TTC技術レポート
Technical Report

TR-M2M-0026v3.0.1
車両領域への適用性

Vehicular Domain Enablement

2019 年 06 月 28 日制定

一般社団法人
情報通信技術委員会

THE TELECOMMUNICATION TECHNOLOGY COMMITTEE

TR-M2M-0026v3.0.1

Vehicular Domain Enablement

＜参考＞ [Remarks]

１．国際勧告等の関連 [Relationship with international recommendations and standards]

本技術レポートは、oneM2M で作成された Technical Report 0026 (Version 3.0.1) に準拠している。

[This Technical Report is transposed based on the Technical Report 0026 (Version 3.0.1) developed by oneM2M.]

２．作成専門委員会 [Working Group]

oneM2M 専門委員会 [oneM2M Working Group]

oneM2M
Technical Report

Document Number	TR-0026-V3.0.1
Document Name:	Vehicular Domain Enablement
Date:	2019-05-13
Abstract:	This oneM2M Technical Report examines how the current oneM2M System can be used in the Vehicular Domain and includes a study of advanced features which the future oneM2M release(s) could support for this vertical domain.

Template Version: 08 September 2015 (Dot not modify)

About oneM2M

The purpose and goal of oneM2M is to develop technical specifications which address the need for a common M2M Service Layer that can be readily embedded within various hardware and software, and relied upon to connect the myriad of devices in the field with M2M application servers worldwide.

More information about oneM2M may be found at: http//www.oneM2M.org

Notice of Disclaimer & Limitation of Liability

The information provided in this document is directed solely to professionals who have the appropriate degree of experience to understand and interpret its contents in accordance with generally accepted engineering or other professional standards and applicable regulations. No recommendation as to products or vendors is made or should be implied.

NO REPRESENTATION OR WARRANTY IS MADE THAT THE INFORMATION IS TECHNICALLY ACCURATE OR SUFFICIENT OR CONFORMS TO ANY STATUTE, GOVERNMENTAL RULE OR REGULATION, AND FURTHER, NO REPRESENTATION OR WARRANTY IS MADE OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR AGAINST INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS. NO oneM2M PARTNER TYPE 1 SHALL BE LIABLE, BEYOND THE AMOUNT OF ANY SUM RECEIVED IN PAYMENT BY THAT PARTNER FOR THIS DOCUMENT, WITH RESPECT TO ANY CLAIM, AND IN NO EVENT SHALL oneM2M BE LIABLE FOR LOST PROFITS OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES. oneM2M EXPRESSLY ADVISES ANY AND ALL USE OF OR RELIANCE UPON THIS INFORMATION PROVIDED IN THIS DOCUMENT IS AT THE RISK OF THE USER.

Contents

1 Scope

The present document examines how the current oneM2M System can be used in the Vehicular Domain and includes a study of advanced features which the future oneM2M release(s) could support for this vertical domain. The present document also analyses use cases and the potential requirements pertaining to the use cases with regard to vehicular domain.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] oneM2M Drafting Rules.

NOTE: Available at http://www.onem2m.org/images/files/oneM2M-Drafting-Rules.pdf.

[i.2] oneM2M TS-0002: "Requirements".

[i.3] oneM2M TS-0001: "Functional Architecture"

[i.4] oneM2M TS-0011: "Common Terminology".

[i.5] Recommendation ITU-T X.1362: "Simple encryption procedure for Internet of Things (IoT) environments".

[i.6] IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".

[i.7] ISO/IEC 29192:2012 "Lightweight cryptography".

[i.8] PRESERVE: "V2X Security Architecture v2".

[i.9] 3GPP TR 33.885: "Study on security aspects for LTE support of Vehicle-to-Everything (V2X) services".

[i.10] 3GPP TS 33.185: "Security aspect for LTE support of Vehicle-to-Everything (V2X) services".

[i.11] EVITA: "Secure on-board architecture specification".

[i.12] EVITA: "Secure on-board protocols specification".

[i.13] Trusted Computing Group: "TCG TPM 2.0 Automotive Thin Profile".

[i.14] oneM2M TS-0003: "Security Solutions".

[i.15] GSMA: "SGP.01 - Embedded SIM Remote Provisioning Architecture".

[i.16] oneM2M TS-0004: "Service Layer Core Protocol Specification".

[i.17] W. Chen, Telcordia Technologies: “Vehicular Communications and Networks: Architectures, Protocols, Operation and Deployment”

[i.18] AUTOPILOT: http://autopilot-project.eu/

[i.19] SAREF: The Smart Appliances REFerence (SAREF) ontology.

NOTE: Available at http://ontology.tno.nl/saref/.

[i.20] HERE: Vehicle Sensor Data Cloud Ingestion Interface Specification(v2.0.2)

NOTE: Available at https://lts.cms.here.com/static-cloud-content/Company_Site/2015_06/Vehicle_Sensor_Data_Cloud_Ingestion_Interface_Specification.pdf

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in oneM2M TS-0011 [i.4] apply.

3.2 Symbols

For the purposes of the present document, the symbols given in oneM2M TS-0011 [i.4] apply.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in oneM2M TS-0011 [i.4] and the following apply:

ADApp	Autonomous Driving Application
ARIB	Association of Radio Industries and Businesses
ATIS	Alliance for Telecommunications Industry Solutions
CCSA	China Communications Standards Association
DSRC	Dedicated Short Range Communications
DTG	Digital Tachograph
ETC	Electronic Toll Collection
ETSI	European Telecommunications Standards Institute
GPS	Global Positioning System
HSM	Hardware Security Modules
ITS	Intelligent Transportation System
LIDAR	Light Detection and Ranging, Laser Imaging Detection and Ranging
OBU	On Board Unit
RSU	Road Side Unit
TIA	Telecommunications Industry Association,
TPM	Trusted Platform Module
TSDSI	Telecommunications Standards Development Society
TTA	Telecommunications Technology Association
TTC	Telecommunication Technology Committee
V2I	Vehicle to Infrastructure
V2V	Vehicle to Vehicle
V2X	Vehicle to Everything

4 Conventions

The key words "Shall", "Shall not", "May", "Need not", "Should", "Should not" in the present document are to be interpreted as described in the oneM2M Drafting Rules [i.1].

5 Introduction to Vehicular Domain

5.1 Vehicular Domain Overview

Existing ITS (Intelligent Transport Systems) services in automotive industry have been provided through the architecture which is composed of connected vehicles to various infrastructures such as the ETC (Electronic Toll Collection) service, VICS (Vehicle Information and Communication System) or telematics service centers. Those systems have been growing by improving mobility convenience for drivers. However, the infrastructure systems are required to change and become more collaborative, in order to meet rising social demands such as energy saving, traffic congestion resolution and fatal accidents avoidance.

With rising computational capabilities of participants in traffic - pedestrians (with their smartphones), vehicles that have communication and computing capabilities, smart traffic lights, smart streets and crossings, etc., corresponding vehicular infrastructure systems are becoming more collaborative, capable of exchanging information with other participants in traffic, with the goal of meeting rising social demands such as energy saving, managing traffic congestion and minimizing and avoiding fatal accidents.

In order to make optimal decisions in traffic, participants, smart traffic lights, vehicles, smart roads, etc. do so by creating their own 'world model' [i.17]. It is vehicle's representation of the state of the external environment (which other participants are, what their state of movement is for given moment). Based on world model, vehicle can make optimal decisions on its own actions. If we look at an example of vehicle, to create world model, vehicle relies on measured values it gets from its own sensors (like ABS sensor, radar, LIDAR, brake switch, tire pressure sensors, accelerator switch, info on vehicle speed from motor management, etc.). By processing data from these sensors, vehicle creates awareness on its own state, and can therefore manage movement of vehicle in better way.

When adding V2V/V2I communication with other vehicles, it is possible to have even better, more complete world model. Immediate environment of vehicle can be better mapped. For example, if vehicle immediately in front brakes, that vehicle's brake switch, and thus motor management is immediately aware of braking process. It takes some time to driver of vehicle behind it to observe that braking lights have turned ON, and it takes further time before driver of vehicle processes that information and takes corrective action (hitting own brakes). By allowing vehicles to communicate directly and exchange part of their own world models that are relevant for other participants in traffic, each vehicle is therefore able to get quickly more complete world model, and allow for vehicles to take corrective measures immediately.

The same reasoning applies to other 'smart' elements in traffic like traffic lights, roads, road crossings, merging lanes on highways, etc.

5.2 Technology Trends in Vehicular Domain

In order to develop vehicular domain architecture as ITS platform, many worldwide organizations are discussing technology and standardization related to this field.

ISO (International Organization for Standardization) is an international standard-setting body which standardized technologies for ITS systems through the work of TC (Technical Committee) 204 (ITS) and 22 (Road Vehicles). In recent years, ISO has focused on CITS (Cooperative ITS) using V2V/V2I communication for some new services such as automated driving and Urban ITS which improves mobility in urban area. Furthermore, it has made an effort to develop gateways between in-vehicle area network and nomadic devices like smartphones.

ITU-T (International Telecommunication Union Telecommunication Standardization Sector) is one of the three sectors of the ITU and it coordinates standards for telecommunications. In SG16 (multimedia) of ITU-T, VGP (Vehicle Gateway Platform) has been standardized in terms of telecommunication. In particular, reference architecture and functional architecture based on supposed applications and interface protocol to vehicles or ICT devices are discussed.

ETSI (European Telecommunications Standards Institute) is a standardization organization in the telecommunications industry (equipment makers and network operators) in Europe. In TC (Technical Committee) ITS, standards, specifications and other deliverables to support the development and implementation of ITS Service provision across the network have been developed. The scope includes communication media, and associated physical layer, transport layer, network layer, security, lawful intercept and the provision of generic web services.

The W3C (World Wide Web Consortium) is the main international standards organization for the World Wide Web. In 2013, it has established the Automotive and Web Platform Business Group and prepared a draft of Vehicle Information Access and Vehicle Data Spec. Currently these topics are in discussion to standardize in the Automotive Working Group. Responding to the growing needs of web services for a connected car, the W3C focuses on interfaces for application vendors to access vehicle data using a standard and secure method.

In reference to V2V/V2I communication technologies which support M2M services in vehicular domain, DSRC (Dedicated Short Range Communications) has been discussed in ITU-R (International Telecommunication Union Radio-communications Sector) and IEEE (The Institute of Electrical and Electronics Engineers). Moreover cellular communication for vehicular domain is also discussed in 3GPP (Third Generation Partnership Project). These wireless communication technologies are expected to resolve some problems such as cost, coverage, latency and power consumption, while a unified communication standard is expected to accelerate the growth of M2M services in business.

Table 5.2-1 Vehicular Domain Standards

No	Organization	Sector	Focus point	Major topics
1	ISO	TC204, TC22	ITS services	Cooperative System, In-vehicular gateway
2	ITU-T	SG16	Telecommunication	Vehicle Gateway Platform, Communication protocol
3	ETSI	TC ITS	Network system, Radio Technique	Cooperative ITS, DSRC
4	W3C	Automotive WG	Web services	Web-API for vehicles
5	ITU-R	SG5/WP5A	Wireless communication	V2V/V2I communication (DSRC)
6	IEEE	802.11p + P1609	Wireless communication	V2V/V2I communication (DSRC)
7	3GPP	-	Wireless communication	V2V/V2I communication (Cellular)

5.3 The focus of oneM2M in Vehicular Domain

The changing trends are enabled by improvements of IT technologies such as V2V/V2I communication, cloud systems or OSS (Open Source Software). The M2M architecture for vehicle domain described in this TR accelerates wide collaboration between not only auto manufacturers or suppliers but also between telecommunication providers or government, which results in improvement of safety, comfort and eco-friendliness. Figure 6.1.1-1 is an example of reference architecture.

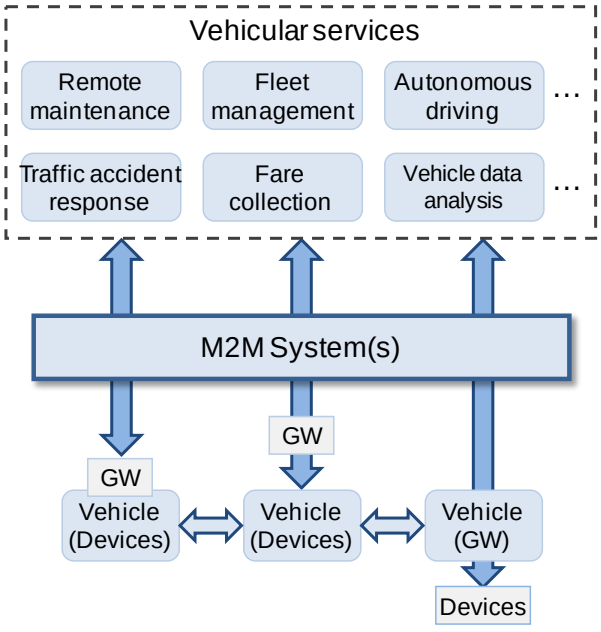

Figure 6.1.1-1: Vehicular Domain Architecture

In this architecture, vehicles can connect to various services via M2M systems. Gateways which collect sensing data could be located both inside (left of figure) and outside of the vehicle (center of figure). A vehicle may act as a gateway

and collect also data generated by sensors located outside of a vehicle (right of figure). Furthermore vehicles may connect to each other and a vehicle may connect to services via other vehicles.

This vehicular domain architecture provides the basic functions for analysing huge amounts of data collected from large number of vehicles which makes it possible to create future services as a result of collaboration with other industries.

Furthermore, multiple security levels according to features of collected data make it possible to provide critical control and management for vehicles such as remote maintenance or automated driving control via communication networks.

In the context of progress achieved through various standardization efforts, oneM2M is continuing to develop a common platform which can be used for information transport functions between vehicles and backend servers over a variety of transport technologies (e.g. cellular or DSRC) to satisfy vehicular domain requirements. The following capabilities are of key importance to the vehicular domain.

Data management

Since vehicles may move fast, the operation and management in this domain will be dynamic, complex and difficult. For example, if the nodes located on roadside act as gateway, status of a connection between device and gateway is varying frequently. Therefore, management of data (including sensing and device information, data streams) as well as filtering and pre-processing functions (at devices or gateways) are required in order to provide efficient analysis services in the complicated environments.

Communication management

M2M services in vehicular domain need to deal with many and widespread devices. On the other hand, the target devices of the operation are depending on the services. For instance, it is expected that the content is distributed to the vehicles in a specific geographical area. Therefore, the communication management function is needed to select appropriate communication protocols and modes (e.g. unicast, multicast or broadcast), according to features of data, for efficient utilization of network resources.

Location

Geographical location information is important in vehicular services. To improve safety, up-to-date location information is important for collision avoidance. Therefore, accurate location providing/managing functions are needed in this vehicular domain. Furthermore, the inherent node mobility in the vehicular domain creates greater variability in the services available along the way. Mobility management functionality is required to allow maintaining the vehicular services as the node moves, and to provide means for selection and migration of services between nodes.

Security

To mitigate the life-threatening risk resulting from malicious control and management of vehicles via network, components of vehicular services should support strict authentication functionality. Furthermore, integrity and encryption functionalities for communication are required, with consideration for on system scalability and resource constraints at the devices.

5.4 Levels of Driving Automation

As mentioned in previous clauses, vehicles can exchange data on its own state, as well as receive data from surroundings (infrastructure - traffic lights, roads, other participants - vehicles, pedestrians, cyclists, etc.) and its state.

For vehicles exchanging data with its surroundings, we can distinguish number of levels of connectivity.

One is **connected car** - meaning car is connected to the Internet, typically for use of mobile network services. Examples of this are use of mobile network to Wi-Fi AP gateway, enabling passengers within vehicle to access internet via Wi-Fi. Other use is for collecting data from GPS unit within car to send information on current vehicle's location and speed. Data collected in this way is typically used for road management and routing advice (detecting congestion in roads).

Next level of connectivity is **cooperative car** - where car is communicating with other vehicles, either directly or indirectly (via RSU, or some other infrastructure element), with the goal of creating LDM (Local Dynamic Map) which contains all vehicles in vicinity and their current state (speed, acceleration/deceleration, changing lanes, …). Further, each vehicle is announcing (via local broadcast, within range of used communication technology) its own state. Based on received information from other vehicles, and own sensors, vehicle can make decision on actions it needs to take. Typically, cooperative driving is providing driver with additional information on state of traffic in immediate vicinity, and vehicle can also take some of actions (keeping distance, braking) itself.

Next stage is **automated driving**, which refers to the capability of the vehicle to drive from one location to another, without intervention from humans, and in a safe way, without incurring damage to surroundings (pedestrians, buildings, other vehicles) and to its (vehicle's) passengers. Here vehicle has all information like in cooperative driving (state of other vehicles in vicinity), but gets information from other participants in traffic as well - infrastructure (roads, cameras, traffic lights, etc.) but also pedestrians, cyclists, etc.

Different levels of automation in driving are shown in Table 5.4-1 Summary of Automated Driving level, . From top to bottom we can see increasing levels of automation, with 'Full automation' being autonomous driving.

SAE Level	Name	Definition	Execution of steering and acceleration/ deceleration	Monitoring of driving environment	Fallback performance of dynamic Driving Track	System Capability (Driving Modes)
Human driver monitors the driving environment						
0	No automation	The full-time performance by the human driver of all aspects of the dynamic driving task, even when enhanced by warning or intervention system	Human driver	Human driver	Human driver	N/A
1	Driver assistance	The driving-mode specific execution by a driver-assistance system of either steering or acceleration/deceleration using information about the driving environment and with the expectation that the human driver performs all remaining aspects of the dynamic driving task,	Human driver and system	Human driver	Human driver	Some driving modes
2	Partial Automation	The driving-mode specific execution by one or more driver-assistance systems of both steering or acceleration/deceleration using information about the driving environment and with the expectation that the human driver performs all remaining aspects of the dynamic driving task,	System	Human driver	Human driver	Some driving modes
Automated driving system (“system”) monitors the driving environment						
3	Conditional Automation	The driving-mode specific execution by an automated driver-assistance systems of all aspects of the dynamic driving task with the expectation that the human driver will respond appropriately to a request to intervene.	System	System	Human driver	Some driving modes
4	High Automation	The driving-mode specific execution by an automated driver-assistance systems of all aspects of the dynamic driving task even if a human driver does not respond appropriately to a request to intervene.	System	System	System	Some driving modes
5	Full Automation	The driving-mode specific execution by an automated of all aspects of the dynamic driving task under all roadway and environmental conditions that can be managed by a human driver.	System	System	System	All driving modes

Table 5.4-1 Summary of Automated Driving level

Using IoT for vehicular domain, vehicles will be able to go beyond cooperative driving, and move into automated driving, levels 3-5 in Table 5.4-1.

Rest of the present document will refer to this figure, when applicable.

6 Vehicular Domain Use Cases

6.1 Vehicle Diagnostic & Maintenance Report

6.1.1 Description

The Vehicle Service Center wants to help the vehicle owner to be aware of the status of the vehicle and remind them to maintain the vehicle in a timely manner to avoid any damages.

