Submitted by the expert form ITU Document No. ITS/AD-08-08
(8th ITS/AD, 9 March 2016, agenda item 4)

ITU activities on secure vehicle software updates

8th meeting of IWG ITS/AD

9 March 2016

T.Russell Shields, Co-Chair, Collaboration on ITS Communication Standards
Martin Adolph, Programme Coordinator, ITU

Outline

1. Draft Report on secure over-the-air vehicle software updates - operational and functional requirements
2. Draft Standard on secure software update capability for ITS communications devices
3. Other ITS communications related activities in ITU
4. Supplementary material

Report on secure over-the-air vehicle software updates - operational and functional requirements

- Objectives:
  - The principal objective of the report is to provide supportive information to the groups working on the technical specifications for a FOTA/SOTA telecommunications standard
  - A secondary objective of the report is to provide background information to the vehicle manufacturer groups working on their internal processes aimed at delivering a vehicle-manufcturer-specific end-to-end FOTA/SOTA update solution for their own vehicles

Report on secure over-the-air vehicle software updates - operational and functional requirements

- Status of Report:
  - Reviewed in two meetings of the Collaboration on ITS Communication Standards (2015/12; 2016/03)
  - Submitted to ITU-T Study Group 17 (Security; meeting late 2016/03) and ITU-T Study Group 16 Multimedia; meeting 2016/05) for adoption as ITU-T Technical Paper
  - ITU-T Technical Papers are available on the ITU website, free of charge

Standard on secure software update capability for ITS communications devices

- Objectives of “ITU-T X.itssec-1”:
  - Assessment of security threats, risks and vulnerabilities
  - Provision of common methods to update vehicle software by a secure procedure
  - Security controls and protocol definition
- Note:
  - ITU-T standards (“Recommendations”) have non-mandatory status until they are adopted in national laws
  - This standard is aimed to provide a guideline for baseline security for networked vehicles

Standard on secure software update capability for ITS communications devices

- Status of “ITU-T X.itssec-1”:
  - Initiated in 2014/09
  - Draft achieved a certain level of maturity through discussions with some vehicle manufactures and suppliers
  - Draft is to be “determined” as an ITU-T Recommendation, the final stage of standardization, during the meeting of ITU-T Study Group 17 (Security), Geneva, 16-23 March 2016
  - ITU-T Recommendations are available on the ITU website, free of charge

Other ITS communications related activities ITU

- An up-to-date list of ITS related work items in ITU is available at **http://www.itu.int/en/ITU-T/extcoop/cits/Documents/ITS%20work%20items.xlsx**

Supplementary Material

1956 2016

CCITT / ITU-T

Report on secure over-the-air vehicle software updates - operational and functional requirements

