wsm-template.c
epan/dissectors/asn1/wsm/packet-wsm-template.h ***// Add "wsm" dissector***
*epan/dissectors/asn1/wsm/wee.asn**
*epan/dissectors/asn1/wsm/wsm.asn**
epan/dissectors/asn1/wsm/wsm.cnf

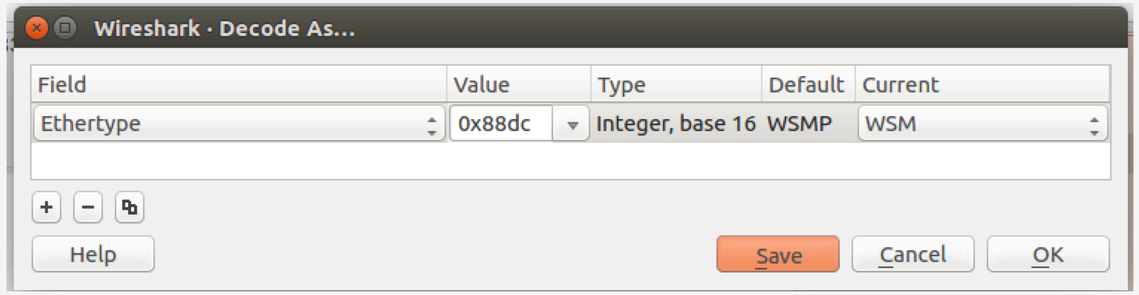
./autogen.sh ***// Generate "wsm" dissector***
cd epan/dissectors/asn1 ***// It will generate following dissector files***
make ***epan/dissectors/packet-wsm.c***
epan/dissectors/packet-wsm.h

cd -
./configure ***// reconfigure and rebuild wireshark***
make

*https://github.com/certificationoperatingcouncil/TCI_ASN1/tree/master/TCI%20Interface/ASN1/1609dot3

IEEE 1609.3 - WAVE Short Message (2/2)

1. **Fix ASN.1 file**
2. **Assign Root PDU : ShortMsgNpdu**
   - Use #.PDU of “wsm.cnf”
3. **Process Object Identifier of “Extension” Tag**
   - Add id variable at “packet-wsm-template.c”
   - Use #.FN_PARS, #.FN_BODY of “wsm.cnf”
   - Add dissect_per_open_type with dissector for id
4. **Fix “unused function” error**
   - Use #.OMIT_ASSIGNMENT of “wsm.cnf”



Menu -> Analyzer -> Decode As…

Assign Ethertype 0x88dc to new “wsm” dissector

```
▼ IEEE 1609.3 - WAVE Short Message
  ▼ ShortMsgNpdu
    ▼ subtype: nullNetworking (0)
      ▼ nullNetworking
          version: c-shortMsgVersionNo (3)
        ▼ nExtensions: 3 items
          ▼ Item 0
            ▼ ShortMsgNextension
                extensionId: c-TxPowerUsed80211 (4)
              ▼ value
                  txpower80211: 20
          ▼ Item 1
            ▼ ShortMsgNextension
                extensionId: c-ChannelNumber80211 (15)
              ▼ value
                  channelNumber80211: 172
          ▼ Item 2
            ▼ ShortMsgNextension
                extensionId: c-DataRate80211 (16)
              ▼ value
                  dataRate80211: 12
    ▼ transport: bcMode (0)
      ▼ bcMode
        ▼ destAddress: content (0)
            content: 32
      body: 038081cb001280c708000100102fea7732d8c9cc49934880...
▼ Data (207 bytes)
    Data: 038081cb001280c708000100102fea7732d8c9cc49934880...
    [Length: 207]
```