Hence the Vehicle Service Center needs to obtain and analyse data from the vehicle periodically. Based on the analysis result, it will notify to the vehicle owner showing what's going on with the vehicle, in simple language and images together with some maintenance suggestions.

6.1.2 Source

oneM2M-REQ-2012-0067R03 Vehicle Stolen and Vehicle Diagnostics.

6.1.3 Actors

- Vehicle Owner:
  - By reading the Vehicle Diagnostic & Maintenance Report sent from the Vehicle Service Center, the vehicle owner would decide whether to maintain his/her vehicle.
- Vehicle Service Center:
  - It operates a service platform for diagnostics and maintenance of vehicles, obtains and analyses the diagnostics data from the vehicle. It will also send vehicle Diagnostic & Maintenance Report in e-mail together with maintenance suggestions to the vehicle owner.
- Mobile Communication Network Operator:
  - As the transmission medium, it supports the network services between Vehicle Service Center and Vehicle for the information transmission.
- M2M Device:
  - It is embedded in a vehicle, which is used to send information to Vehicle Service Center and implement diagnostics function from Vehicle Service Center.

6.1.4 Pre-conditions

1) The vehicle supports the diagnostics pre-configured to report the diagnostics data collected from sensors within the vehicle periodically.
2) The vehicle is already ignited.

6.1.5 Triggers

None.

6.1.6 Normal Flow

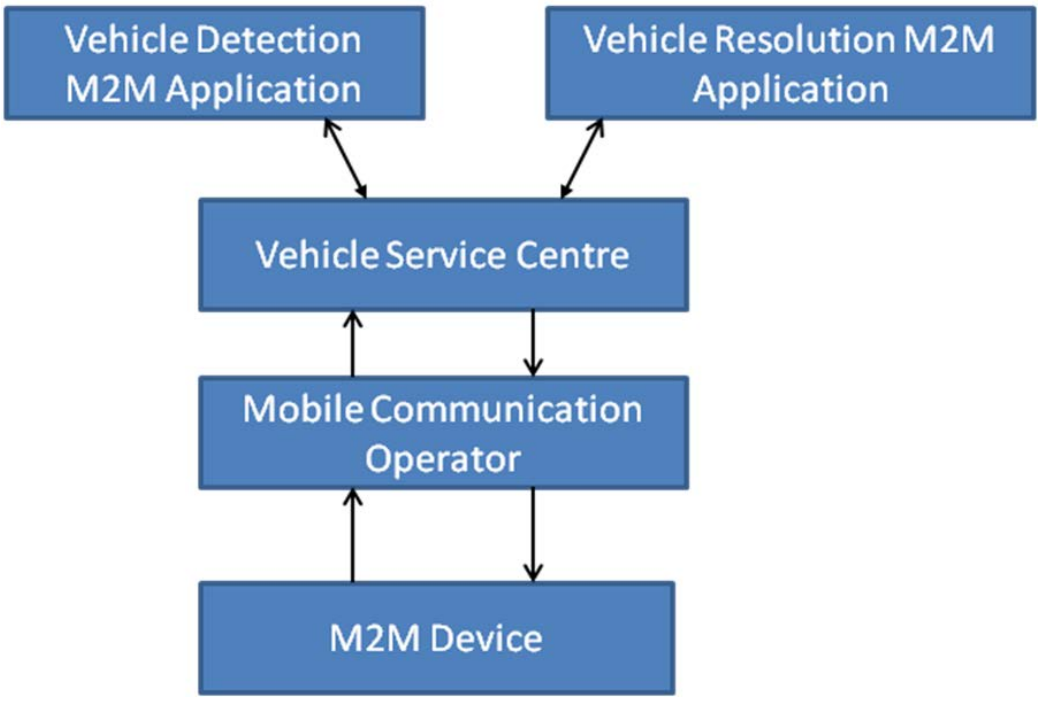

Figure 6.1.6-1: Vehicle Diagnostics Normal Flow

1) The vehicle collects the diagnostics data from sensors within the vehicle and sends it to the Vehicle Service Center. The diagnostics data includes information from Engine and Transmission System, Stability Control System, Air Bag System, Emission System, Antilock Brake System and so on. The information includes tyre pressure, odometer data, life of engine oil, engine and gear-box status, antilock braking system status, etc.

2) The Vehicle Service Center sends the diagnostics data to the "Vehicle Detection M2M Application". This M2M application receives and analyses the diagnostics data.

3) The "Vehicle Detection M2M Application" finds that the Brake pads need to be replaced. It queries the maintenance services provided by "Vehicle Resolution M2M Application" and gets the information of the company who can provide the components.

4) The "Vehicle Detection M2M Application" finally sends the Diagnostic & Maintenance Report to the vehicle owner together with the suggested component providers either by email or alert message displayed in the vehicle terminal.

5) The vehicle owner will decide whether to maintain his/her vehicle based on the Diagnostic & Maintenance Report.

6.1.7 Alternative Flow

None.

6.1.8 Post-conditions

For normal flow, the vehicle owner maintains his/her vehicle according to the Diagnostic & Maintenance report in time.

6.1.9 High Level Illustration

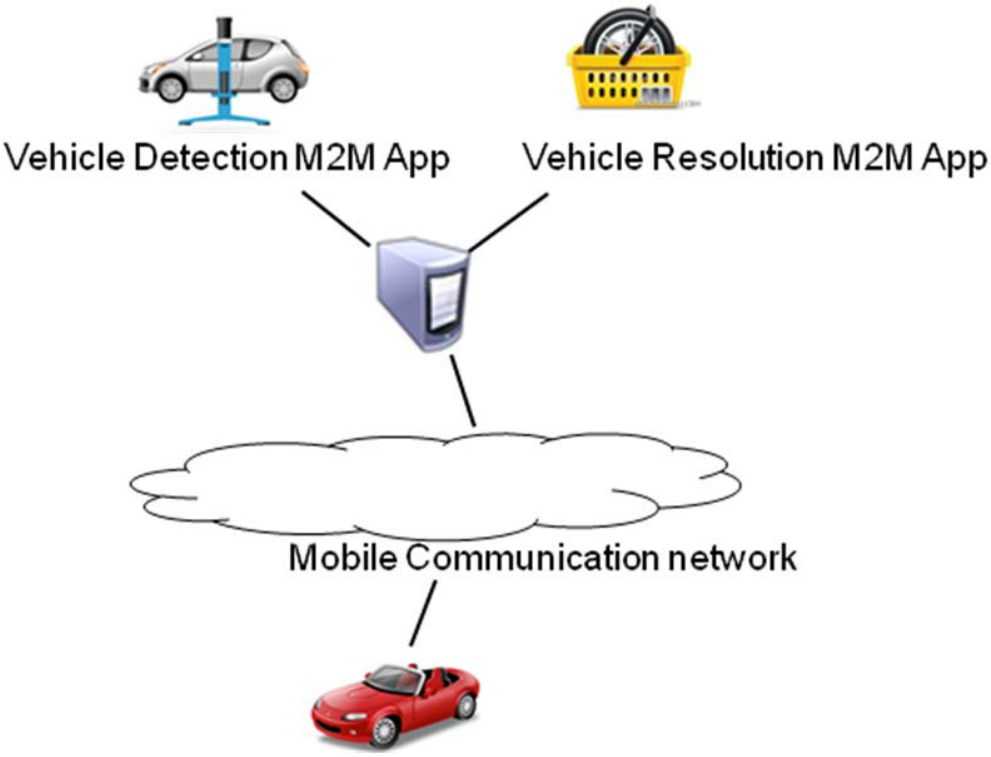

Figure 6.1.9-1 Vehicle diagnostics High Level Illustration

6.1.10 Potential Requirements

1) The M2M System shall enable the M2M Devices to exchange M2M application to diagnostic data periodically with the M2M Application in the network domain.

2) The M2M System shall enable the M2M Application to configure the notification interval in the M2M Devices.

3) The M2M system shall support a mechanism to describe the syntax and semantics format of the M2M application diagnostics data exchanged between the M2M Devices and the M2M Application in the network domain.

6.2 Use Case on Remote Maintenance Services

6.2.1 Description

This use case introduces a remote maintenance service for the automobiles (cars).

Because integrity of the cars is a matter of human life, the remote maintenance service of the car (treated as M2M Gateway in this use case) should be strongly secured.

Therefore, the integrity measurements both before and after the remote maintenance operation should also be securely performed.

One of the methods to endorse the measurement process might be guaranteed by HSM (Hardware Security Modules) in the M2M Gateway. This method provides a higher reliability level than that by a software emulator, the decision on the level of security is based on the information sent to the center. In the HSM method, the integrity measurement report can be made through an internal the mechanism in the HSM and put in the electronic signature using the key in the HSM.

This use case is derived from the automobiles, but similar cases of remote maintenance services could be considered with Medical equipment, Household applications, financial transaction terminals and Industrial control and machinery.

6.2.2 Source

oneM2M-REQ-2013-0188R06 Use Case Remote Maintenance.

6.2.3 Actors

Relevant to the name in the figure in clause 6.2.9, High Level Illustration:

- Car: the machine works as a M2M Gateway in which M2M Device(s) is implemented as the parts of it.
- Center: the M2M Platform which provides remote maintenance.
- The Hardware Security Module (HSM): a module in the M2M Gateway (e.g. Trusted Platform Module) that helps determining the level of security functions to endorse the integrity measurement process and holds the electronic signature key.
- A white list: data base which is accessed by the center may be used for verifying the integrity measurement report from the M2M Gateway (car), using a secure communication protocol e.g. Trusted Network Connect TNC protocol.
- Support software: installable software module to check the integrity of the Car assisted by TPM or the emulator and to support the newly implemented M2M Device(s) (i.e. sensor(s)).

6.2.4 Pre-conditions

Center recognizes the software which is installed in the Car that shall be updated.

6.2.5 Triggers

None.

6.2.6 Normal Flow

1) Mutual authentication between the Car (M2M Gateway) and the Center (M2M Platform) is performed.
2) Center requests the Car to report the integrity check on that Car.
3) Support software which is installed in the Car runs integrity check of the Car assisted by TPM or the emulator.
4) Generated integrity status/configuration information report is endorsed by the hardware key which is protected by TPM. This report may contain a detection of the newly implemented sensor(s) (M2M Device(s)).
5) Support software sends the report based on TNC (Trusted Network Connect, which is application level secure communication protocol) to the Center.
6) Center verifies the report securely based on the White list which is based outside the M2M network.
7) Center determines whether the Car contains the software which shall be updated.
8) Center selects corresponding software modules.
9) Center delivers the support software module to the Car.
10) The support software is applied at the Car.
11) The applied result endorsed by the device key (actual process is done by TPM or the emulator) is reported to the Center.
12) Center side confirms the completion of delivery/embedding.

13) Center side stores the sequence of operations log as certifiable evidence for indemnity.

6.2.7 Alternative Flow

None.

6.2.8 Post-conditions

Newly installed software/sensor(s) is correctly identified as authorized part(s) on the Car, and working correctly with installed support software. The Car's integrity status/configuration information data which is endorsed by the hardware key which is protected by TPM or the emulator is sent to the Center side.

6.2.9 High Level Illustration

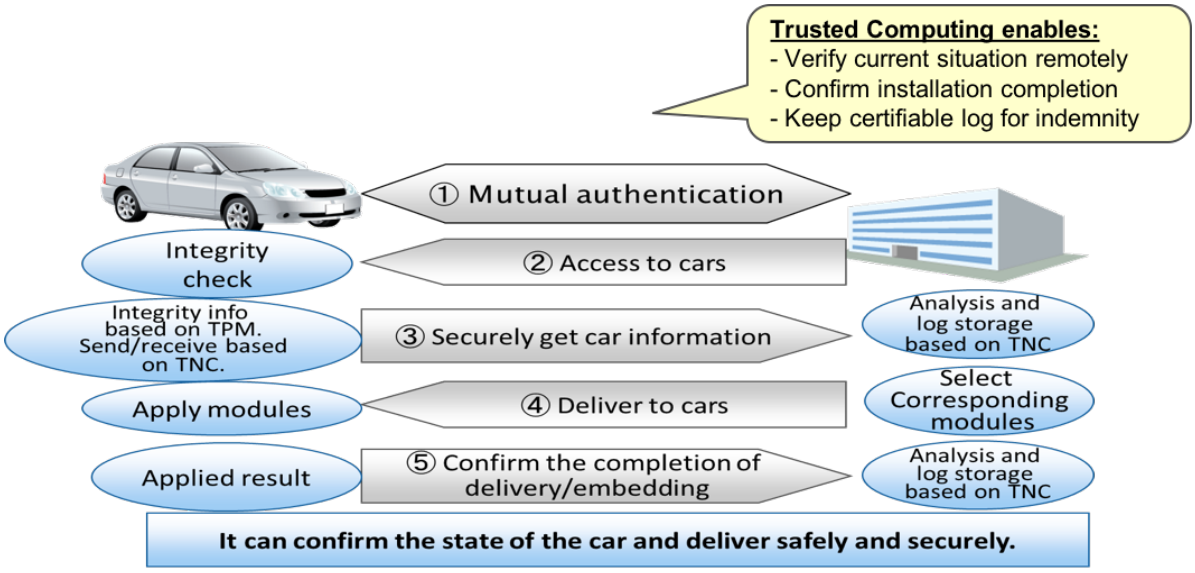

Figure 6.2.9-1: Remote Maintenance Flow

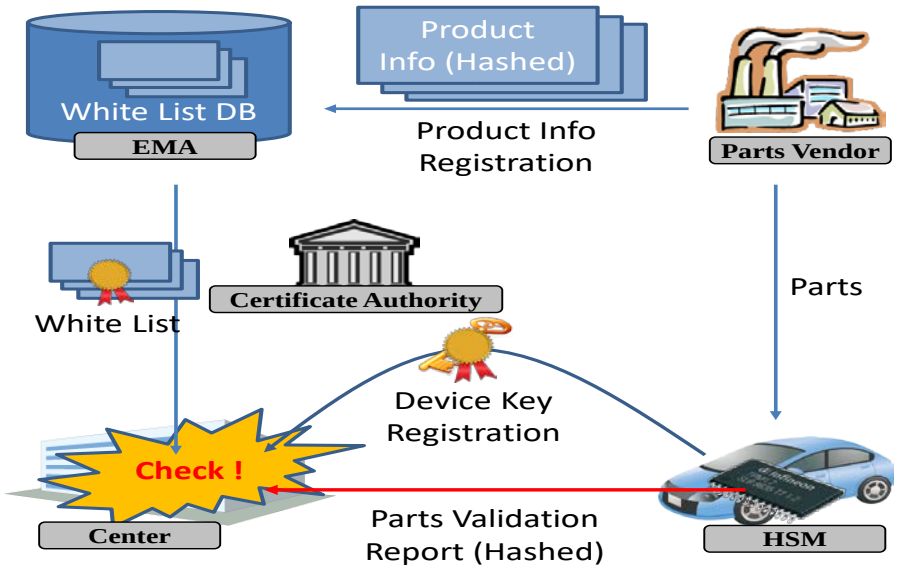

Figure 6.2.9-2: Remote Maintenance High Level Illustration

6.2.10 Potential Requirements

1) The M2M service shall be able to provide the mechanism for authorization for integrity-checking and installing processes of software/hardware/firmware component(s) on M2M Device(s) ([i.2], SER-013).

2) The M2M system shall be able to support authentication using device key on the integrity check for M2M Device(s).

3) The M2M Device shall be able to support HSM (Hardware Security Module) to protect its integrity depending on the security level requirement.

6.3 Traffic Accident Information Collection

6.3.1 Description

The Intelligent Transportation System (ITS) is mainly used for avoiding collision of vehicles. If doing some extension an ITS can also be used for other purposes such as electronic payment of road tolls, traffic information collection and broadcast, local service advertisements, etc.

It is for sure that the ITS will save a lot of lives, but some traffic accidents will occur any way. So we still need rescue teams to go to the accident sites to help the victims and police to ease the traffic jam caused by the accident. A rescue team can make a more proper rescue plan if they are able to see the scene of accident. Similarly police can make a better traffic control plan if they are able to get an overview of traffic situation near the accident site.

This use case will show how the M2M technologies can help people to timely access to the detailed information of a traffic accident.

6.3.2 Source

oneM2M-REQ-2013-0264R05 Use Case Traffic Accident Information Collection.

6.3.3 Actors

M2M Platform: It stores M2M data and runs M2M applications. It provides various M2M services to M2M service subscribers.

ITS Center: It is responsible for managing ITS on M2M Platform. It decides what service is provided to an ITS service subscriber.

Police Station: It is a subscriber of ITS service on M2M platform and responsible for controlling the traffic.

Rescue Center: It is a subscriber of ITS service on M2M platform and responsible for carrying out rescue missions.

ITS-Station (ITS-S): It is a kind of M2M Device installed in vehicles. It broadcast its travel status in a fixed interval or upon specific events in order to inform other ITS-S where it is. The ITS-S is equipped with a digital camera used for taking pictures according to the command given by a driver, ITS center or ITS-S itself. The ITS-S is able to communicate with M2M Platform through wireless network or a RSU using DSRC.

Road Side Unit (RSU): It is a kind of M2M Gateway installed at roadside. The RSU is able to communicate with ITS-S using DSRC and communicate with M2M Platform through wired or wireless network.

6.3.4 Pre-conditions

The ITS-Ss are equipped with a digital camera.

The ITS-Ss nearby the accident site are able to connect to M2M platform through either the wireless network or a RSU.

Police Station and Rescue Center are the subscribers of ITS services.

6.3.5 Triggers

There are two ways to start an accident reporting process. One is the ITS-S involved in an accident detects the crash and then starts an accident reporting process automatically; the other is a driver in a passing by vehicle manually starts an accident reporting process through giving a command to the ITS-S in his vehicle. Alternately, RSUs (e.g. a RSU involved in collision avoidance at an intersection) may be able to detect an accident and start the reporting process automatically, or they may be remotely requested to trigger such process. This use case only considers the case of a vehicle involved in an accident independently of the ITS infrastructure.

6.3.6 Normal Flow

1) The ITS-S in the vehicle that is directly involved in an accident detects a crash has happened, and then starts an accident reporting process automatically.

2) An accident reporting process may also be started manually. For example, a driver of a vehicle that is passing by the accident site stops and then manually starts an accident reporting process through giving a command to the ITS-S in his vehicle.

3) The ITS-S first takes some pictures with its digital camera, and then uses these pictures together with current time and geographical coordinates to generate an accident report. This report shall be signed by the ITS-S.

4) The ITS-S tries to connect to M2M Platform and then sends the accident report to the M2M Platform. (step 1 in Figure 6.3.9-1).

5) There are two ways for an ITS-S to connect to the M2M Platform. One is through wireless network; the other is through a nearby RSU using DSRC.

6) The M2M Platform receives and verifies the accident report, and then does some necessary analysis. The analysis result will be pushed to the subscribers, i.e. the Police Station and the Rescue Center.