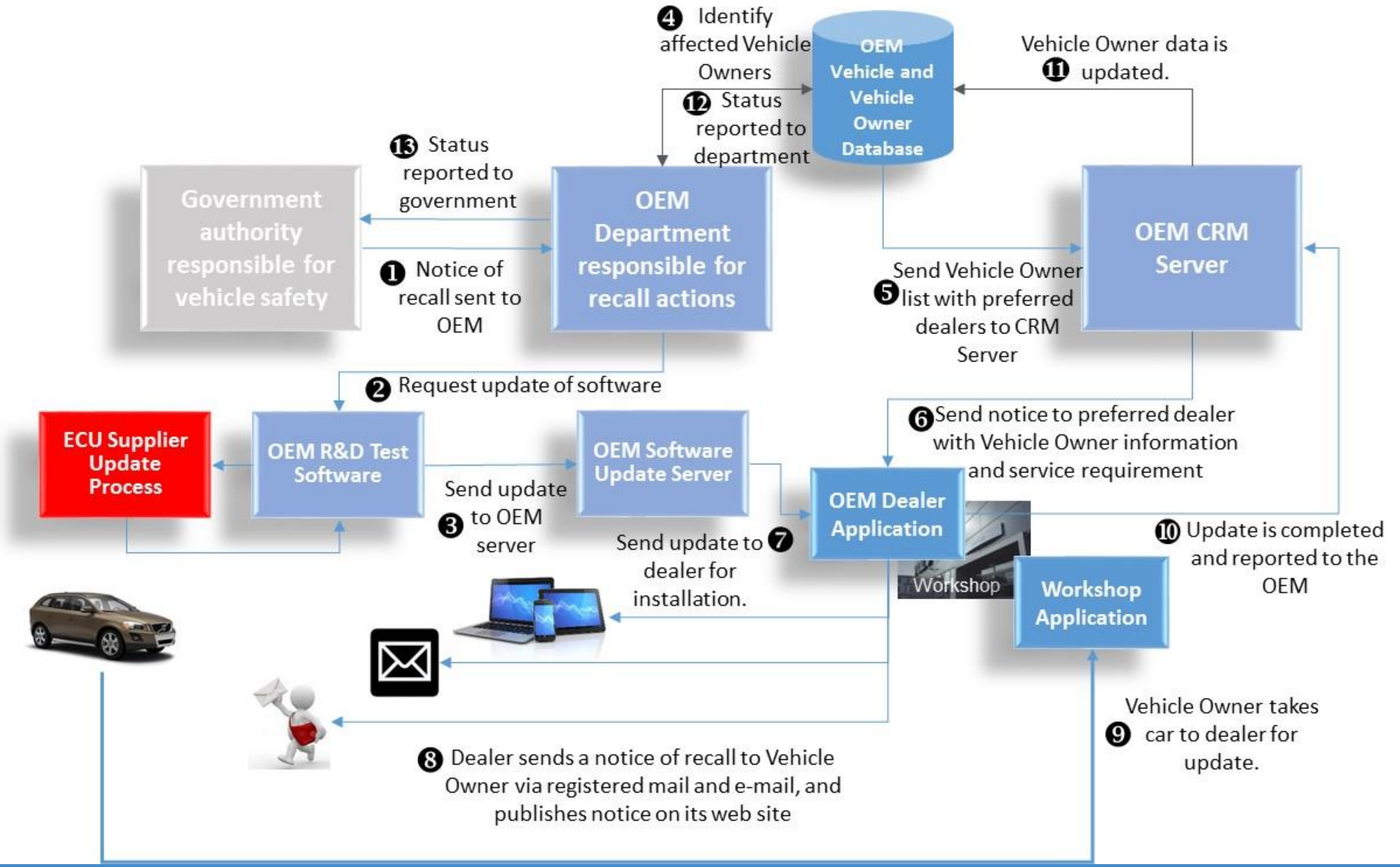
Structure

1. Introduction
   - Executive Summary
   - Objectives
   - References
   - Acronyms and Definitions
2. The Automotive Context
   - Are we ready for OTA Updates?
   - Use Cases
   - Conditions
   - National Standards Regulation and Type Approval Regulation Compliance
   - National Standards Initiatives for Security Risk Mitigation
3. Operational Requirements
   - Update preparation
   - Regulatory approvals
   - Permissions to perform update
   - End-to-end update management
   - Confirm receipt and proper functioning
   - Perform administrative tasks
4. Functional Requirements
   - Recall
   - Non-recall Operation Updates
   - Improvements to Performance
   - Security Risk Corrective Action

1956 2016
CCITT / ITU-T

Report on secure over-the-air vehicle software updates - operational and functional requirements