IEEE 1609.2 - WAVE Security Service

Direct Implementation of WSM dissector

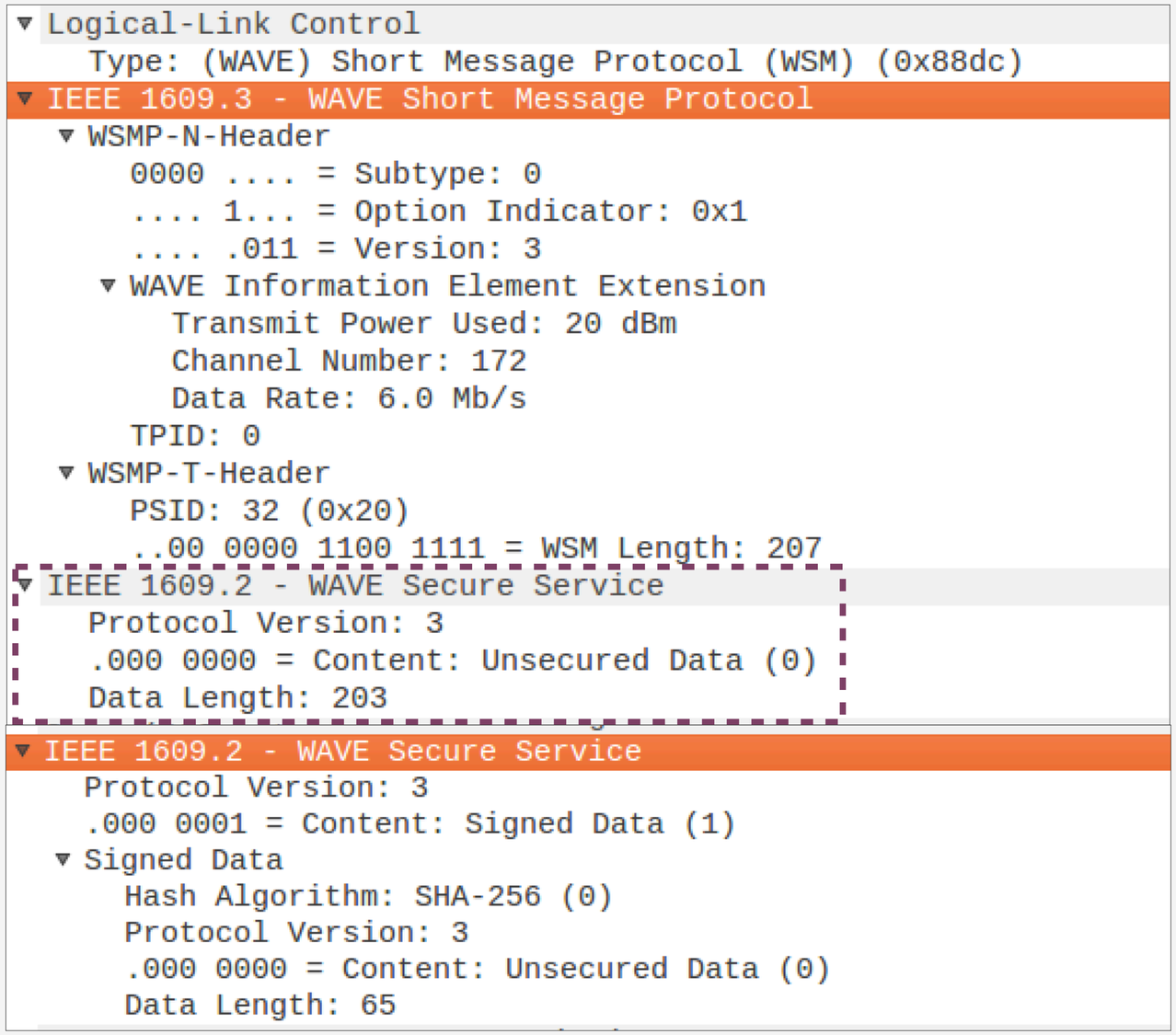
- “ieee1609” dissector
- epan/dissectors/packet-ieee1609.c
- Naming and hierarchy that described in Standard

Direct Implementation of 1609.2 dissector

- Currently, Wireshark does not support OER
- “ieee1609dot2” dissector
- UnsecuredData : fully support
- SignedData : UnsecuredData and Length only