7) The subscribers receive, verify and parse the information coming from M2M platform, and then do some necessary analysis. Based on different situation the subscribers may ask the M2M Platform to provide further information.

8) In this scenario the Police Station asks the M2M Platform to provide an overview of the traffic situation near the accident site, and the Rescue Center asks the M2M Platform to provide more visual information about the accident. These service requirements are submitted to the M2M Platform.

9) The M2M Platform receives and verifies the service requirements from Police Station and Rescue Center, and then sends data collection commands to the ITS-S that originally sends the accident report. (step 2 in Figure 6.3.9-1).

10) The command generated for Police Station requires the ITS-Ss near the accident site to report their travel status.

11) The command generated for Rescue Center requires the ITS-Ss around the accident site to provide pictures.

12) The ITS-S that originally sent the accident report receives the commands sent from the M2M Platform. It verifies and parses the commands, and then broadcasts the commands that should be broadcasted. (step 3 in Figure 6.3.9-1).

13) In this scenario the broadcasted commands are generated by the M2M platform for Police Station and Rescue Center respectively.

14) The ITS-Ss nearby the accident site receive, verify, parse and execute received commands, i.e. take pictures, get current travel status, generate reports, sign the reports and upload signed reports to M2M Platform. These reports could be sent anonymously. (step 4 in Figure 6.3.9-1).

15) Some commands need to be rebroadcasted within a predetermined area and predetermined period of time. (step 5 in Figure 6.3.9-1).

16) In this scenario the command generated for the Police Station needs to be rebroadcasted. The ITS-Ss receiving this command will only report their travel status. (step 6 in Figure 6.3.9-1).

17) M2M Platform accumulates and verifies the reports uploaded by the ITS-Ss, and then generates a report containing visual information about the accident scene for the Rescue Center and a report about traffic situation near the accident site. These reports will be pushed to Rescue Center and Police Station respectively.

18) The Rescue Center analyses the report about the accident scene, and then makes a proper rescue plan. The Police Station analyses the report about traffic situation, and then makes a proper travel control plan.

6.3.7 Alternative Flow

None.

6.3.8 Post-conditions

Based on the detailed information provided by the ITS service on the M2M platform, the rescue team can make a proper rescue plan, and the police can make a proper travel control plan.

6.3.9 High Level Illustration

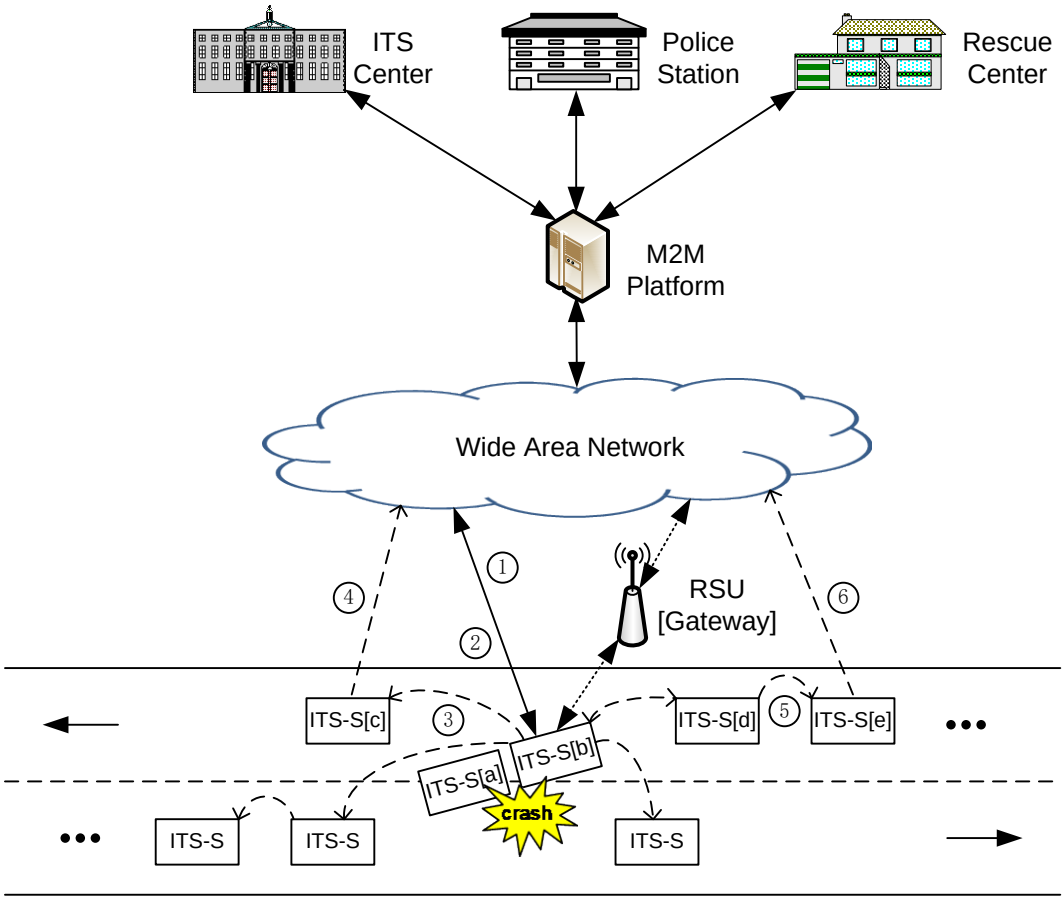

Figure 6.3.9-1: High Level Illustration of Traffic Accident Information Collection

6.3.10 Potential Requirements

1) A M2M System shall support communication between M2M Platform and a M2M device either directly or via a gateway.

2) A M2M System shall be able to exchange information between M2M applications via M2M Platform.

3) A M2M System shall be able to take actions according to the received service requests from M2M Applications.

4) A M2M system shall be able to support service requests from M2M applications for communication with QoS requirement, such as, higher delivery priority, reliable delivery, etc.

5) A M2M System shall support mutual-authentication among M2M device, M2M gateway, M2M platform and M2M Application ([i.2] SER-040).

6) The information sent by a M2M device or the M2M platform or a M2M application shall use cryptographic technology to ensure information authentication and information integrity.

7) A M2M system shall permit information being provided in anonymous way.

8) A command issued by a M2M System shall be able to have time expiration or geography restriction.

6.4 Fleet Management Service using DTG (Digital Tachograph)

6.4.1 Description

"DTG-based fleet management service" is the fleet management services utilizing DTG data and related service, to facilitate extensive service features of fleet management.

DTG provides vehicle data such as driving speed, RPM (Revolution Per Minute), brake's status, and mileage, etc.

DTG data management service, based on M2M gateway and DTG data management server, reports and manages DTG data in real-time to store it in the memory of M2M device in vehicle at a certain rate (i.e. one second in this case) to submit it to the national authority or transfer it to central office managing the data in a server.

The fleet management service utilizing the above mentioned service functionality provides advanced service features such as the precise quest of vehicles based on location and the tracking of cargo along with the route of the carrier vehicle, by means of the capability of remote monitoring and control of vehicle status provided by the DTG data management service.

6.4.2 Source

oneM2M-REQ-2013-0219R01 Use case - Fleet management using DTG.

6.4.3 Actors

- DTG device manufacturer to provide DTG devices and DTG management system.
- M2M device manufacturer to provide M2M gateway and related functionalities.
- The service provider for fleet management service using DTG.
- The network provider supporting the communication for fleet management service.
- The national agency that manages and operates DTG data (in case of Korea).

6.4.4 Pre-conditions

- The DTG device records the DTG data occasionally or periodically to transfer it to an application server through M2M Gateway.
- M2M service gateway delivers the DTG data, useful to fleet management service, from terminal system to the application server.
- Application server provides fleet management service, using DTG data, to customer.
- A taxi call service provider operates fleet management service using DTG data, such as for reporting the taxi location and passenger status and for call arrangement.

- A bus traffic service provider operates fleet management service using DTG data, including for providing guide information on bus arrival/estimated time, bus schedules on web-site, and status information such as route and air pressure of tire, etc.
- A fleet management service provider of truck operates fleet management service based on vehicle information (location, route, gas, tire pressure, etc.) and peripheral device information (temperature, humidity, door lock and goods weight, etc.)

6.4.5 Triggers

The following triggers could initiate the information exchanging process according to the flows described hereafter followings:

- Creation of DTG data that M2M device occasionally or periodically transfers to an application server.
- Arrangement of taxi service calls delivered to a DTG device.
- Report of information about vehicle location and route to application server.

6.4.6 Normal Flow

DTG service (Common service):

- DTG data is periodically (normally once in a second in this case) transferred and stored into DTG management server, and in case of an accident event, the data is stored at an immediate mode (within 10ms in this case).
- The DTG data stored in DTG device will be transferred to DTG management server periodically, and once after the engine stopped.
- The DTG management server stores DTG data and accident event file which is to be posted onto the web site of national agency.
- Analysis of DTG data and accident data to provide driving behavioural habits (quick start/stop, excessive speed) or the accident causes.

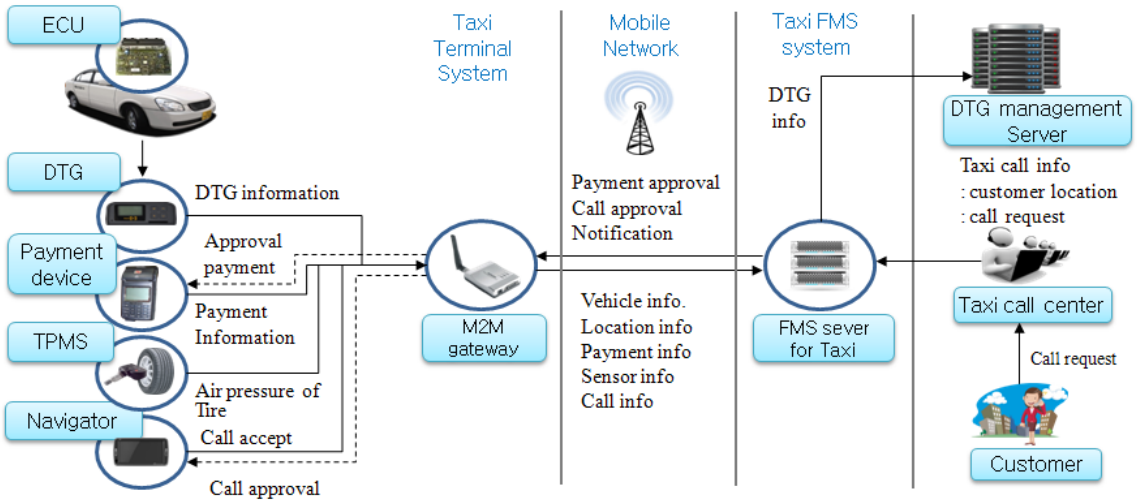

Figure 6.4.6-1: Taxi Call Service Normal Flow

Taxi call service:

- Terminal system occasionally or periodically reports location, passenger status information to application server (FMS server).
- Customer requests taxi call service to the taxi call l through a phone call or smart phone application.
- Taxi call center sends call request to a terminal system in the taxi through the application server.

- The taxi driver accepts the call request through the terminal system, and then the taxi will come to the customer's location.

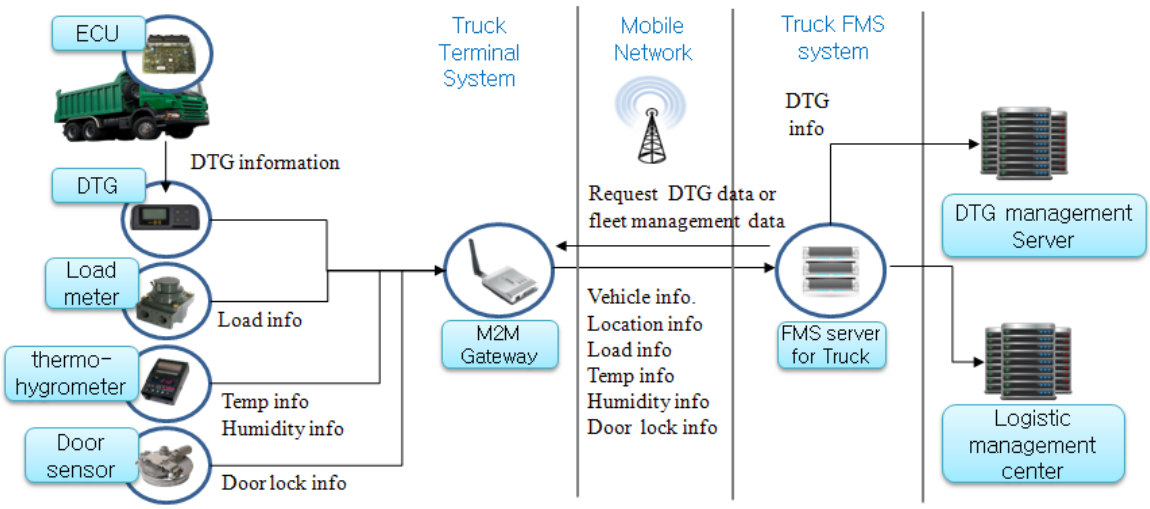

Figure 6.4.6-2: Normal Flow - Fleet Management Service (Truck)

Fleet Management Service (Truck):

- Terminal system occasionally or periodically reports the vehicle status information including the location, current route, ignition status, terminal version, and driver information to application server (FMS server).
- When the application server receives the information, it delivers it to logistic management center.
- Terminal system also reports the peripheral information (air pressure of tire, gas gauge, temperature, humidity, door lock, etc.) to the logistics management center through the application server.
- Logistic management center can request the information about vehicle itself or peripheral device, to enforce possible controls to them when it is needed.
- Terminal system reports the emergency events, such as fire in car, unlocked doors when unattended, and puncture while driving, etc. to FMS server

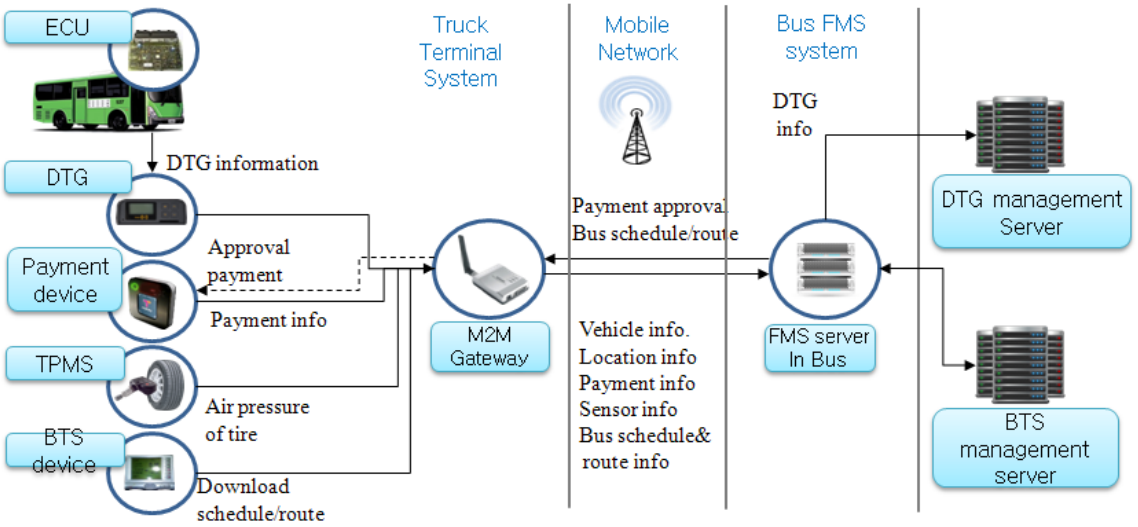

Figure 6.4.6-3: Normal Flow - Fleet Management Service (Bus)

Fleet Management Service (Bus):

- When the application server receives the vehicle information (engine ignition, terminal version, car S/N, and driver ID, etc.) from terminal system, it provides the received information to the BTS management server.

- BTS management server sends time schedule, route of bus and the fare information to terminal system through the application server (FMS server).
- Terminal system sets the time schedule, the route, and the fare information. And then it occasionally or periodically reports its location and the driving route to application server.
- Terminal system also reports the information about peripheral devices such as air pressure of tire, gas gauge level, and bus fare status to BTS management server occasionally or periodically.
- BTS management server provides an arrival/estimated time and a bus schedule on web-site.

6.4.7 Alternative Flow

None.

6.4.8 Post-conditions

None.

6.4.9 High Level Illustration

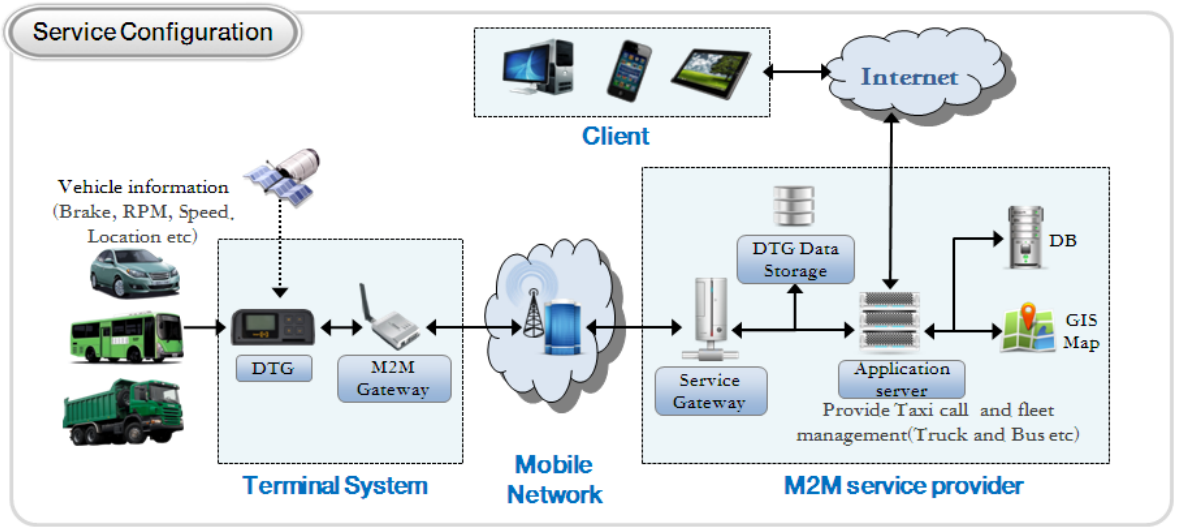

Figure 6.4.9-1: High Level Illustration Fleet Management

6.4.10 Potential Requirements

1) Provisioning, installation, configuration and registration method of terminal system:
   - Especially for the case of overlapping two different system for DTG management system (owns and manages the device) and the application system using DTG data (utilizing the data from the device).
2) DTG/FMS data storing method and delivery protocol:
   - There is no dominant standard specifying data formats and protocols for vehicle related applications.
3) Vehicle location based service method:
   - M2M service platform is expected to provide the service capability supporting location based service.
4) Control, configuration, error logging, and management method for the terminal system Over-The-Air:
   - M2M service platform is expected to provide the service capability supporting the Over-The-Air management.