Current Process for Safety Recall



Government authority responsible for vehicle safety

OEM Department responsible for recall actions

OEM Vehicle and Vehicle Owner Database

OEM CRM Server

ECU Supplier Update Process

OEM R&D Test Software

OEM Software Update Server

OEM Dealer Application

Workshop

Workshop Application

1 Notice of recall sent to OEM

2 Request update of software

3 Send update to OEM server

4 Identify affected Vehicle Owners

5 Send Vehicle Owner list with preferred dealers to CRM Server

6 Send notice to preferred dealer with Vehicle Owner information and service requirement

7 Send update to dealer for installation.

8 Dealer sends a notice of recall to Vehicle Owner via registered mail and e-mail, and publishes notice on its web site

9 Vehicle Owner takes car to dealer for update.

10 Update is completed and reported to the OEM

11 Vehicle Owner data is updated.

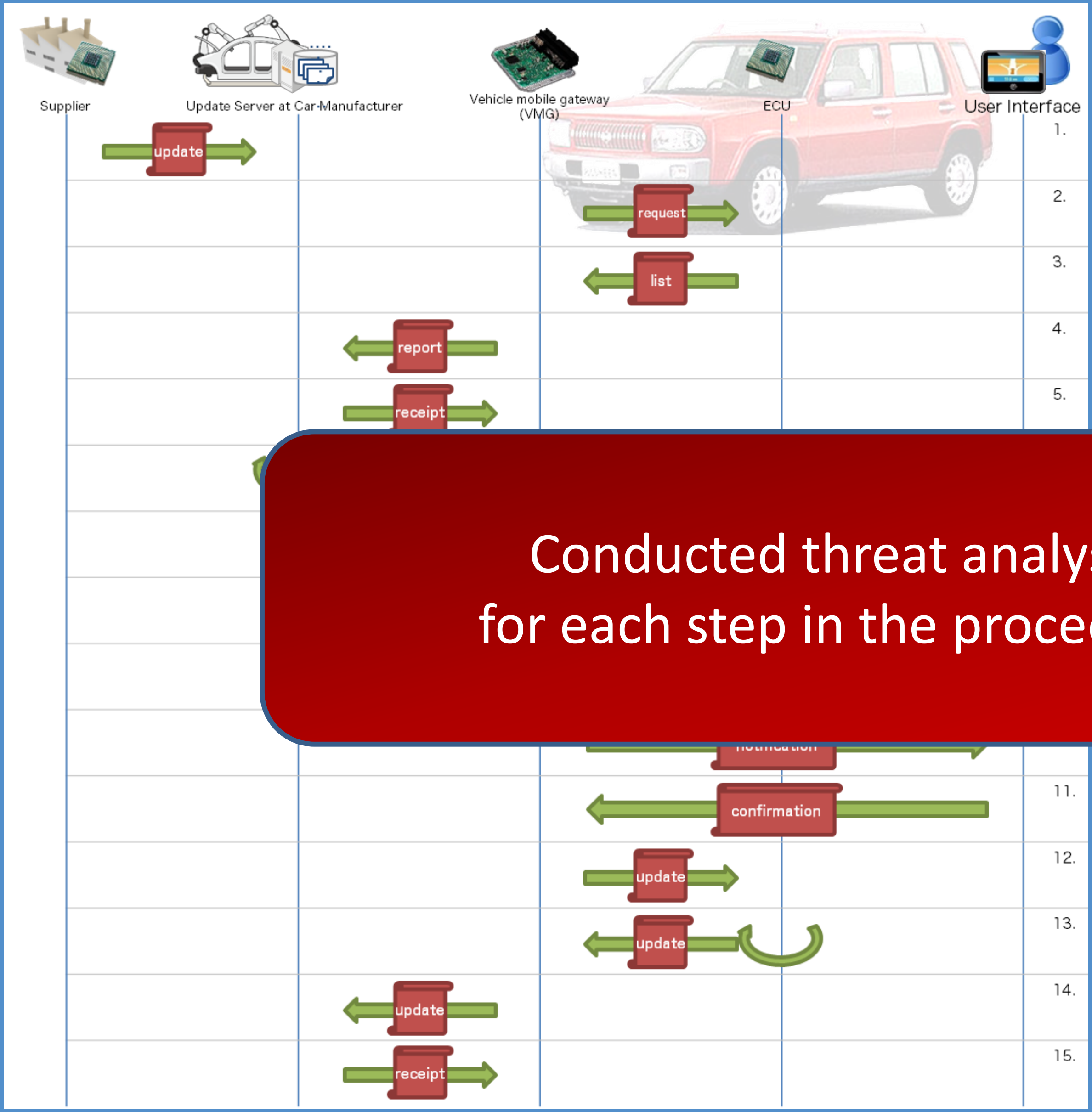
12 Status reported to department

13 Status reported to government

Report on secure over-the-air vehicle software updates - operational and functional requirements