Menu -> Analyzer -> Decode As…

Assign Ethertype 0x88dc to new “ieee1609” dissector



IEEE 1609.3 - WAVE Service Advertisement

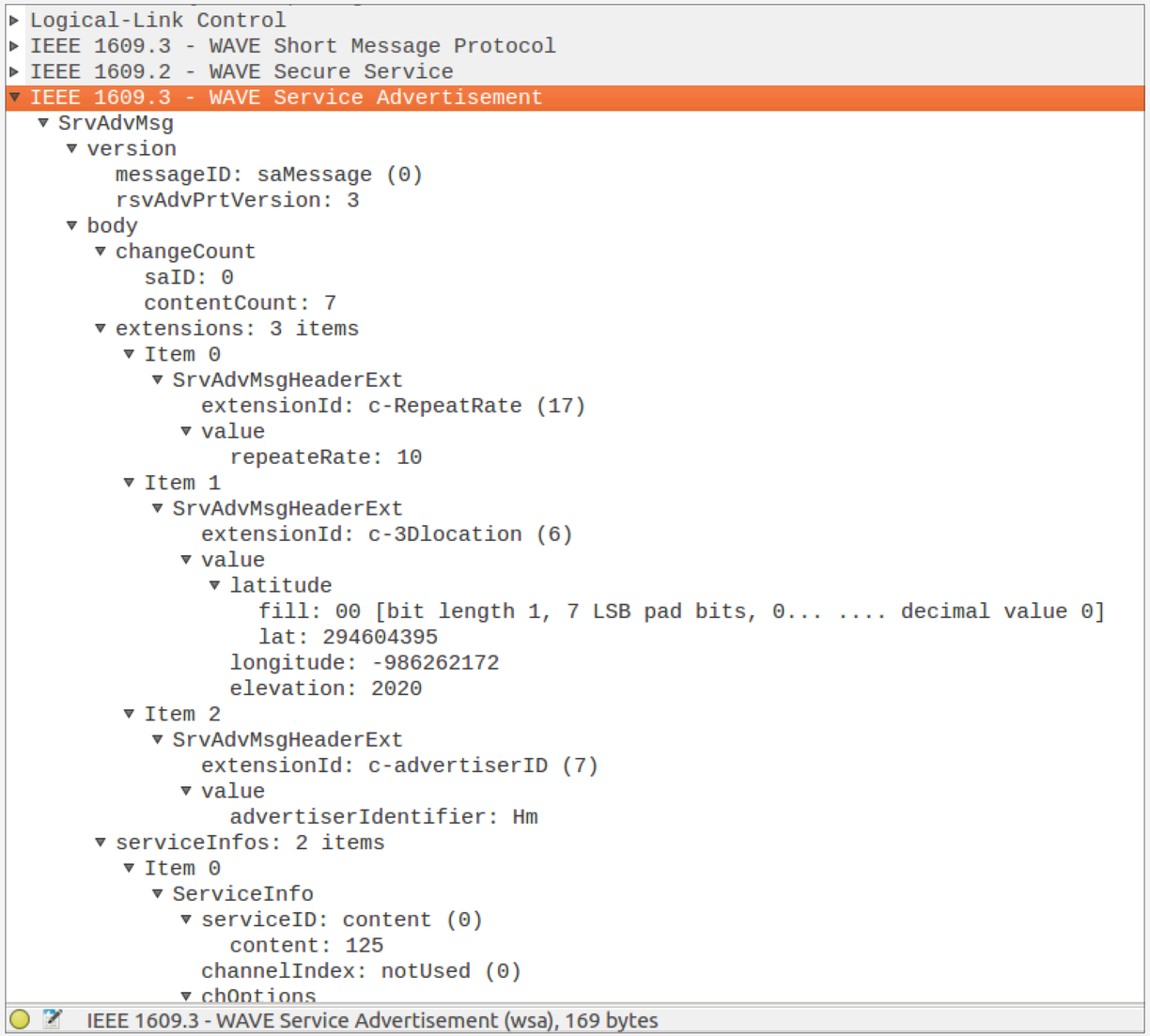
Do work similar to “wsm" dissector using “wsa.asn”

- “ieee1609dot3_wsa” dissector
- Fix ASN.1 file
- Assign Root PDU : SrvAdvMsg
- Process Object Identifier of “Extension” Tag
- Fix “unused function” error
- Call from “ieee1609dot2” dissector

```
next_tvb = tvb_new_subset_length(tvb, offset, datalen);
/* call_data_dissector(next_tvb, pinfo, tree); */
wsa_version = tvb_get_guint8(tvb, offset);

/* When WSA version is matched */
if ((wsa_version & 0xF0) == 0x30)
  call_dissector(ieee1609dot3_wsa_handle, next_tvb, pinfo, tree);
else
  call_data_dissector(next_tvb, pinfo, tree);
```

epan/dissectors/packet-ieee1609.c
- After checking WSA version, call “wsa” dissector



SAE J2735 - Mar. 2016

Prepare J2735 201603 ASN.1 file

- Copyright, not publishable
- Issued: 2016-03-30
- http://standards.sae.org/j2735asn_201603/