6.5 Use cases for Electronic Toll Collection (ETC) service

6.5.1 Description

ETC is an important part of Intelligent Transportation System (ITS). It is vigorously promoted in many countries.

ETC aims to eliminate the delay on toll roads by collecting tolls electronically. ETC determines whether the vehicles passing are enrolled in the program, alerts enforcers for those that are not, and electronically debits the accounts of registered vehicle owners without requiring them to stop.

In the ETC Use Case, On Board Unit (OBU) is a dedicated device located in the vehicle and it can communicate with the local RSUs only. Stated differently, the communication between OBU and ETC platform is via the RSUs. With a vehicle moving, the OBU would connect to the next RSU and release connection with the previous RSU. It is necessary for the OBU to 'register' with ETC Service Platform for receiving M2M system services. The term 'registration' here refers to an OBU having its contextual information available at the ETC Service Platform. Such registration information is for the duration of an OBU being subscribed to services from the ETC Service Platform. At the same time, OBU 'registers' with local RSUs also, as OBU connects to different RSUs. Such registration context at the RSUs is temporary, and gets released as the OBU moves out of the range of the RSU and connects to the next RSU. The OBU uses its connectivity with local RSUs for achieving communication with ETC Service Platform.

As regards the 'registration context' between the OBU and the ETC Service Platform, the contextual information needs to reflect the complete profile of the OBU at the ETC Service Platform. Such information can include OBU identity, credentials, service subscription information, payment history, account balance, etc. This is referred to here as 'full registration'.

As regards the 'registration context' between the OBU and the RSU, such contextual information is a subset of the contextual information at the ETC Service Platform for the said OBU. The information in this subset needs to be sufficient for the RSU to identify the OBU to the ETC Service Platform for the OBU to receive desired services without compromising any sensitive information to the RSU. This is referred to here as 'lightweight registration'.

6.5.2 Source

REQ-2014-0431R03 Use cases for Electronic Toll Collection (ETC) service.

REQ-2014-0449R02 Use cases for Electronic Toll Collection (ETC) service.

6.5.3 Actors

- Vehicle Owner enrols for ETC service.
- On Board Unit (OBU) is a M2M device used to store information such as identifier of the vehicle. The OBU typically does not communicate with the ETC Service Platform directly while receiving services. Direct communication between OBU and the ETC Service Platform may however be supported, out of band, for example for subscribing for ETC services.
- Road Side Unit (RSU) is a device which is an intermediate entity and is available to connect from OBU to the ETC Service Platform. Typical role of RSU can be implemented as an M2M gateway, and it may provide other functions as well (e.g. traffic light control, barrier control, etc.).
- ETC Service Platform is responsible for collecting the information regarding the OBU via the RSU, including information such as location information from the RSU.
- ETC Service Provider provides its own M2M services for the user (OBU) through the ETC Service Platform.

6.5.4 Pre-conditions

- Vehicle owner enrols his/her vehicle for ETC services.
- All RSUs have their respective context (registration information) available at the ETC Service Platform.

6.5.5 Triggers

A vehicle equipped with OBU drives close to an electronic road charging station which equipped with RSU.

6.5.6 Normal Flow

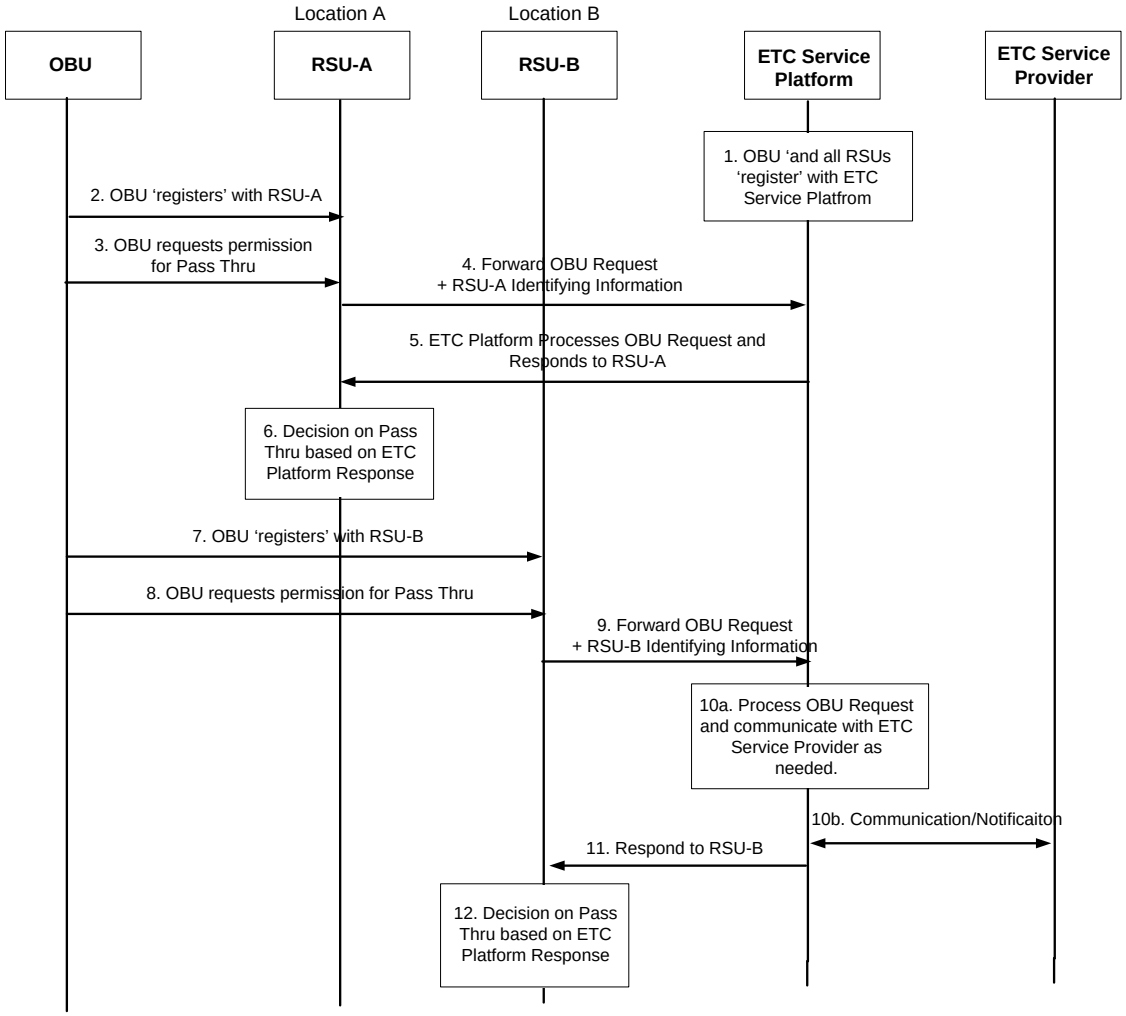

Figure 6.5.6-1: Normal Flow for Electronic Toll Collection (ETC) service

1) Vehicle owner subscribes to ETC services and the OBU is 'registered' with the ETC Service Platform. All RSUs are also registered with the ETC Service Platform.

NOTE: The term 'registration' refers to communication and resulting contextual information of the registering entity at the registered entity.

2) When the vehicle goes to a highway entrance (location A), OBU registers with RSU-A.

3) The Vehicle (OBU) sends a request to ETC Service Platform via RSU-A asking for permission to pass through.

4) RSU-A forwards the request from the OBU along with its own identifying information e.g. location information, to the ETC Service Platform.

5) ETC Service Platform processes the request for the OBU, received from RSU-A, and responds to RSU-A. The processing at ETC Service Platform can include charging the 'toll' to OBU account and updating account balance etc., for the OBU. The response to RSU-A includes the recommended action for the OBU for pass through.

6) RSU-A decides whether vehicle can pass through according to the response from the ETC Service Platform.

7) The vehicle drives to the highway exit at location B, and OBU registers to RSU at location B (RSU-B).

8) The vehicle sends a request to ETC Service Platform via RSU-B asking for permission for pass through.

9) RSU-B forwards the request for the OBU along with its own information e.g. location information to the ETC Service Platform.

10) ETC Service Platform processes the request for the OBU received from RSU-B. The processing at ETC Service Platform can include charging the 'toll' to OBU and updating account balance etc. for the OBU. The ETC Service Platform may communicate with the ETC Service Provider, as needed, while processing the request from the OBU.

11) ETC Service Platform responds to RSU-B by including information such as the recommended action for the OBU to pass through.

12) RSU-B decides whether vehicle can pass through according to the response from the ETC Service Platform.

6.5.7 Alternative Flow

None.

6.5.8 Post-conditions

None.

6.5.9 High Level Illustration

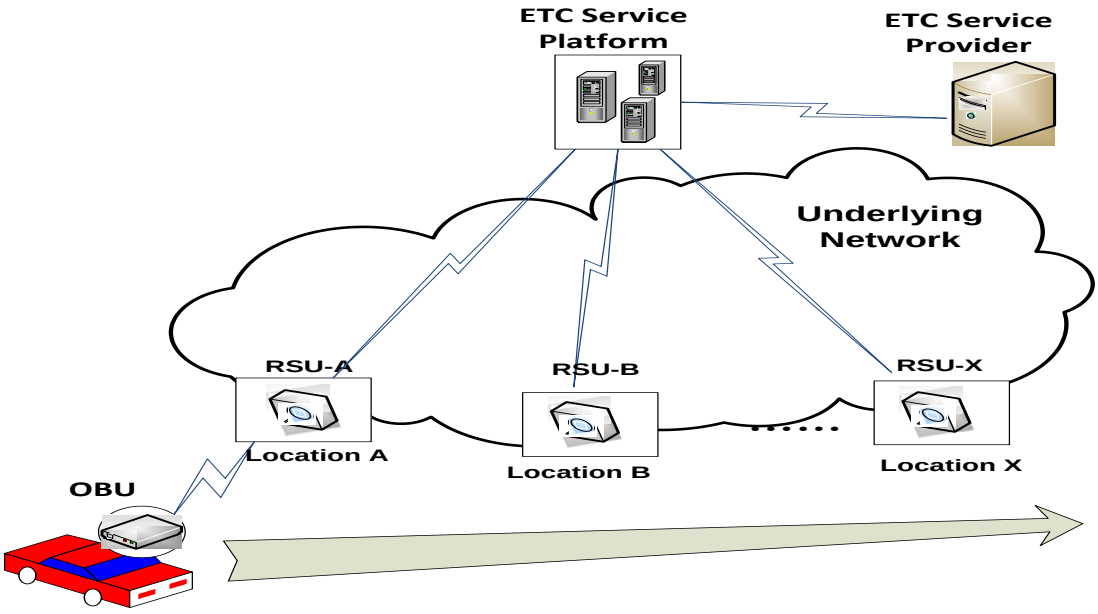

Figure 6.5.9-1: High Level Illustration for Electronic Toll Collection (ETC) service

6.5.10 Potential Requirements

1) The M2M System shall provide the capability for an M2M device to maintain registration with multiple entities simultaneously.

2) The registration shall be able to include information that identifies the peer entity, and other information necessary for the establishment of the respective peer relationships (e.g. management privilege, subscription).

3) The M2M System shall be able to hold the complete set of information context about the peer entity for some registrations (i.e. "full registration").

4) The M2M System shall be to hold only a subset of information context about the peer entity for some registration (i.e. "lightweight registration").

5) The M2M System shall be able to perform "lightweight registration" at different entities pertaining to a common peer entity and to hold different sets of information about the common peer entity.

6) The M2M System shall be able to correlate the "full registration" and the "lightweight registration" pertaining to a common peer entity.

7) The M2M System shall be possible to distinguish the "full registrations" and the "lightweight registrations" pertaining to a common peer entity.

8) The M2M System shall enable each of many M2M devices to perform mutual authentication with each of many M2M gateways.

9) The M2M System shall enable mutual authentication of M2M devices and M2M gateways in a timely manner.

6.6 Use cases for Taxi Advertisement

6.6.1 Description

For the taxi advertisement device, it does NOT need to access to the network during the day when the network is busy (e.g. from 8:00 to 23:00), but need to access the network at night when the network is idle (e.g. from 23:00 to 8:00) to download a large number of advertisement data.

6.6.2 Source

REQ-2014-0467R02 Use case for taxi advertisement.

6.6.3 Actors

- Taxi advertisement device, which can download advertisement data from advertisement data server through M2M service platform and show to passengers.
- The M2M service platform, which can control the taxi advertisement device and its access to the network.
- Advertisement data server, which can provide advertisement data for the advertisement device to download.

6.6.4 Pre-conditions

The taxi advertisement device and the advertisement data server registered to the M2M service platform.

6.6.5 Triggers

The taxi advertisement device accesses to the advertisement data server though M2M platform to download data.

6.6.6 Normal Flow

- The taxi advertisement device accesses to the advertisement data server through M2M platform.
- The M2M service platform checks the time policy. If current time is permitted, it allows the device to access the advertisement data server. Otherwise, it denies access and instructs the taxi advertisement device not to access the advertisement data server until the time when access is permitted.
- When access is permitted the taxi advertisement device downloads data from advertisement data server.

6.6.7 Alternative Flow

None.

6.6.8 Post-conditions

None.

6.6.9 High Level Illustration

Figure 6.6.9-1: High Level Illustration of Taxi advert use case

6.6.10 Potential Requirements

1) The M2M service platform shall be able to support the time-based policies to access the Underlying network ([i.2] CMR-014).

6.7 Use Case on Vehicle Data Service

6.7.1 Description

This use case introduces several services based on various data collected by sensor devices via smart vehicle (vehicle with on-board communication system) which is regarded as M2M gateway.

The sensor devices may be located in the vehicle but may also be located outside the vehicle, e.g. on the road side. Some sensor devices are equipped with M2M area network module and measure individual data. The smart vehicle connects to the sensor devices and collects data from sensor devices by using the M2M area network technology such as Wireless LAN, ZigBee, Bluetooth, etc., and sends the data to application server in infrastructure domain via mobile network.

Management server and Application Server in the M2M infrastructure domain connect to the smart vehicle via a mobile network in order to control its configurations updating software and exchanging M2M data (e.g. updating a map).

It is important to observe that the smart vehicle as M2M gateway has mobility. For instance, there are possibilities for a mobile device to simultaneously connect too many sensor devices, and to newly connect to sensor devices which have never been connected before.

This use case illustrates potential requirements from the use case of services utilizing mobile device.

6.7.2 Source

REQ-2014-0472R06: Use Case on Vehicle Data Services.

6.7.3 Actors

- M2M Device: In-vehicle sensor device and outdoor sensor device. In-vehicle sensor includes on-board (built-in) sensor used for monitoring of machine health and also sensor on user carry-on device, such as smart phone and wearable device like Fitbit™, Nike Fuelband™, Sony SmartBand™ for example. In-vehicle sensor may monitor status of machine health (diagnosis), mobility, passenger's health and environment. Outdoor sensor device may be located outdoor and monitor roadway infrastructure, agriculture, property (surveillance) and utility (telemetering), for example. It may be equipped with several kind of communication protocol.

- M2M Area Network: Area network which connects M2M Device with M2M Gateway and also provides connectivity among M2M Gateways. It may include Wireless LAN, Bluetooth, ZigBee and Ethernet.
- M2M Gateway: On-board communication system equipped on the smart vehicle, which communicates with M2M Devices and other M2M Gateways via M2M Area Network and also communicates with Application Servers and Management Servers via Mobile Network.
- Mobile Network: Network which has functions to transfer data and control messages between M2M Gateway and M2M Application Server/Management Server. It may include cellular base station.
- M2M Management Server: Server which manages M2M applications in M2M Gateway and M2M Device by installing, uninstalling and updating them.
- M2M Application Server: Server which maintains database and provides the data access services such as accepting data publication from and issuing data subscription to M2M Gateway/Device. This server also manages non-M2M applications such as navigation system as well as contents such as map data in M2M Gateway and M2M Device by installing, uninstalling and updating them.

6.7.4 Pre-conditions

- It is possible to establish a connection among M2M Application Server, M2M Management Server and Smart Vehicle via Mobile Network.
- It is possible to establish a connection between Smart Vehicle and M2M Device and among Smart Vehicles via M2M Area Network.
- The M2M Gateway has been configured by the M2M Management Server.
- The M2M Device has not yet been configured by the M2M Management Server.

6.7.5 Triggers

Subject to capabilities of the M2M Area Network, the M2M Device detects a M2M Gateway that can be associated, or vice versa. The association may require explicit operation (permission) for that association by users (e.g. owners of Gateway and/or Device) or administrator (a manufacturer of Gateway and/or Device and/or Management Server). Examples are below:

- Sensors are built into the smart vehicle at the factory (Permanent Association, no triggering is needed).
- Sensors are equipped to the smart vehicle after market, e.g. at auto dealers or auto parts stores. Triggering occurs when the smart vehicle is being equipped with the new sensor.
- Users bring sensors with user carry-on devices into the smart vehicle.
- The smart vehicle detects nearby outdoor sensors.
- The smart vehicle detects other smart vehicles nearby.

6.7.6 Normal Flow

M2M Configuration step: Management Server configures M2M Device.

Upon triggering, i.e. when the M2M Device detects a M2M Gateway that can be associated or the M2M Gateway detects a M2M Device that can be associated the following sequence is initiated:

- M2M Gateway establishes a connection to M2M Device (or the M2M Device establishes the connection to the M2M Gateway) via the M2M Area Network.
- M2M Device sends its attribute information (e.g. type of device, service certificates of the device, required application software, etc.) to the M2M Gateway.
- M2M Gateway establishes a connection to the M2M Management Server via Mobile Network.

- M2M Gateway relays the attribute information to Management Server.
- M2M Management Server provides M2M Gateway with the appropriate software and configuration data for the M2M Device.
- M2M Gateway relays the software and configuration data to M2M Device.
- M2M Device configures itself according to the software and configuration data.

Transmission of data between M2M Device and M2M Application Server:

- Data Publication of M2M Device.
- M2M Gateway (re-)establishes a connection to M2M Device via M2M Area Network.
- M2M Device publishes measured data to M2M Gateway.
- M2M Gateway establishes a connection to M2M Application Server via Mobile Network.
- M2M Gateway relays the data to M2M Application Server. M2M Gateway may distribute the data (i.e. data publication) to multiple M2M Application Servers whose data subscriptions have been previously grouped/aggregated at M2M Gateway.
- M2M Application Server receives the data.

Data Subscription from M2M Application Server:

- M2M Gateway establishes a connection to M2M Application Server via Mobile Network.
- M2M Application Server sends M2M Gateway attribute information of data in need.
- M2M Gateway (re-)establishes a connection to M2M Device via M2M Area Network.
- M2M Gateway relays the attribute information to M2M Device. M2M Gateway may group/aggregate the attribute information (i.e. data subscription) from multiple M2M Application Servers.
- M2M Device publishes measured data that meets the attribute information to M2M Gateway.
- M2M Gateway relays the data to Application Server. M2M Gateway may distribute the data (i.e. data publication) to multiple M2M Application Servers whose data subscriptions have been previously grouped/aggregated at M2M Gateway.
- Application Server receives the data.