- A properly designed system that provides for as high a level of security as is possible must protect each of the four levels of vehicle electronics systems that could be addressed by OTA software updates:
  - Information and entertainment systems
  - Fail safe body function systems
  - Fail safe driving and vehicle dynamic function systems
  - Fault functional driving and vehicle dynamic function systems

X.itssec-1: Model data flow of remote software update



Request of diagnose of software status

Result of diagnose with software status

Report of results of ECUs in a vehicle

Receipt for submit of diagnose report

e

ed

Notification to User (driver) for Updates

Confirmation for the update

Request for updates to ECUs

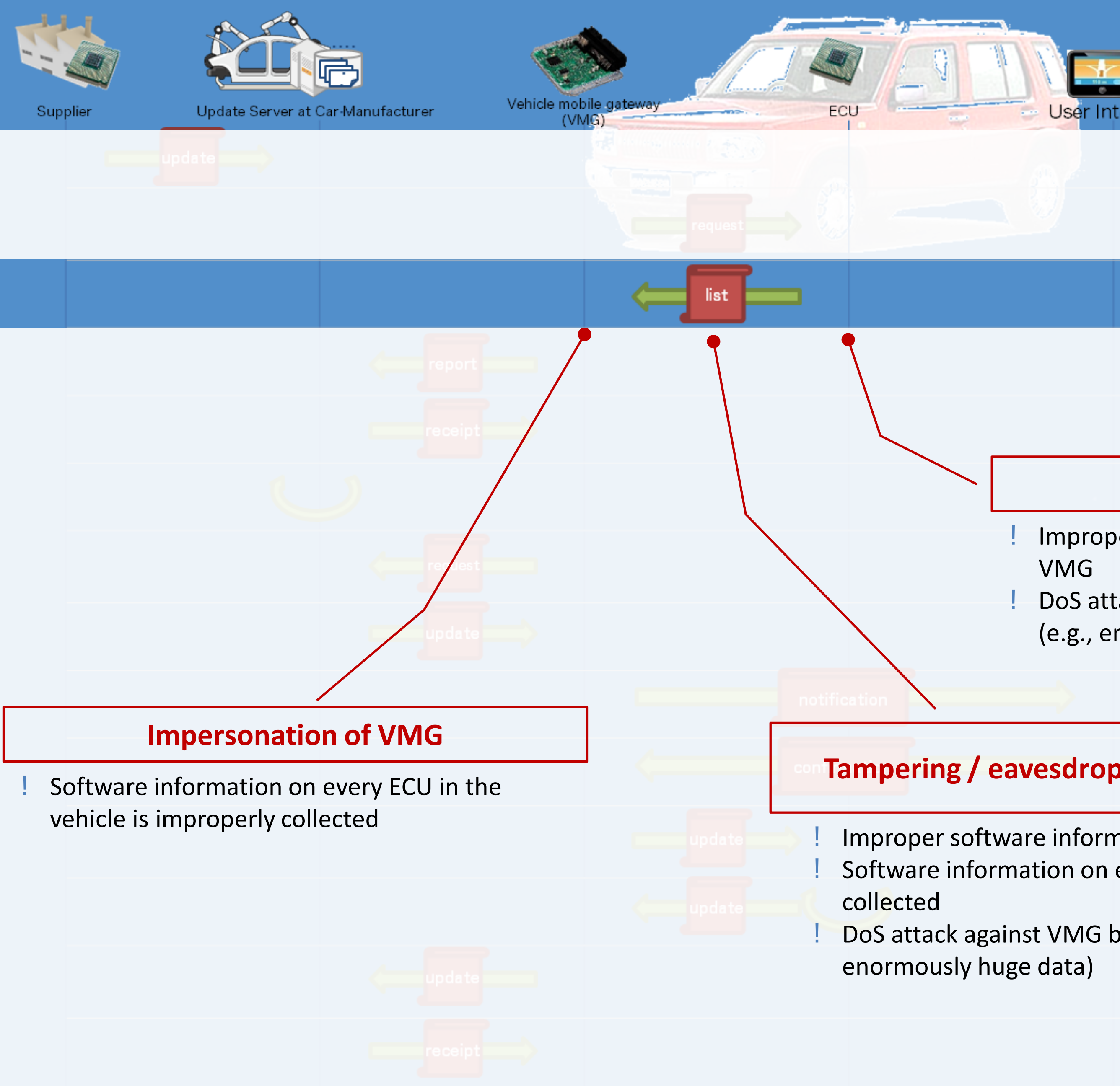
Results for updates in ECUs

Report of application of the update

Conformation from the Update server

Conducted threat analysis
for each step in the procedure

X.itssec-1: Threat analysis: example case



1. Supplier provides an update module to a car manufacturer.
2. Vehicle mobile gateway requests ECUs to diagnose themselves and submit their software list.
3. ECUs generate a software list and submit it to the Vehicle mobile gateway.
4. The vehicle mobile gateway gathers the lists of software and submit them to update server.
5. Update server issues a receipt of the software list for vehicle mobile gateway.
6. Update server determines necessary software modules for each ECU.
7. After a certain period of time, the vehicle mobile gateway requests update modules for the vehicle.
8. Update software modules are delivered to vehicle mobile gateway.
9. The gateway pushes a notification to a user interface.
10. The car owner confirms to apply the update via the user interface.
11. Vehicle mobile gateway delivers the updates to corresponding ECUs and request them to apply the updates.
12. Each ECU applies the update and reports the application result to the vehicle mobile gateway.
13. Finally the vehicle mobile gateway submits a report of application results to the update server.
14. Finally the vehicle mobile gateway submits a report of application results to the update server.

Impersonation of ECU

- Improper software information is uploaded to VMG
- DoS attack against VMG by sending forged data (e.g., enormously huge data)

Impersonation of VMG

- Software information on every ECU in the vehicle is improperly collected

Tampering / eavesdropping / replay of a message

- Improper software information is uploaded to VMG
- Software information on every ECU in the vehicle is improperly collected
- DoS attack against VMG by sending forged data (e.g., enormously huge data)