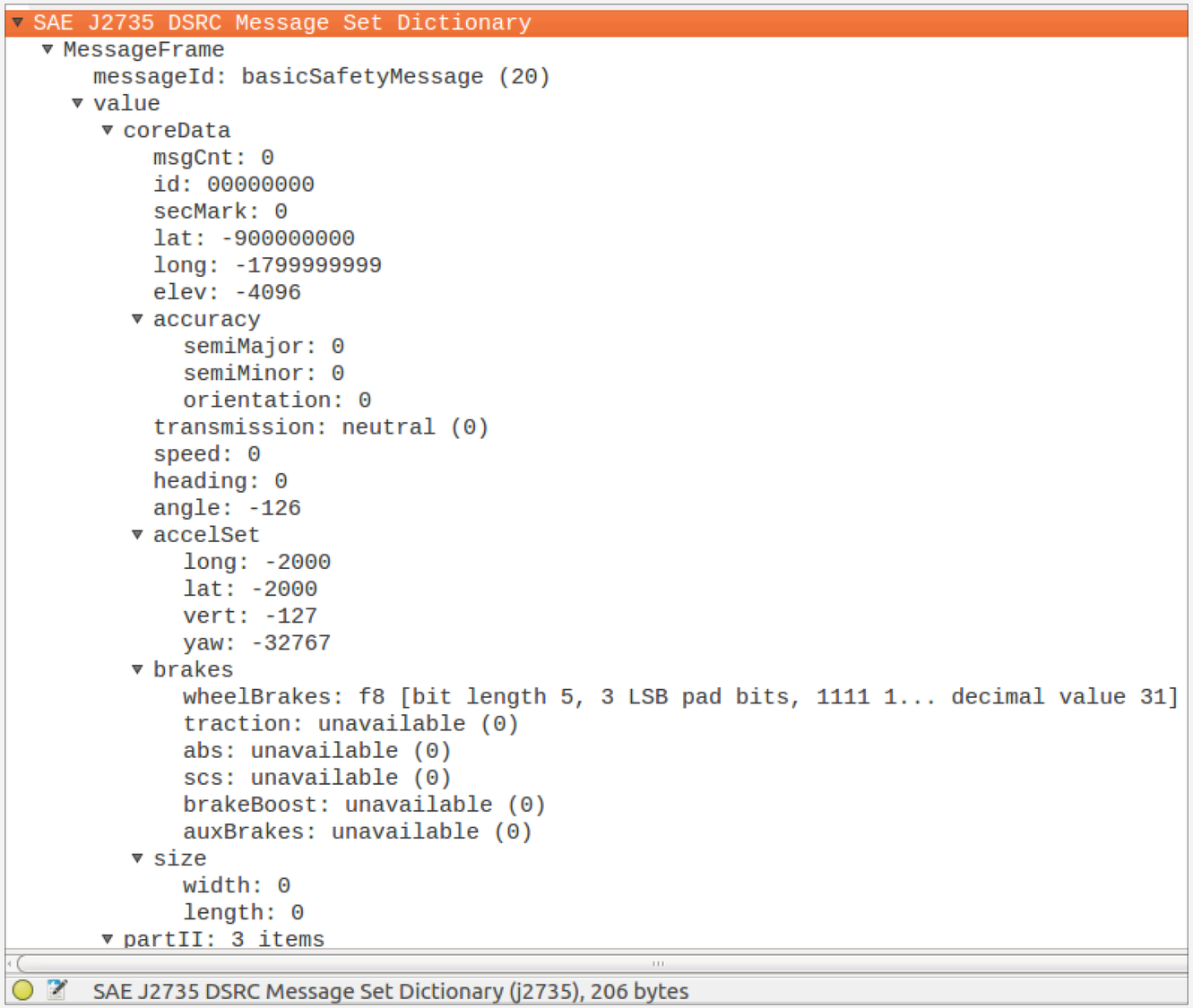
1. Put ASN.1 file into epan/dissectors/asn1/j2735

2. Run c.sh

- It will generate fixed ASN.1 file

3. Do work similar to “wsm" dissector

- Assign Root PDU
- Process Object Identifier Tag
- Fix “unused function” error



Wireshark V2X Message Dissector Demo

DEMO

- MAP (MapData)
- BSM (Basic Safety Message)
- TIM (Traveler Information Message)
- WSA (WAVE Service Advertisement)
- IPv6 Echo Request over IEEE 1609.3

Conclusion & Summary

- **Now, we have opened Wireshark dissector for the latest V2X message protocol with some limitations**

- **DONE**
  - Logical Link Layer
  - IEEE 1609.3 - WAVE Short Message
  - IEEE 1609.2 - WAVE Security Service - Unsecured & Signed (Partial Implementation)
  - IEEE 1609.3 - WAVE Service Advertisement
  - SAE J2735 Mar. 2016 - DSRC Message Set Dictionary

- **TODO**
  - IEEE 1609.2 - WAVE Security Service - Full Direct Implementation
  - or implementation OER (Octet Encoding Rule) support of asn2wrs.py

Thank you

Wayties

One Step Ahead

5F, 320 Gangnam-daero Gangnam-gu,

Seoul, 06252 Republic of Korea

Contact: CEO, Steve Kwon