Update of non-M2M Application 1: M2M Application Server and M2M Gateway:

- M2M Gateway establishes a connection to M2M Application Server via Mobile Network.
- M2M Gateway sends its attribute information to M2M Application Server.
- M2M Application Server provides M2M Gateway with the appropriate software, its configuration data and contents such as map.
- M2M Gateway configures itself according to the software, configuration data, etc.

Update of non-M2M Application 2: M2M Application Server and M2M Device:

- M2M Device establishes a connection to M2M Gateway via M2M Area Network.
- M2M Device sends its attribute information to M2M Gateway.
- M2M Gateway establishes a connection to M2M Application Server via Mobile Network.
- M2M Gateway relays the attribute information to M2M Application Server.
- M2M Application Server provides M2M Gateway with the appropriate software and configuration data.

- M2M Gateway relays the configuration software and data to M2M Device.
- M2M Device configures itself according to the software and configuration data.

6.7.7 Alternative Flow

Alternative Flow 1

This alternative flow may occur in the case where the M2M Gateway only occasionally connects to devices and servers - e.g. via M2M Area networks that can only occasionally be used:

- M2M Gateway Opportunistic Communication (Store and Forward):
  - M2M Gateway may store data that are destined to M2M Device, M2M Management Server and M2M Application Server.
  - M2M Gateway may send the stored data to other M2M Gateway.
  - Both, the originator and the other M2M Gateway may deliver the stored data to the destination when connected.
  - M2M Gateway may erase the stored data that has been already sent to the destination.

Alternative Flow 2

This variant flow may occur in the case where the M2M Gateway processes the data flow between M2M, Gateway and M2M Application Server:

- M2M Gateway Data Processing:
  - M2M Gateway may aggregate, statistically summarize (e.g. average) and/or erase data based on criteria that are indicated in the data subscription or the configuration data from M2M Application Server.
- M2M Gateway Data Subscription:
  - M2M Application Server may be interested in joint changes to multiple resources (either on the same M2M Device or multiple M2M Devices). In other words, M2M Application Server makes a data subscription where automatic notification (i.e. data publication) depends on two or more resources, not a single resource. Notifications are then generated when the expected changes occur within each of the resources.

Alternative Flow 3

This variant flow may occur in cases where the M2M Gateway broadcasts its interest to subscribe to specific data to all M2M Devices in its vicinity:

- M2M Gateway Cross Layer Optimization:
  - M2M Gateway may indicate a full or part of data subscription (attribute information of data in need) in wireless pre-association information such as beacon.
  - Smart Vehicle and Wi-Fi Hotspot may periodically broadcast beacon that contains the subscription in order to prevent sensor devices from establishing unnecessary connection and wasting radio resources as well as battery power.

NOTE: M2M Application Data Security (TBD).

6.7.8 Post-conditions

- M2M Application Server stores data and provides data access service via API for user applications.
- M2M Devices and M2M Gateways are well maintained by M2M Management Server and M2M Application Server.

6.7.9 High Level Illustration

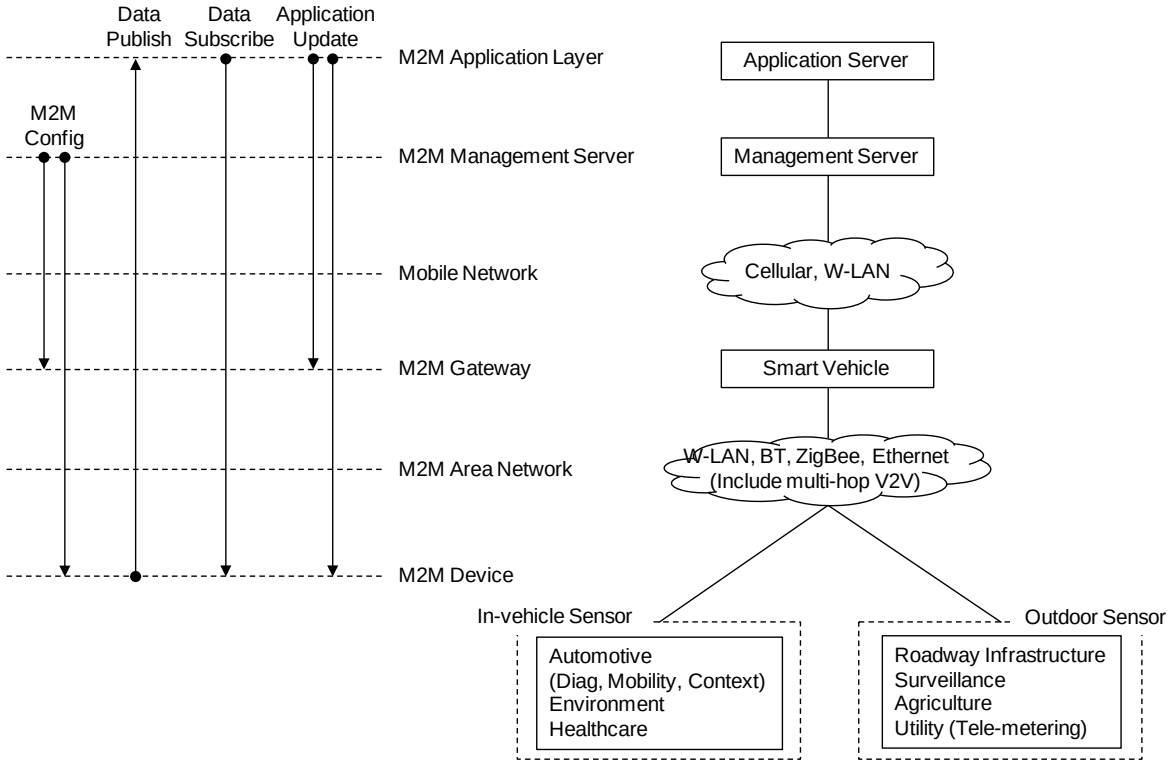

Figure 6.7.9-1 High Level Illustration - Vehicle Data Service

6.7.10 Potential requirements

1) The oneM2M System shall enable discovery of M2M Application Servers, M2M Management Servers and M2M Devices available to an M2M Gateway for data exchange ([i.2] OSR-086).

2) The oneM2M System shall enable discovery of M2M Gateways available to a M2M Management Server and an M2M Device for data exchange ([i.2] OSR-087).

3) The oneM2M System shall be able to support the capabilities for data repository (i.e. to collect/store) and for data transfer from one or more M2M Devices or M2M Gateways, for delivery to one or more M2M Gateways via M2M Area Network without any assistance or instruction of M2M Management Servers and M2M Application Servers ([i.2] OSR-088).

4) Upon request from M2M Application Server, an M2M Gateway or M2M device shall enable functions that pre-process (e.g. average) M2M data before providing them to the recipient ([i.2] OSR-104).

5) Upon request, an M2M Gateway or M2M device shall enable functions that erase M2M data (e.g. that have been sent or could not be sent to the recipient within a certain time) based on criteria from an M2M Application ([i.2] OSR-105).

6) An M2M Gateway and/or an M2M Device shall be able to broadcast to all M2M Devices and/or M2M Gateways in the vicinity its need to receive/deliver specific data ([i.2] OSR-106).

7) M2M Gateway and/or M2M Device shall be able to establish a connection to each other if it is able to receive/deliver the required specific data ([i.2] OSR-107).

8) The oneM2M System shall enable M2M Gateways to set conditions used for processing jointly data subscriptions and distribute the resulting notifications according to the set conditions ([i.2] OSR-108).

9) The oneM2M System shall enable subscriptions to changes to multiple resources which aim to generate notifications if and only if the expected changes to those resources occur concurrently ([i.2] OSR-109 and OSR-110).

6.8 Smart Automatic Driving

6.8.1 Description

Attention to automatic driving is increasing. An automatic driving technology normally superimposes dynamic information gathered by car-mounted sensors and/or cameras on a static high-definition map. It makes possible to calculate the way which is appropriate for the vehicle to take.

Because an automatic driving vehicle gathers dynamic surrounding circumstances by sensors and/or cameras, the coverage of the vehicle's perception is normally almost a few hundred meters. Therefore, the automatic driving vehicle may be forced to make a sharp lane change, a sudden stop or a return of its control to a driver at the worst, in case of unusual state such as a crash, a road work or a dropping on the road.

In order to make a smooth lane change, a gradual slow down or a fully prepared return of its control to a driver even in such unusual states, the system which can collect current road conditions and feedback credible information to relevant vehicles is required. The information can be used by each vehicle to calculate the driving way such as a lane, path or speed.

6.8.2 Source

REQ-2015-0554-Smart Automatic Driving.

6.8.3 Actors

Vehicle Driving Support Center

It distributes high-definition maps to vehicles supporting an automatic driving. Furthermore, it collects unusual states from vehicles on the road, validates its credibility and feeds back credible information to relevant vehicles. If there are devices/GWs located at the roadside, some of the functions of vehicle driving support center may be provided via these devices/GWs.

OneM2M System

It connects between the vehicle driving support center and vehicles. Furthermore, the devices/GWs located at the roadside may provide some of the functions of vehicle driving support center.

Vehicles

The automatic driving vehicles have multiple on-board sensors and cameras for the automatic driving which superimposes dynamic information gathered by the sensors and/or cameras on a static high-definition map distributed by the vehicle driving support center. Furthermore, the automatic driving vehicles use unusual states information from the center to calculate the driving way such as a lane, path or speed.

6.8.4 Pre-conditions

1) Some vehicles support automatic driving.

2) The automatic driving vehicles have a mobile communication module and a function to notice their own location and on-board camera image to the vehicle driving support center.

6.8.5 Triggers

The automatic vehicles report occurrence of an unusual state such as a crash, a road work or a dropping on the road.

6.8.6 Normal Flow

1) The automatic driving vehicles reaching the location where the unusual state occurs detect the state by on-board sensors and cameras. As a result, the vehicles change a lane, slow down, stop or return the control to the driver, which become sudden operations depending on those situations. Additionally, the vehicles report the unusual state to the vehicle driving support center. The state is reported as on-board camera image with additional information such as current time and locations of the vehicle.

2) The center receiving those reports identifies the location where the unusual state occurs, compares the reports which seem to be sent from same location and validates its credibility.

3) The center notices the unusual state to the vehicles which are reaching the location of the unusual state after finishing the credibility validation. The notice is protected in an appropriate way such as digital signature, because this information is used by each vehicle to calculate the way of the automatic driving.

4) The automatic driving vehicles receiving the notice from the center make a smooth lane change, a gradual slow down or a fully prepared return of its control to the driver before reaching the location of the unusual state.

6.8.7 Alternative flow

None.

6.8.8 Post-conditions

None.

6.8.9 High Level Illustration

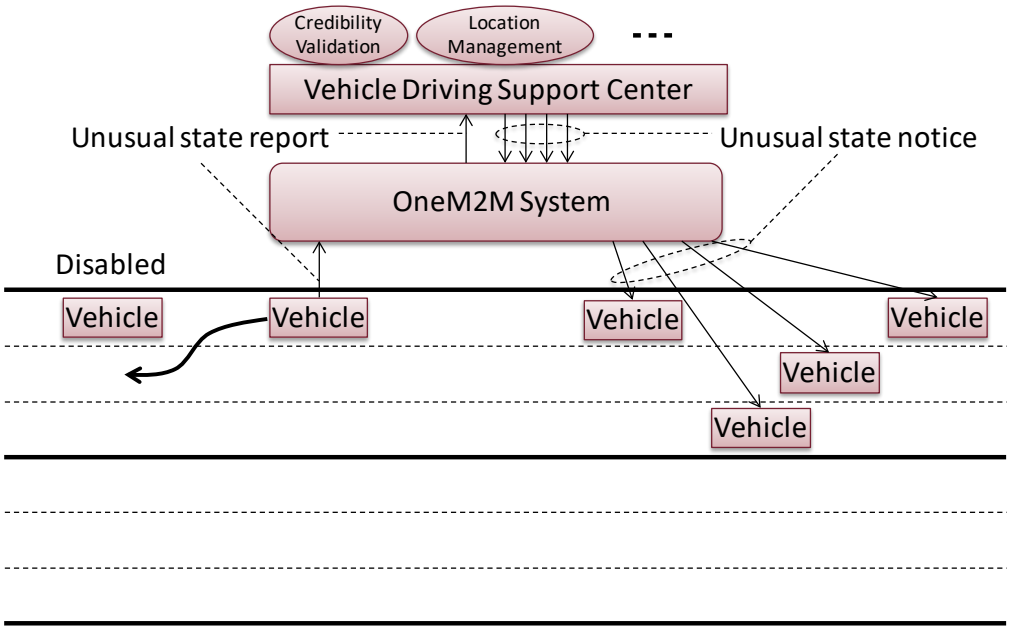

Figure 6.8.9-1: High Level Illustration of smart automatic driving

6.8.10 Potential requirements

1) OneM2M System shall be able to send the information to intended vehicles by unicast, multicast and/or broadcast.

2) OneM2M System shall be able to securely transfer the information by using an appropriate means such as digital signature.

3) OneM2M System shall be able to transfer the information on real-time basis for feeding back current road states to automatic driving control. The feedback time should be less than a few seconds (the distance between vehicles normally corresponds to a few seconds) to avoid unnecessary slow down/stop of following vehicles.

4) OneM2M system shall be able to guarantee its reliability in order to receive/feedback messages from/to related vehicles.

5) oneM2M System shall enable sharing of service information between devices/GWs based on proximity.

6) oneM2M System shall enable sending and receiving service information between devices/GWs with minimized interruption.

6.9 Use Case on Vehicle Data Wipe Service

6.9.1 Description

This use case introduces vehicle data wipe services in addition to the use case on vehicle data services as REQ-2014-0472R06.

Background: A data center on the cloud collects sensor data from vehicles using the mobile network (e.g. cellular and wireless LAN). The data may relate to diagnosis, mobility and context of vehicles. The diagnostic data are useful for vehicle design improvement and the mobility data for dynamic route guidance services. The contextual data are captured by stereo cameras and radar scanners in the automated driving system. And then those data may constitute the complete 3D roadway map which is also essential for the automated driving system in turn. In the use case of REQ-2014-0472R06, the data center, that is the M2M application server, requests data by sending a meta-data to vehicles. Then vehicles prepare data which meet the criteria of the meta-data and publish it to the server. The meta-data may describe attributes of requested data such as time period, geographical area, data type, statistic process options and so forth. Figure 6.9.9.2-1 shows how this simply works.

Problem statement: Though a certain volume of data (big data) obtains utility value, the bigger data gradually show less increase of the value (as is shown in Figure 6.9.9.3-1 and 6.9.9.3-2). For instance, no more samples are required for statistical analysis once the number of samples gets sufficient. Duplicated collections of collected pieces of map are also redundant. Such wasteful data deliveries consume multiple expensive resources of vehicles, mobile network, network backhaul and servers. Data in the vehicle may also lose its value when it gets delivered, obsolete, false or even malicious.

Solution: Such unwanted data collection needs to be cut off in order to spare those resources for other useful data delivery. In addition, unwanted data in a limited capacity of the vehicle storage need to be wiped out. These operations should be carried on under the right authorization.

6.9.2 Source

REQ-2015-0589R04-Usecase_on_vehicle_data_wipe_service.

6.9.3 Actors

- M2M Device: Data source node such as sensors in vehicles.
- M2M Gateway: Data relay node such as a vehicle on-board communication system with storage.
- M2M Application Server: Data requesting node such as a data center on the cloud.

6.9.4 Pre-conditions

The M2M Application Server has disseminated a data request with the meta-data to vehicle(s) (as is shown in Figure 6.9.9.4-1).

NOTE: The meta-data may describe attributes of requested data such as time period, geographical area, data type, statistic process options and so forth.

6.9.5 Triggers

The M2M Application Server decides to cancel the data request and erase the data in the vehicles. (One of the reason for this could be that the data has been collected.)

6.9.6 Normal Flow

1) The M2M Application Server transmits a wipe request with the same meta-data in the target data request to the vehicle(s).

2) On arrival of the wipe request, the vehicle deletes the meta-data and data which meet the criteria of meta-data from in its storage.

6.9.7 Alternative flow

- If it is additionally instructed in the wipe request, the wipe request can be delivered from the vehicle to other vehicles.
- If it is additionally instructed in the wipe request, the vehicle may only stop publishing data which meets the criteria of meta-data, instead of deleting.
- The M2M Application Server and vehicles may share a credential and use the credential to authenticate the wipe request (as is shown in Figure 6.9.9.6-1). The credential could be a pre-shared key or could be based on public key infrastructure. The credential could also be identified in the data request.

6.9.8 Post-conditions

There is no data relating to the cancelled data request in the vehicle storage.