X.itssec-1: Security controls for the software update

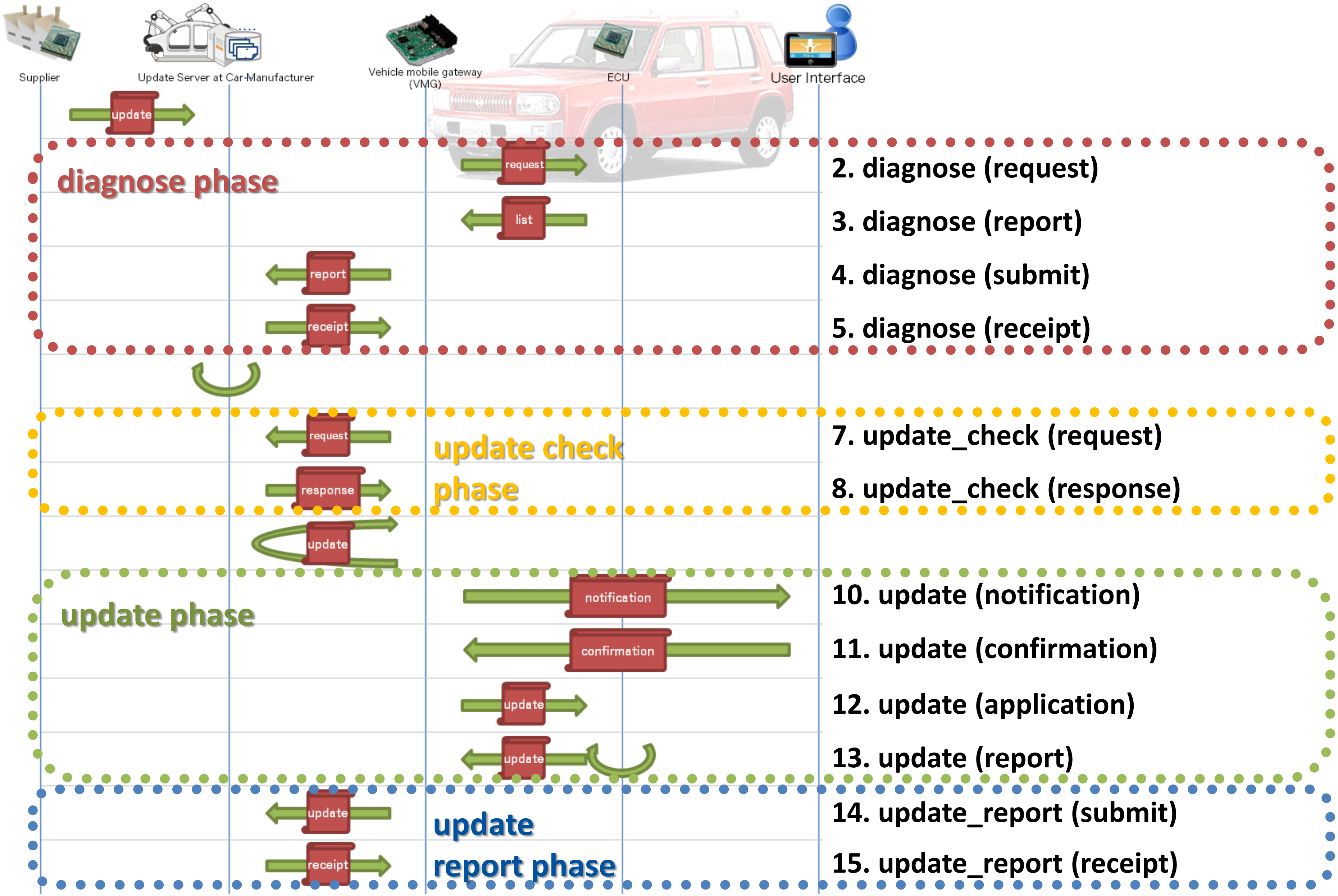
- ✓ **Message verification**
  - Threats: tampering, eavesdropping and replaying of messages
  - Measure: message verification mechanism based on Message Authentication Code (MAC) or digital signature method
- ✓ **Trusted boot of ECUs**
  - Threats: tampering of software in ECU
  - Measure : hardware Security Module (HSM) to verify software modules in ECUs' boot sequences

- ✓ **Authentication of communication entity**
  - Threats: impersonation of the entities
  - Measure : authentication of both client and server of each communication based authentication protocol such as SSL/TLS

X.itssec-1: Security controls for the software update

- ✓ **Message filtering**
  - Threats: DoS attack against VMG or update server
  - Measure : message filtering based on white listing of senders and frequency limitation of received messages, etc.
- ✓ **Fault tolerance**
  - Threats: DoS attack against VMG
  - Measure : measures such as auto-reboot for recovery of normal state, safe suspension of operation should be taken if something irregular is detected on the operation of VMG.