6.9.9 High Level Illustration

6.9.9.1 Data Request and Response

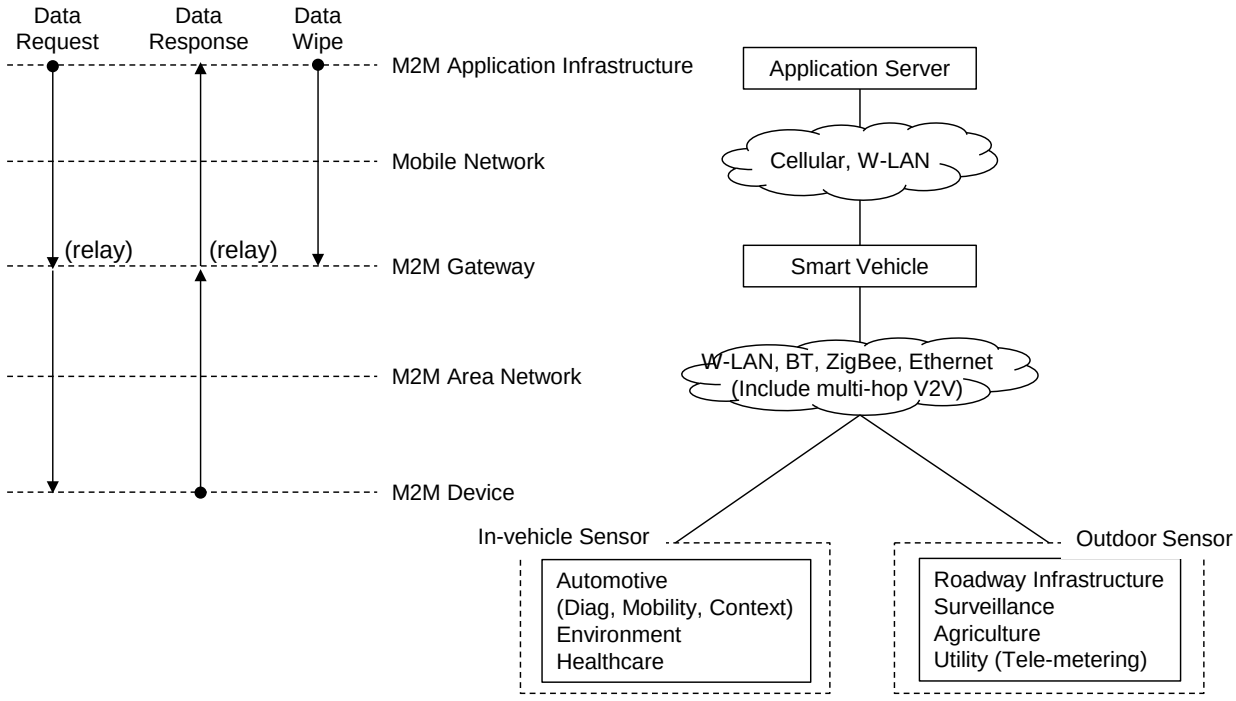

Figure 6.9.9.1-1 Deployment

6.9.9.2 Data Request and Response

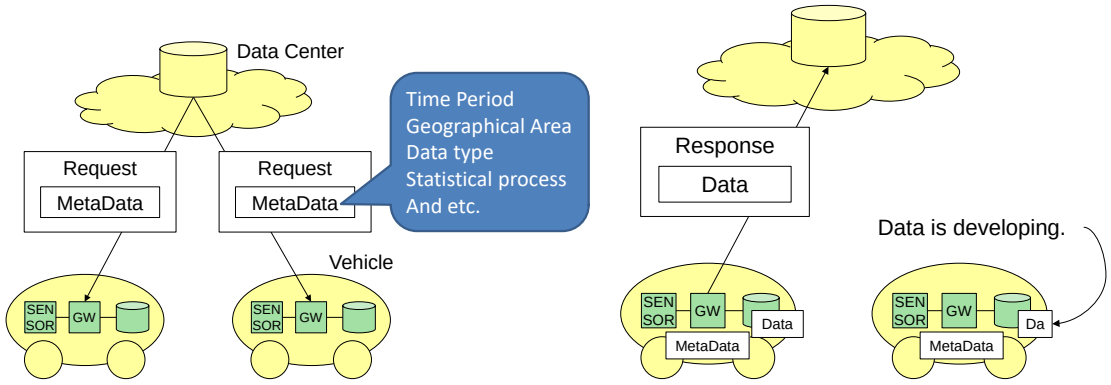

Figure 6.9.9.2-1 Data Request and Response

6.9.9.3 Issue of Bigger Data

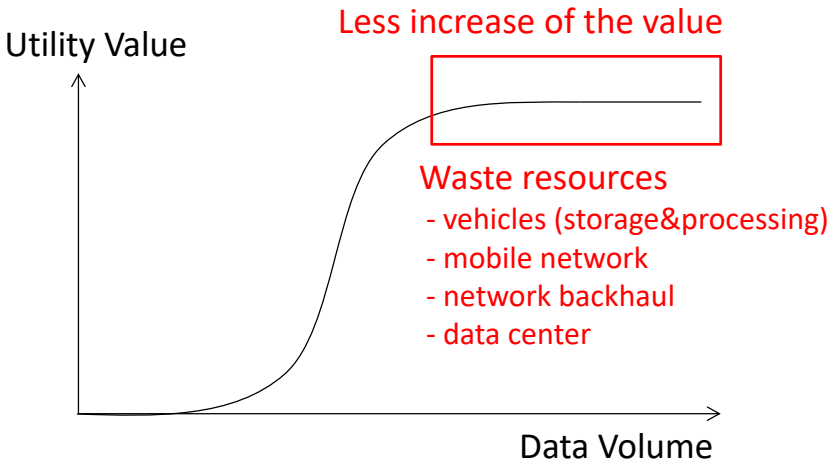

Figure 6.9.9.3-1 Less increase of value

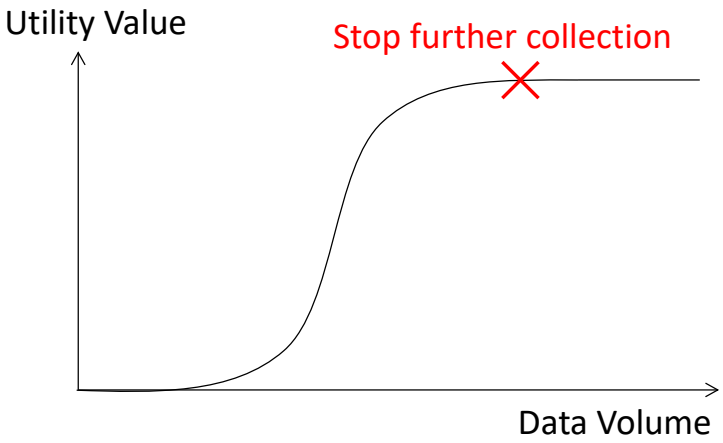

Figure 6.9.9.3-2 Stop further collection

6.9.9.4 Pre-condition of Data Wipe (and Post-condition of Data Request and Data Response)

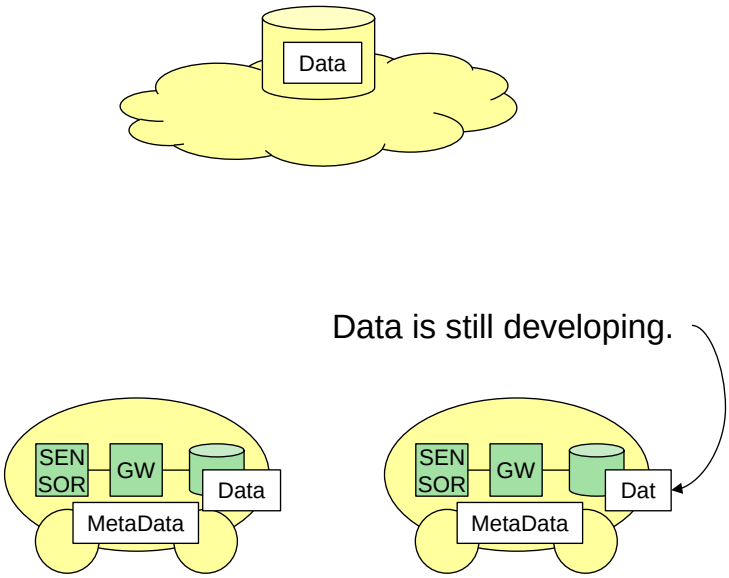

Figure 6.9.9.4-1 Pre-condition of data wipe

6.9.9.5 Data Wipe

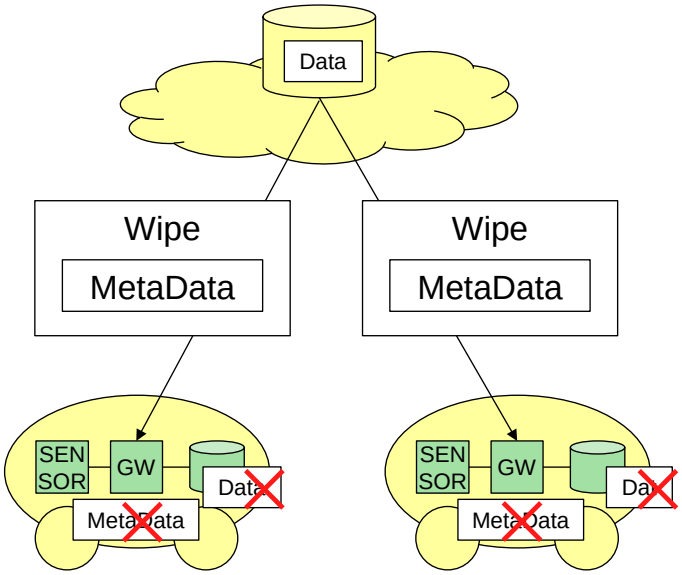

Figure 6.9.9.5-1 Data wipe

6.9.9.6 Data Wipe with Authentication

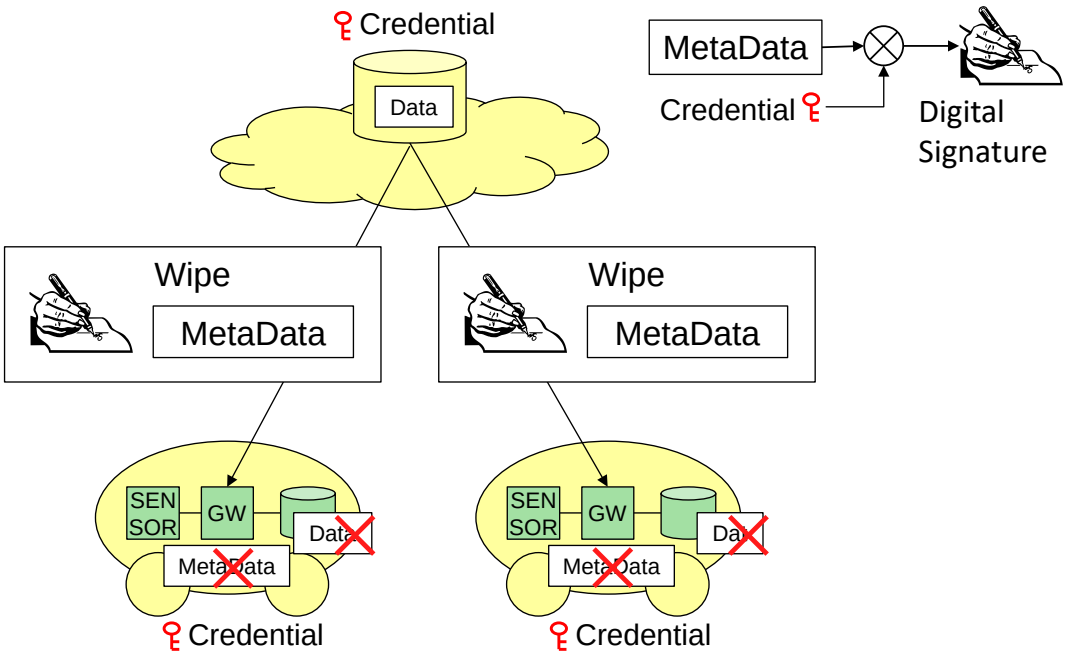

Figure 6.9.9.6-1 Data wipe with authentication

6.9.9.7 Post Condition of Data Wipe

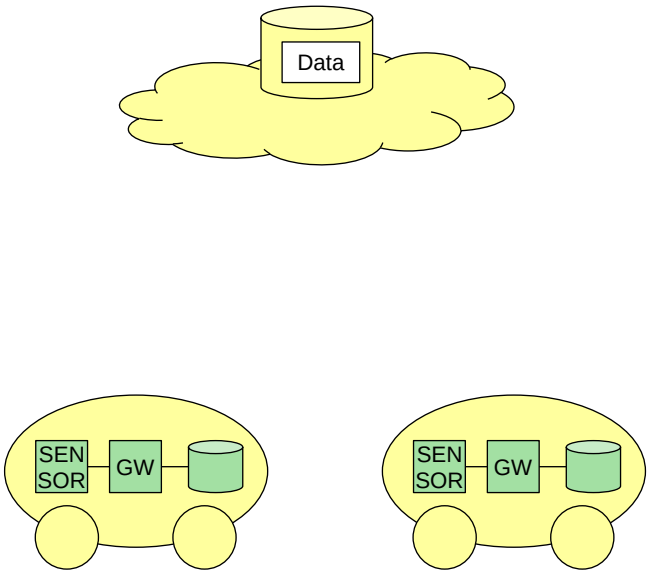

Figure 6.9.9.7-1 Post condition of data wipe

6.9.10 Potential requirements

1) The oneM2M System shall enable the cancellation of continuous data collection and/or the deletion of collected data when pre-defined conditions are met ([i.2] OSR-089).

2) The oneM2M System shall enable pre-defined conditions to be protected from unauthorized modification ([i.2] SER-050).

3) The oneM2M System shall enable the deletion of M2M data produced/stored by the M2M Devices/Gateways based on request from an authorized entity ([i.2] OSR-051).

6.10 Vehicle Management based on Geo-Fence

6.10.1 Description

Since a vehicle is a nomadic object, localization of vehicles and collecting information for management are crucial features for vehicle management.

From the holistic viewpoint, vehicle management is a generic term and may address various vehicle and smart car services such as diagnostic, maintenance and so on.

Specifically, in this use case, vehicle management based geo-fence feature is about how to monitor the location and movement of a target vehicle efficiently (e.g. judge where the movement is illegal or legal for example).

6.10.2 Source

REQ-2016-0002R01 Vehicle Management based on Geo-fence.

6.10.3 Actors

Vehicle

Vehicle is a nomadic object which is monitored by oneM2M infrastructure node or Location Server and is equipped with hardware devices for localization (e.g. GPS, Cellular modem).

oneM2M Infrastructure Node

A system exposes the service functions for vehicle management such as Location, Device management and potentially additional services.

Location Server

A Location Server can localize the target device(s) using collected measurements (e.g. RSSI, AP ID) and detect Geo-Fence event based on set configurations.

LBS/IoT Application

An Application requests Geo-Fence based location service toward oneM2M Infrastructure Node.

6.10.4 Pre-conditions

N/A.

6.10.5 Triggers

A LBS/IoT Application requires to set Geo-Fence-based Location-Based services depending its demand.

6.10.6 Normal Flow

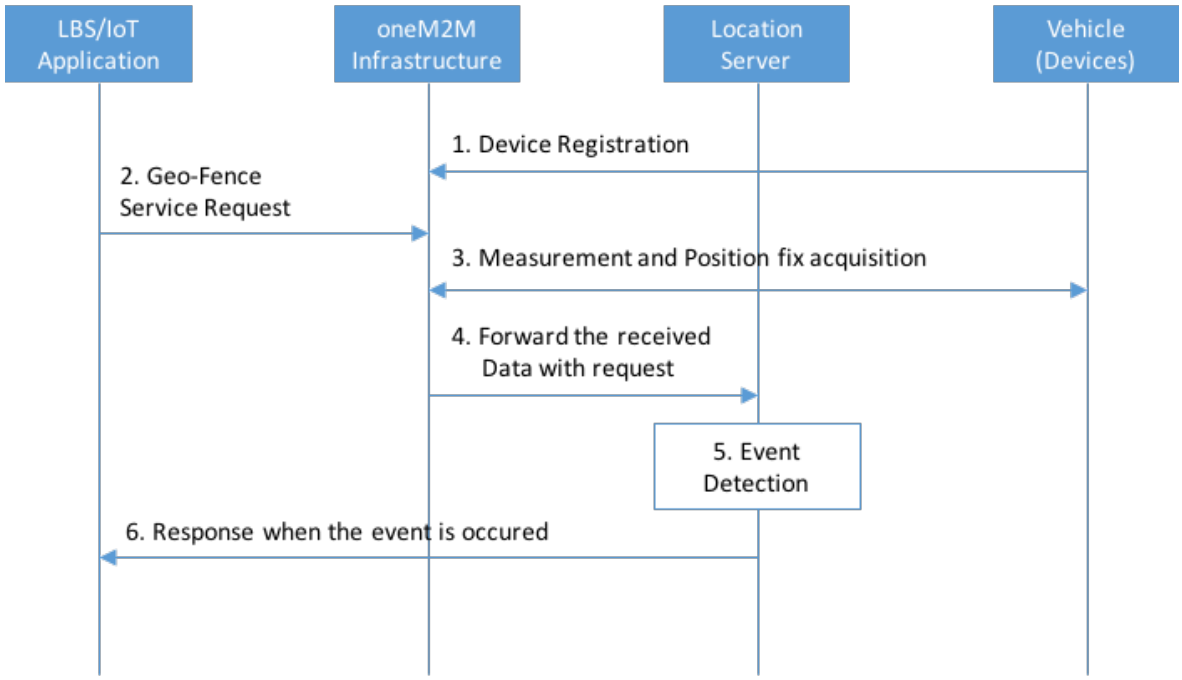

Figure 6.10.6-1 Normal Flow - Vehicle Management based on Geo-Fence

1) A vehicle Device registers with oneM2M Infrastructure.

2) A LBS/IoT Application requests the Geo-Fence Service to oneM2M Infrastructure. The Application set the configuration regarding Geo-Fence Service such as Geo-Fence area, event criteria, quality of event detection and so on.

3) oneM2M Infrastructure acquires Measurement and/or Position fix from the Vehicle devices.

4) The Measurement and/or Position fix are/is forwarded to Location Server by oneM2M infrastructure.

5) Location Server runs Geo-Fence detection engine and analyses whether the Geo-Fence event has occurred or not. The detection is operated following the configuration of application.

6) When any event has occurred, the event reports to the LBS/IoT Application.

6.10.7 Alternative flow

N/A.

6.10.8 Post-conditions

N/A.

6.10.9 High Level Illustration

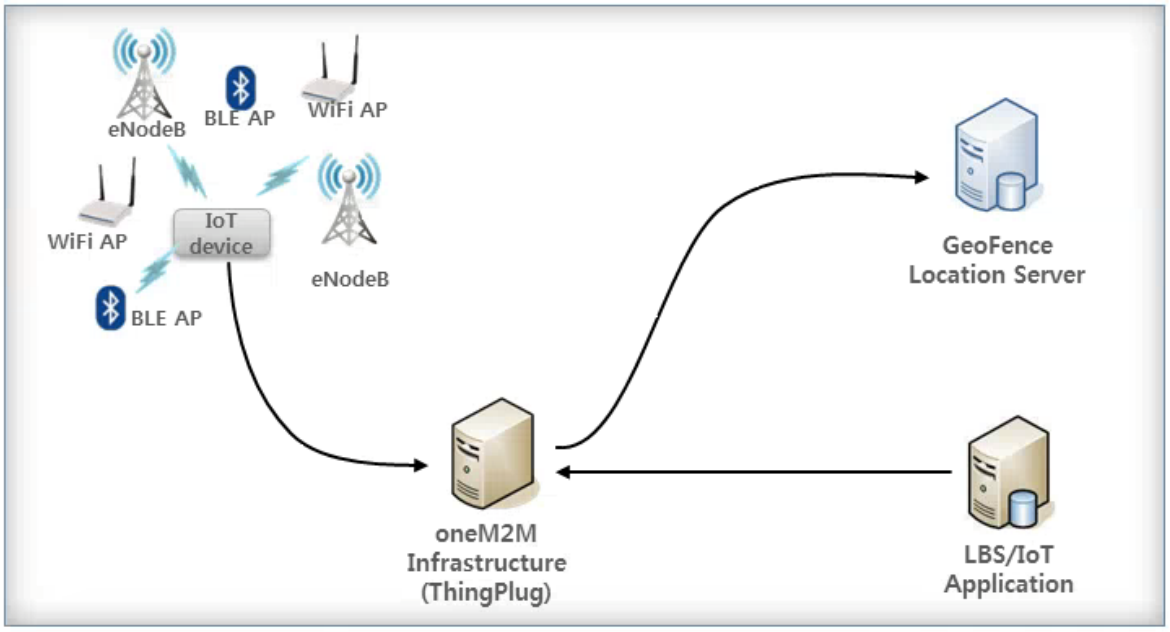

Figure 6.10.9-1 High Level Illustration - Vehicle Management based on Geo-Fence

- Vehicle Devices can gather the signal measurement and/or position fix.
- The Devices are registered and connected to oneM2M Infrastructure.
- GeoFence Server can detect any movement event based on the Application's configuration.

6.10.10 Potential requirements

1) The oneM2M System shall support reporting of Geo-Fence based Location event of the target M2M Device to the M2M Application based on the Application's configuration ([i.2] OSR-047).

2) The oneM2M System shall support the M2M Application setting the configuration for Geo-Fence based location service.

6.11 Use Case on Secure Over-The-Air Firmware Update for Automotive ECUs

6.11.1 Description

This use case introduces secure remote Over-The-Air (OTA) Firmware Update scheme for M2M Devices such as ECUs (Electronic Control Units) through on-board M2M Gateway in automobiles. An automobile which has an on-board M2M GW is called "Smart Vehicle".

Each M2M Device is connected to the on-board M2M Gateway Unit by M2M Area Network such as Controller Area Network (CAN).

M2M Management Server in Infrastructure Domain collects information on version of currently installed firmware in ECUs and distributes updated firmware dedicated for each M2M Device through the M2M Gateway using Mobile Network.

6.11.2 Source

REQ-2016-0012R01 Use Case on OTA Firmware Update for ECUs.

6.11.3 Actors

- M2M Device: Electronic Control Unit (ECU) in an automobile is assumed as an M2M Device. There are many kinds of on-board ECUs in automobile, e.g. Power train ECU, Transmission ECU, Brake ECU, etc.
- M2M Gateway (M2M GW): M2M GW is connected with M2M Devices to collect M2M Data and distribute firmware through multiple M2M Area Networks depending on kinds of functionality or security requirements. M2M GW is equipped with a data communication module to communicate with M2M Management Servers and external servers in Infrastructure Domain. Also, it has a display device in order to confirm user's authorization for the firmware update operation.
- M2M Area Network: CAN (Controller Area Network) or LIN (Local Interconnect Network) is used as wired M2M Area Network in an automobile.
- Mobile Network: Mobile Network is assumed to transfer M2M Data and messages between M2M GW and M2M Management Server.
- M2M Management Server: This server is utilized to conduct firmware update operation and management of version of firmware currently installed in M2M Devices.
- OEM: Original Equipment Manufacturer means automobile manufacturer who is ultimately responsible for the product (e.g. firmware for ECU).
- Firmware Developer: A developer of firmware dedicated for on-board M2M Device (ECU).
- User: A person who makes a decision for conducting firmware update of ECU, when a request for the update is notified from M2M Management Server. This assumes to be an owner of automobile.

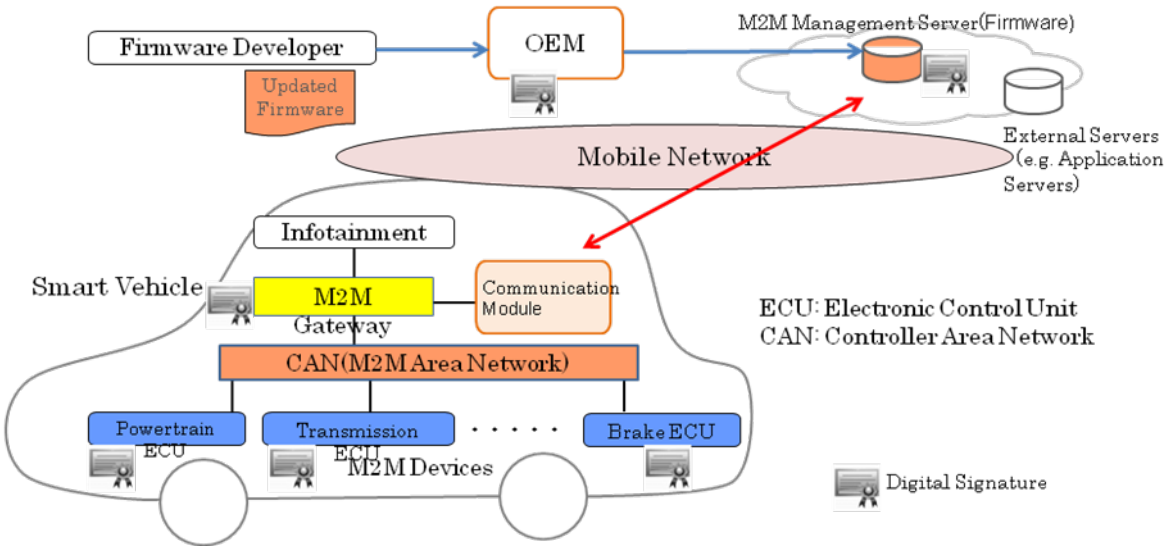

Figure 6.11.3-1: High Level Structure for Over-The-Air Firmware Update for Automotive ECUs

6.11.4 Pre-conditions

- M2M GW is connected to M2M Management Server via Mobile Network.
- M2M GW is connected to M2M Device through CAN.
- M2M GW has capability to manage status of on-board equipment in Smart Vehicle.

6.11.5 Triggers

- Trigger from Server:
  - When a new firmware becomes available, M2M Management Server notifies it to M2M GW and M2M GW notifies it to User via a display device of M2M GW.
- Trigger from User:
  - User can operate M2M GW to check availability of updated firmware for ECUs in M2M Management Server.

6.11.6 Normal Flow

A: Generation of updated firmware

1) Firmware Developer generates updated firmware for an ECU and submits it to OEM for approval.
2) OEM verifies the updated firmware and attaches OEM's Digital Signature to it as token of approval.
3) OEM uploads the updated firmware with the Digital Signature to M2M Management Server.

B: Authentication

M2M Management Server and M2M GW conduct mutual authentication. It is possible to avoid unintentional outflow of firmware by distributing it to the right M2M GW using the authentication mechanism. M2M GW can conduct firmware update operation based on request from the trusted Management Server. Widely used authentication procedures such as Challenge and Response using shared key cryptosystem and public key cryptosystem are applicable.

C: Establishment of Secure Channel

Encryption of communication channel between M2M Management Server and M2M GW is carried out after the mutual authentication. Encryption procedure such as SSL is applicable.

D: Verification of current version of Firmware installed in M2M Device

1) M2M Management Server sends a request to M2M GW in order to verify version information of firmware currently installed in M2M Device.
2) M2M GW transfers the verification request to M2M Device.
3) M2M Device sends version information of firmware currently installed back to M2M GW together with its own Digital Signature.
4) M2M GW transfers the version information with the Digital Signature of M2M Device to M2M Management Server.
5) Management Server verifies the version information with the Digital Signature and checks presence of firmware to be updated for the M2M Device.

E: Distribution of updated Firmware

1) M2M Management Server distributes the updated firmware with OEM's Digital Signature to M2M GW.
2) M2M GW carries out verification of the Digital Signature of the firmware.

F: User Authorization for Firmware Update

M2M GW sends a message to User in order to confirm permission from the User to conduct firmware update operation, with information regarding firmware update operation.

G: Firmware Update Implementation

1) M2M GW transfers the updated firmware to M2M Device.

2) M2M Device verifies Digital Signature of the firmware and gets updated with the firmware if it is the right firmware.

H: Notification of Firmware Update completion

1) M2M Device sends version information of the updated firmware with its Digital Signature when reboot with the updated firmware is completed.

2) M2M GW transfers the firmware version information with Digital Signature of M2M Device to M2M Management Server.

3) M2M Management Server verifies the Digital Signature of M2M Device and then the operation of Firmware Update is finally completed.

6.11.7 Alternative flow

(1) User Initiated Firmware Update Operation

- Steps A through C are the same as shown in Normal Flow.
- Step D: User Verification of Presence of Firmware to be updated:

    1) User operates Control Display Device of M2M GW in order to check availability of updated firmware for M2M Device.

    2) M2M GW sends a request to M2M Device in order to verify version information of firmware currently installed in M2M Device.

    3) M2M Device sends version information of firmware currently installed back to M2M GW together with its own Digital Signature.

    4) M2M GW transfers the version information with the Digital Signature of M2M Device to M2M Management Server.

    5) Management Server verifies the version information with the Digital Signature and checks presence of firmware to be updated for the M2M Device.

    6) If there is firmware to be updated, follow Steps E - H of Normal Flow.

    7) If there is no firmware to be updated, M2M GW notifies User of it on the display.

(2) Constrained Device (ECU)

This alternative flow is provided for a Constrained Device which is unable to conduct attachment of its own Digital Signature or verification of Digital Signature of other servers and devices by itself due to limitation of capability. In this case, it is possible to conduct Firmware Update operation by trusting M2M GW on the assumption that M2M GW and M2M Device are mutually trusted:

- Steps A through C are the same as shown in Normal Flow.
- Step D: Verification of current version of Firmware installed in M2M Device:

    1) M2M Management Server sends a request to M2M GW in order to verify version information of firmware currently installed in M2M Device.

    2) M2M GW transfers the verification request to M2M Device.

    3) M2M Device sends version information of firmware currently installed back to M2M GW.

    4) M2M GW transfers the version information with its Digital Signature to M2M Management Server.

    5) M2M Management Server verifies the Digital Signature and version information, and checks presence of firmware to be updated for the M2M Device.

- Step E: Distribution of updated Firmware (same as Step E in Normal Flow).

- Step F: User Authorization for Firmware Update (same as in Step F in Normal Flow).
- Step G: Firmware Update Implementation:
  1) M2M GW transfers updated firmware to M2M Device.
  2) M2M Device gets updated with the firmware trusting M2M GW.
- Step H: Notification of Firmware Update completion:
  1) M2M Device sends version information of the updated firmware when reboot with the updated firmware is completed.
  2) M2M GW transfers the firmware version information with its Digital Signature to M2M Management Server.
  3) M2M Management Server verifies the Digital Signature and then the operation of Firmware Update is finally completed.

6.11.8 Post-conditions

- M2M Management Server has to verify the status of M2M Device (firmware version, number of update failures, error information, etc.) and manage these types of information elements.
- M2M Device has to have rollback function in order to restore to the original state, when the Firmware Update operation fails.

6.11.9 High Level Illustration

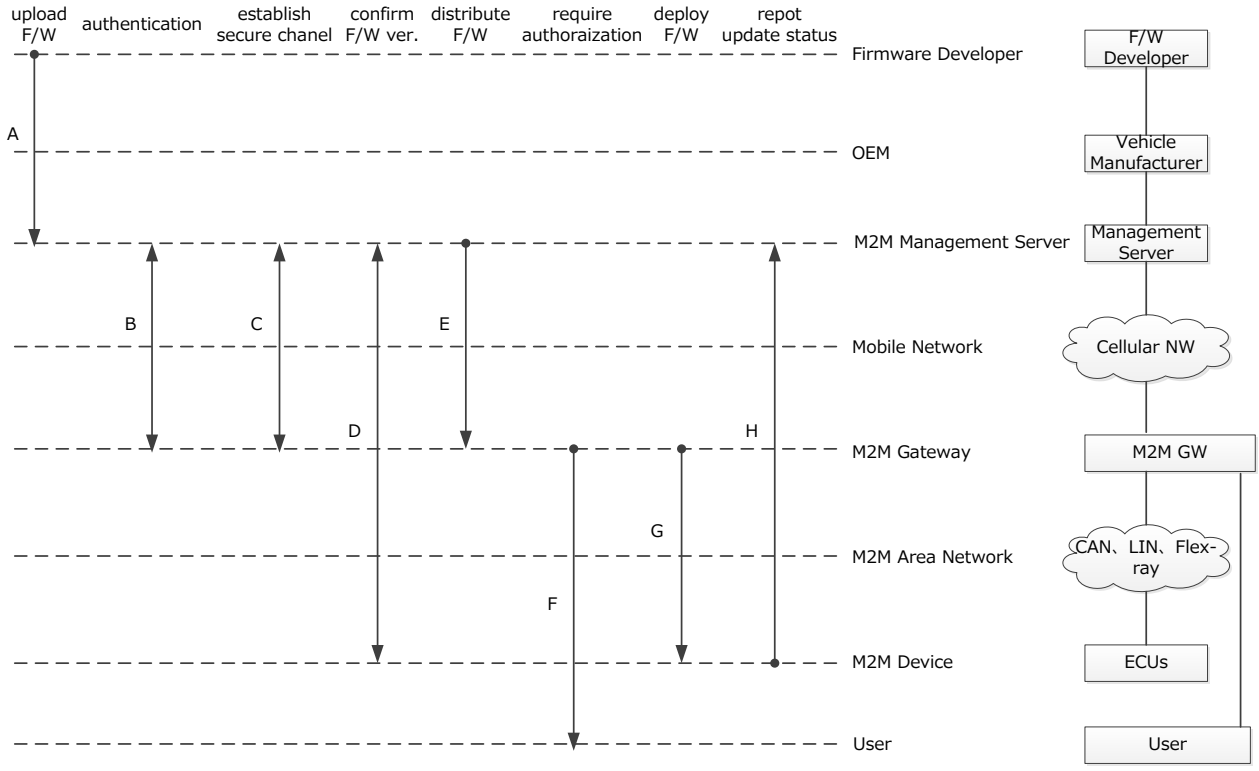

Figure 6.11.9-1 High Level Illustration - Secure Over-The-Air Firmware Update for Automotive ECUs

6.11.10 Potential requirements

1) The oneM2M System shall be able to prevent unauthorized modification of the firmware of M2M Device ([i.2] SER-064).

2) The oneM2M System shall be able to detect unauthorized modification of the firmware of M2M Device ([i.2] SER-065).

3) The oneM2M System shall be able to stop operation of M2M device when it is updated with wrong firmware.

4) The oneM2M System shall be able to support security mechanisms to protect their cryptographic keys and cryptographic operations by using tamper resistant elements such as TPM (Trusted Platform Module), HSM (Hardware Security Module) and SIM (Subscriber Identity Module) ([i.2] SER-066).

5) The oneM2M System shall be able to prevent malfunction of M2M Device caused by receiving unsolicited messages or information ([i.2] SER-067).

6.12 Car/Bicycle Sharing Services

6.12.1 Description

As seen clearly, automation already penetrates all aspects of life even in our urban life. The goal of this use case is to describe several automation services which are occurred in different urban space in different life style, e.g. bicycle/car sharing services.

Brief Features of Services.

Car Sharing Service

Car Sharing is to offer a new service model for automobile transportation. Simply, Car Sharing is a self-service, on-demand alternative to car ownership; a service that is offered to urban residents (B2C) and businesses (B2B).

This service is mainly designed around a particular user profile - first of all, people who live in cities but do not drive a car every day and secondly tourists who visit cities but do not bring their car. Thus, people who need a car at short notice but take an alternative to car ownership.

The brief procedure of this service is:

1) joining the membership;

2) unlocking the car door;

3) driving away;

4) parking to any reserved spot provided by the service provider and/or public; and

5) paying as you drive (including gas, insurance, etc.).

Bicycle Sharing Service

Bicycle sharing service is also a new service in which bicycle are made available for shared use to individuals who do not own a bicycle. Generally, bicycle sharing service is organized by a local government agency but may be operated privately.

The procedure of this service is similar to the car sharing service, but different type of services such as healthcare service can be combined.

6.12.2 Source

oneM2M-REQ-2012-0132R01 Use Case: Car/Bicycle Sharing Services.

6.12.3 Actors

User

A user who makes use of the shared things which are cars or bicycles.

Sensors (or Sensor Devices)

Sensor Devices can vary based on usage, and do not have any direct communication interfaces to the M2M Service Platform:

- For Car Sharing Service - Door Control Sensor, Tire Pressure Sensor, Fuel Indication Sensor, GPS.
- For Bicycle Sharing Service - Lock Control Sensor, Accelerometer, Tire Pressure Sensor, Heart-rate Sensor.

Smartphone

A device which is an intermediate entity and is available to connect sensors to a M2M Service Platform. The basic role is similar to the general M2M gateway, but it has some sensors and some applications (navigation) itself that may be used by services.

M2M Service Platform

In charge of providing common functionalities for the M2M services. It is mainly in charge of collecting the status and configuration information of sensors and controlling them via the smartphone and/or M2M gateway.

M2M Service Providers

Companies which provide their own M2M services for the user through the M2M Service Platform. The M2M Service Providers can be multiple according to the types of services.

The providers include Car Sharing Service Provider, Insurance Company, Gas Station, Bicycle Sharing Service Provider, and Healthcare Service Provider.

6.12.4 Pre-conditions

See sub-case flows.

6.12.5 Triggers

See sub-case flows.

6.12.6 Normal Flow

Sub use case 1 - Car Sharing Case

Trigger

A user wants to make use of the car.

Pre-conditions

The user preliminary joins a membership of the Car Sharing Service.

Sensors built in the car are required to periodically (normal) and non-periodically (urgent) send sensor data to the M2M Service Platform based on the trigger defined by the M2M Service Providers.

The M2M Service Platform collects and manages data and configurations related to the services. Generally, each service has its own data and configuration set, simply called resources.

The M2M Service Providers in the service domain have a service agreement between each other for unified services.

The Smartphone has a navigation and car sharing application.

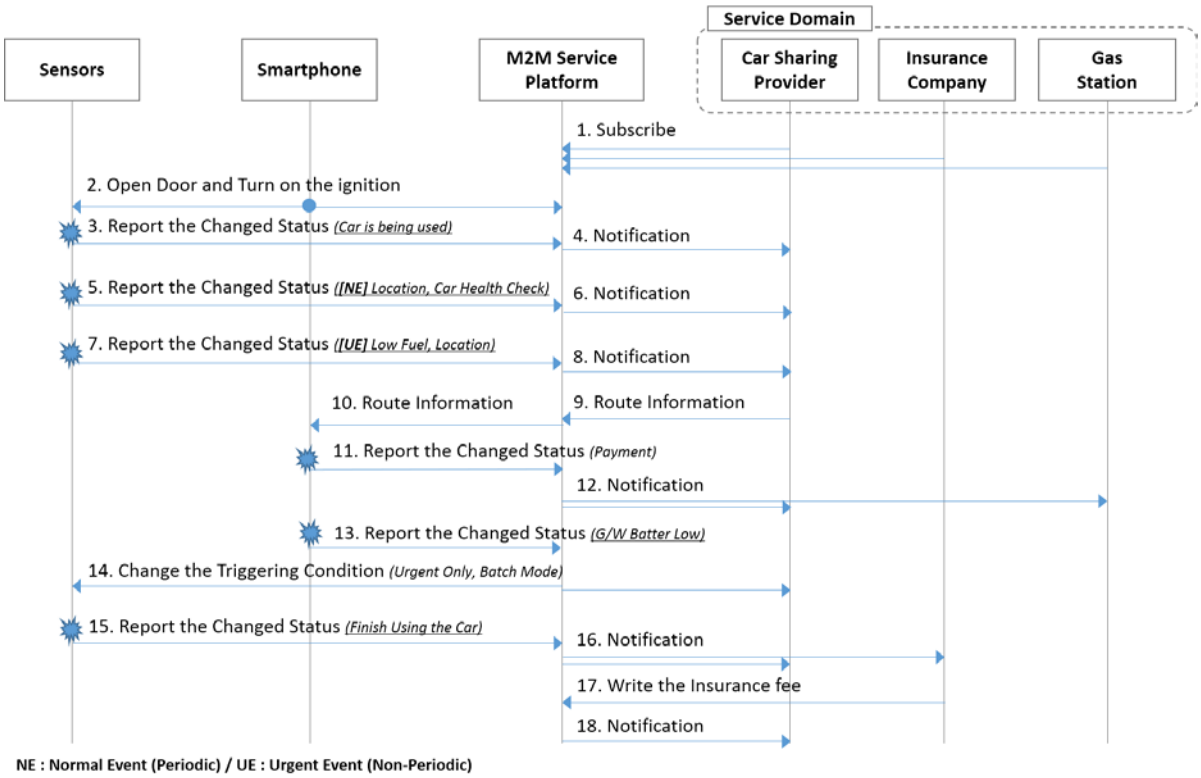

Figure 6.12.6-1: Car Sharing Normal Flow

Detailed Flow Descriptions

1) The Applications of each Service Provider in the service domain register and subscribe to changes of resources (or information) about the Car Sharing Service in the M2M Service Platform.