X.itssec-1: Procedure definition (Phases)



2. diagnose (request)

3. diagnose (report)

4. diagnose (submit)

5. diagnose (receipt)

7. update_check (request)

8. update_check (response)

10. update (notification)

11. update (confirmation)

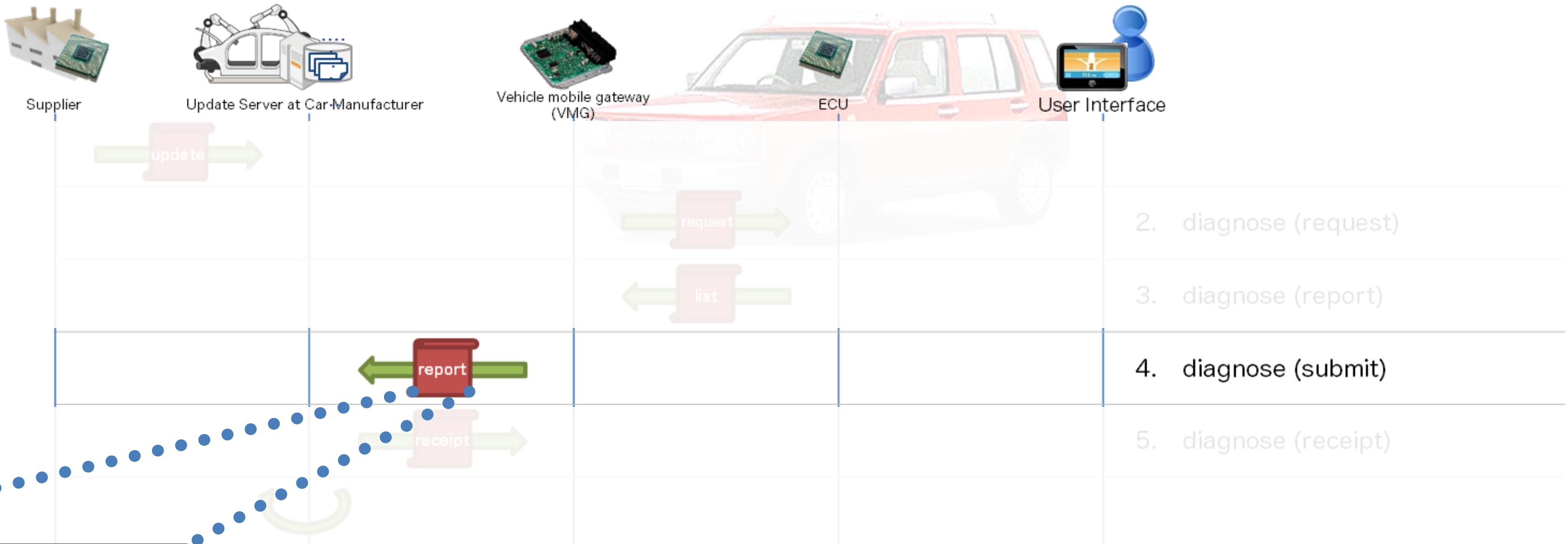
12. update (application)

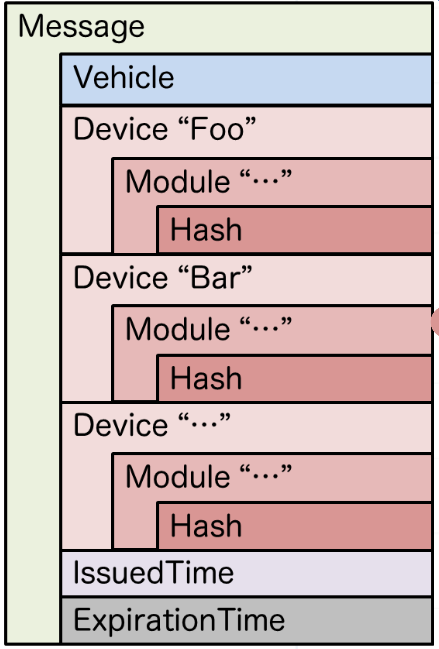
13. update (report)

14. update_report (submit)

15. update_report (receipt)

X.itssec-1: Example of a message: diagnose (submit)





Message

Vehicle

Device “Foo”

Module “…”

Hash

Device “Bar”

Module “…”

Hash

Device “…”

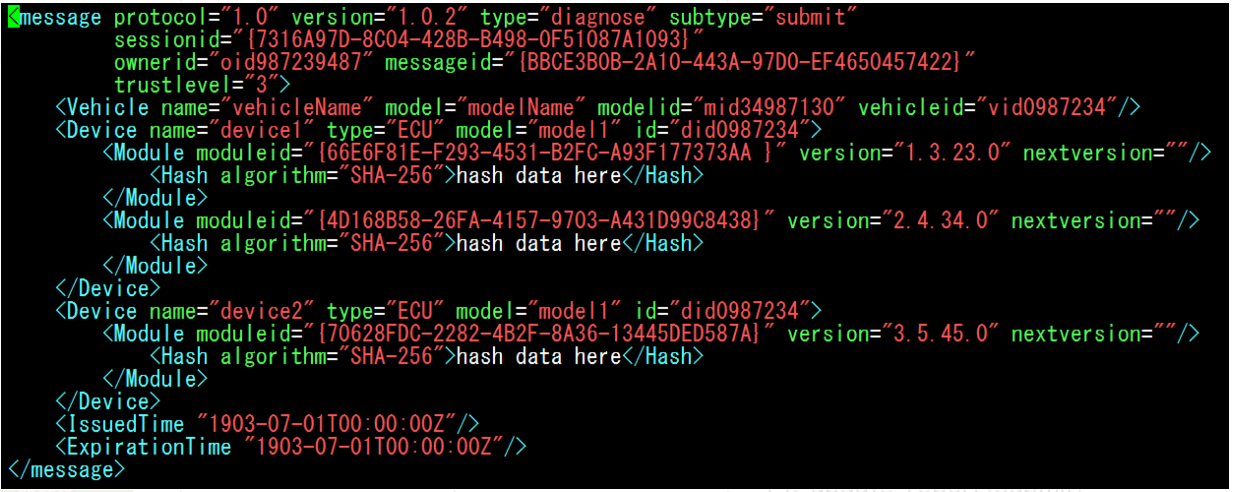
Module “…”

Hash

IssuedTime

ExpirationTime

Structure of diagnose (submit) message



```
<message protocol=”1.0” version=”1.0.2” type=”diagnose” subtype=”submit”
          sessionid=”{7316A97D-8C04-428B-B498-0F51087A1093}”
          ownerid=”oid987239487” messageid=”{BBCE3B0B-2A10-443A-97D0-EF4650457422}”
          trustlevel=”3”>
    <Vehicle name=”vehicleName” model=”modelName” modelid=”mid34987130” vehicleid=”vid0987234”/>
    <Device name=”device1” type=”ECU” model=”model1” id=”did0987234”>
        <Module moduleid=”{66E6F81E-F293-4531-B2FC-A93F177373AA }” version=”1.3.23.0” nextversion=””/>
            <Hash algorithm=”SHA-256”>hash data here</Hash>
        </Module>
        <Module moduleid=”{4D168B58-26FA-4157-9703-A431D99C8438}” version=”2.4.34.0” nextversion=””/>
            <Hash algorithm=”SHA-256”>hash data here</Hash>
        </Module>
    </Device>
    <Device name=”device2” type=”ECU” model=”model1” id=”did0987234”>
        <Module moduleid=”{70628FDC-2282-4B2F-8A36-13445DED587A}” version=”3.5.45.0” nextversion=””/>
            <Hash algorithm=”SHA-256”>hash data here</Hash>
        </Module>
    </Device>
    <IssuedTime ”1903-07-01T00:00:00Z”/>
    <ExpirationTime ”1903-07-01T00:00:00Z”/>
</message>
```

Practical expression of a diagnose (submit) message (example: XML)

1956 2016

CCITT / ITU-T