2) Since each resource in the M2M Service Platform is owned by the Car Sharing Provider, Insurance Company or Gas Station, if an application needs to access another resource, it shall request proper access right for the resources and that request will be granted if appropriately based on the service agreement.

3) As the user finds a shared car, he/she opens the car door and turns on the ignition using interfaces of the Smartphone such as Bluetooth and NFC, if the user is authorized.

4) The Sensors report the changed status to the M2M Service Platform via the Smartphone as a gateway when the specific condition is triggered (car is just being used).

5) The M2M Service Platform notifies the Car Sharing Service Provider of the changed status.

NOTE 1: The Car Sharing Service Provider can update the situation that the car is being used on its website.

6) (Normal Reporting Case for managing the Service) The Sensors report the changed status to the M2M Service Platform via the Smartphone when the specific condition is triggered. (Periodic location reporting and car health check for maintenance reasons)

7) The M2M Service Platform notifies the Car Sharing Service Provider of the changed status.

NOTE 2: Agreement on privacy policy of location is preliminarily confirmed.

8) (Urgent Reporting Case for handling any emergency) The Sensors report the changed status to the M2M Service Platform via the smartphone as a gateway when the specific condition is triggered (the fuel is low).

9) The M2M Service Platform immediately notifies the Car Sharing Service Provider of the changed status.

10) The Car Sharing Service Provider finds out the nearest Gas Station according to the received location information and service agreements between the Car Sharing Service Provider and Gas Stations, and the Provider sends the route information to M2M Service Platform.

11) The M2M Service Platform notifies the Smartphone of the route information.

12) After filling the fuel, the user virtually pays the fuel fee by using the Smartphone's NFC tag. The payment information is reported to the M2M Service Platform.

13) The M2M Service Platform notifies the Car Sharing Provider and the Gas Station of the payment information.

NOTE 3: This procedure is for the Car Sharing Provider to pay to Gas Station the fuel fee instead of the user.

14) Afterwards, due to the low battery of the Smartphone (less than 30 % remain), the Smartphone reports the changed status to the M2M Service Platform.

15) The M2M Service Platform automatically changes the subscription and reporting attributes of the Sensors and the Car Sharing Service Provider.

EXAMPLE 1: If the Platform changes the subscription attributes to "only emergency case", only emergency subscription case will be notified. The others cannot be notified, but at the end of service, will be sent in batch-mode.

16) As the user arrives at the destination, and turns off the ignition, the sensors report the accumulated information and normal event subscription information, to the M2M Service Platform via smartphone.

17) The M2M Service Platform notifies the Car Sharing Providers and Insurance Company of the usage of the shared car.

18) The Insurance Company claims the insurance fee by writing onto the Car Sharing Service Provider's resource in the M2M Service Platform, in this case the Insurance Company preliminarily acquires proper access right to write the resource.

19) The M2M Service Platform notifies the Car Sharing Provides of the insurance fee.

Post-conditions

- The User will pay as he/she drive according to the recorded data.
- The Car Sharing Service Provider can update the position and status of the car on its website using the recorded data. Thus, next users can make use of the Car Sharing Service.

Sub Use Case 2 - Bicycle Sharing Service

Trigger

A user wants to make use of the bicycle.

Pre-conditions

The user preliminary joins a membership of the Bicycle Sharing Service.

The sensors built in the car and in the smartphone are required to periodically (normal) and non-periodically (urgent) send sensor data to the M2M Service Platform based on the trigger defined by the Service Provider.

The M2M Service Platform collects and manages data and configurations related to the services. Generally, each service has its own data and configuration set, simply called resources.

The Smartphone has a navigation and bicycle sharing application.

The M2M Service Providers in the service domain have a service agreement between each other for unified services.

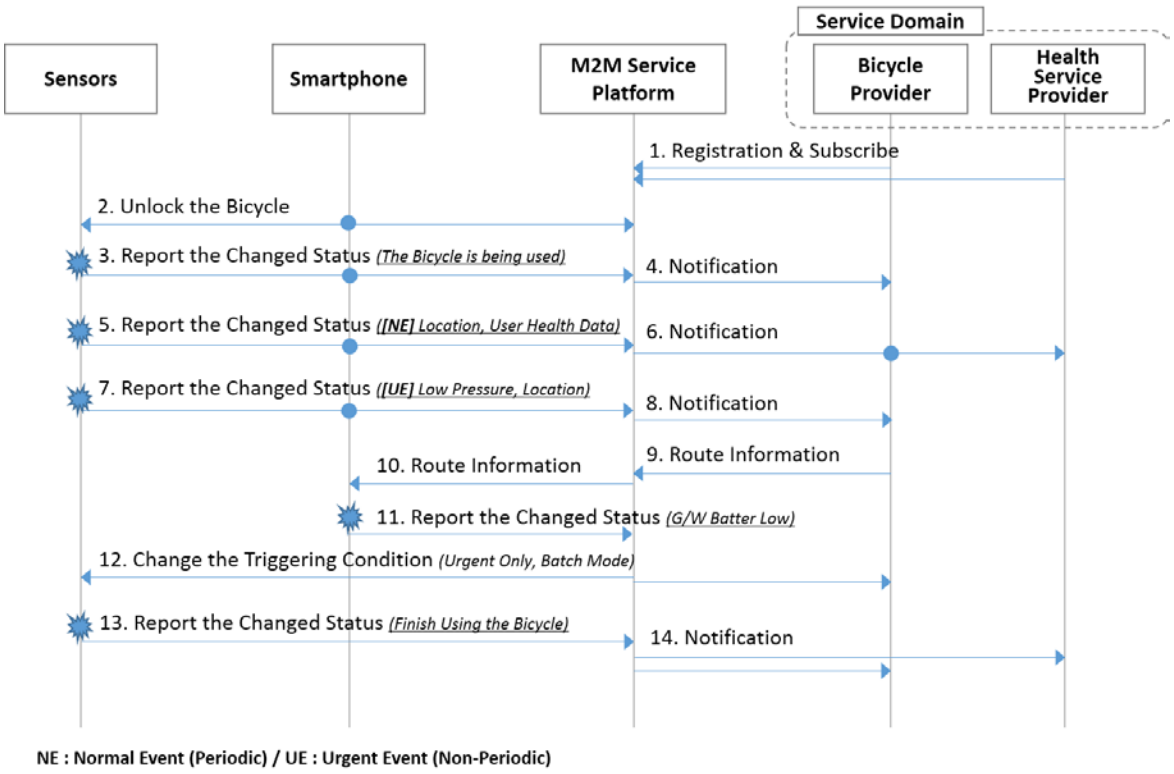

Figure 6.12.6-2: Bicycle Sharing Normal Flow

Detailed Flow Descriptions

1) The Applications in the service domain register the service and subscribe to changes of information about the Bicycle Sharing Service.

2) Since each resource in the M2M Service Platform is owned by the Bicycle Sharing Service Provider or Health Service Provider, if an Application needs to access another resource, it shall request proper access right to the resources and that request will be granted if appropriately based on the service agreement.

3) To unlock the bicycle, the user tags the locker of the bicycle through the NFC interface.

4) The Sensors report the changed status to the M2M Service Platform via the smartphone as a gateway when the specific condition is triggered.

EXAMPLE 2: The bicycle is being used.

5) The M2M Service Platform notifies the Bike Sharing Service Provider of the changed status.

NOTE 4: The Bicycle Sharing Service Provider can record the situation on its web-site that the car is being used.

6) (Normal Reporting Case for managing the Service) The heart-rate of the user is continuously collected by the heart-rate sensor on the handlebar, and the health-related information such as heart-rate, location, time is reported periodically to the Service Operator.

7) The M2M Service Platform notifies the Bicycle Sharing Service Provider and the Health Service Provider of the health Service information.

8) (Urgent Reporting Case for handling any emergency) While riding the bicycle, the tire pressure sensor detects the low pressure of the front tire, the information is immediately sent to the M2M Service Platform via the Smartphone with location information.

9) The M2M Service Platform notifies the Bicycle Sharing Service Provider of the changed status.

10) The Bicycle Sharing Service Provider finds out the nearest bike repair shop according to the received location information, and the Provider sends the route information to M2M Service Platform.

11) The M2M Service Platform forwards the route information to the Smartphone which has a navigation application.

12) Afterwards, due to the low battery of the Smartphone (less than 30 % remain), the Smartphone reports the changed status to the Service Operator. (Low battery indication)

13) The M2M Service Platform automatically changes the subscription attributes of sensors and the Bicycle Service Provider such as delay tolerance to reduce battery-consumption.

EXAMPLE 3: If the Platform changes the subscription attributes to "only emergency case", only emergency subscription case will be notified. The others cannot be notified, but at the end of service, will be sent in batch-mode.

14) As the user arrives at the destination and parks to the reserved spot, the Sensor reports the accumulated information and normal event subscription information to the M2M Service Platform via the Smartphone.

15) The M2M Service Platform notifies Bicycle Sharing Service Provide and Healthcare Service Provider of the usage of the shared bicycle.

Post-conditions

- The User may pay as he/she drive according to the recorded data (not for the public service case).
- The Bicycle Sharing Service Provider can update the position and status of the bicycle on its website using the recorded data. Thus, next users can make use of the Bicycle Sharing Service.

6.12.7 Alternative Flow

None.

6.12.8 Post-conditions

See sub-case flows.

6.12.9 High Level Illustration

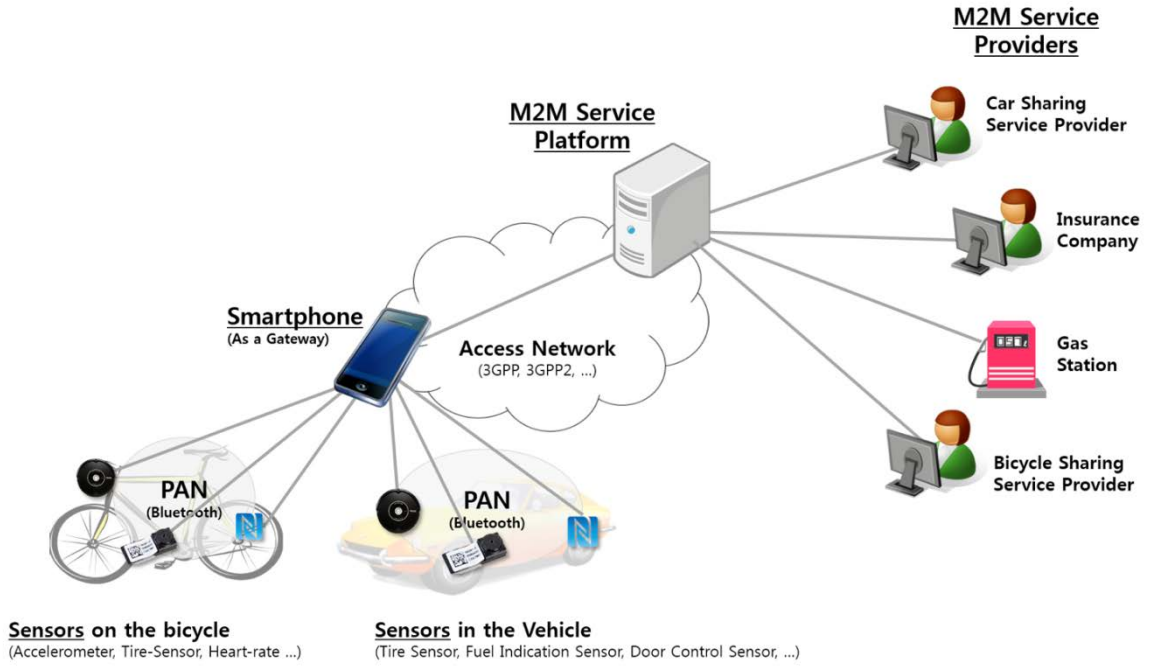

Figure 6.12.9-1: Car/Bicycle Sharing High Level Illustration

6.12.10 Potential Requirements

1) The M2M System shall support integration of mobile/portable user owned equipment (e.g. smartphone) as M2M Gateway and/or Device.

2) The M2M System shall support distinguishing multiple event levels for reporting and handle them differently ([i.2] OSR-032).

NOTE: For example, the event levels may be divided into normal and urgent events.

3) Based on the condition of the M2M Gateway and/or Device, the M2M System may change the reporting (or subscription) mechanisms and/or configurations related to a service ([i.2] OSR-033).

4) The M2M System shall support processing of access right requests to a resource and grant the requests if required conditions are met.

6.13 Smart Parking

6.13.1 Description

Smart parking helps address one of the biggest problems on driving in urban areas; finding empty parking spaces and controlling illegal parking. Parking spaces are wide spread and owned by different providers so that it is not easy to access at one place/time.

With smart parking service, drivers can easily find available parking spaces, pay parking fees and even can make reservations. Making parking reservations would be available for limited people such as VIPs or the disabled, since ordinary parking service needs to satisfy first-come-first-served rule.

In this use case, law enforcement authority is also included as an actor. This implies M2M technologies supports rightness/safety as well as convenience.

6.13.2 Source

oneM2M-REQ-2013-0169R03 Use Case Smart Parking.

6.13.3 Actors

M2M Service Platform

This is a platform that interacts with M2M Gateways/Devices and M2M Application Service Providers.

Smartphone

This is a M2M Device acts as a car navigator and a wallet to pay parking fee by connecting parking meters.

On-street Parking Meter

This is a M2M Device installed near parking slots to charge drivers parking fees.

In-building Parking Sensor

This is a M2M Device with a small camera that can recognize a plate on cars, and is installed near disabled-only parking spaces.

Parking Provider

This is a M2M Application Service Provider who owns parking lots, in this use case there are two parking providers; in the mall and on street.

Billing Provider

This is a M2M Application Service Provider (e.g. financial institution) who provides billing service for M2M Users, e.g. for parking fees. When bills are issued by M2M Application Service Providers, coupons may be used for compensation schemes. This can also apply for fines issued by police centers.

Police Center

This is a law enforcement authority, one of M2M Application Service Providers, who charges fine to whom break laws.

User

This is a M2M service user who drives a car. In the second sub case, dedicated parking space, there are two users. One originally makes a reservation who is handicapped, and the other who illegally parks a car on disabled-only parking area.

6.13.4 Pre-conditions

See sub-case below.

6.13.5 Triggers

See sub-case below.

6.13.6 Normal Flow

(1) Finding Space, Parking Car & Paying Bill

Pre-condition

User sets a destination such as a shopping mall using the smartphone navigator, and also checks availability of parking space in the building before or while driving.

Triggers

The car approaches near the destination.

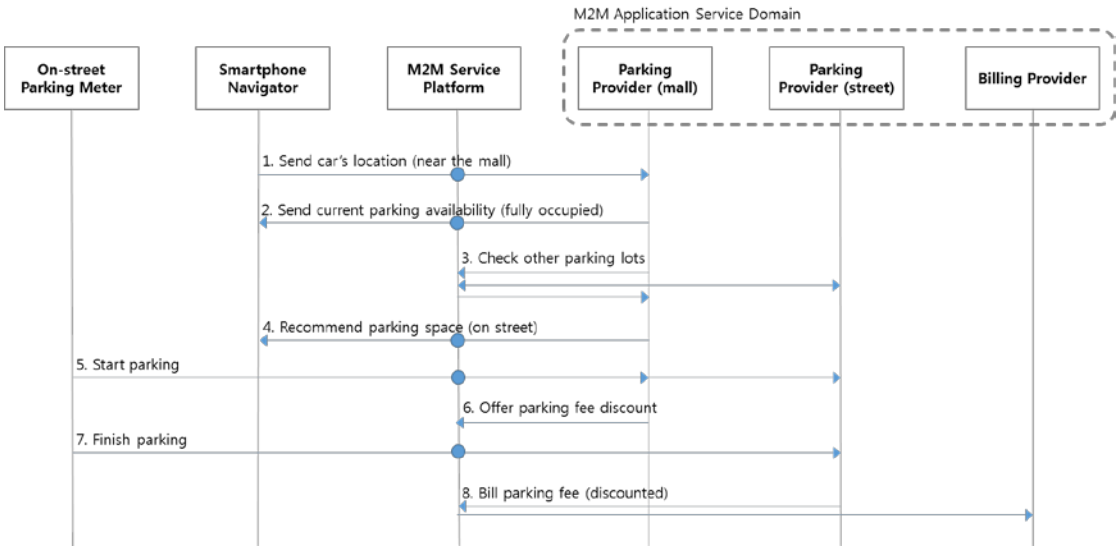

Figure 6.13.6-1: Normal Flow - Finding Space, Parking Car & Paying Bill

Normal Flow

1) Since the user set the mall as the destination before, when the user is near the mall, the navigator sends the location to the mall parking provider automatically.

2) The mall parking provider informs the navigator that there's no empty parking space now.

3) Based on the car's location, which is near the mall, the mall parking provider inquires availability of other parking spaces through M2M service platform.

4) There are empty spaces on street, so the mall parking provider recommends that parking space.

5) The user approaches the smartphone from a parking meter to start parking. Then the street parking provider is notified, as well as the mall parking provider.

6) The mall parking provider offers discount coupon for parking outside as compensation.

7) The user touches the smartphone on the meter to finish parking.

8) The street parking provider bills parking fee. The bill with discount coupon is sent to billing provider through M2M service platform.

(2) Dedicated Parking Space

Pre-condition

Before driving, the user (user A) makes a parking reservation for a slot in a shopping mall, which is especially for the disabled. It is normally assured because there will be fines for illegal parking on this dedicated parking space.

Triggers

None.

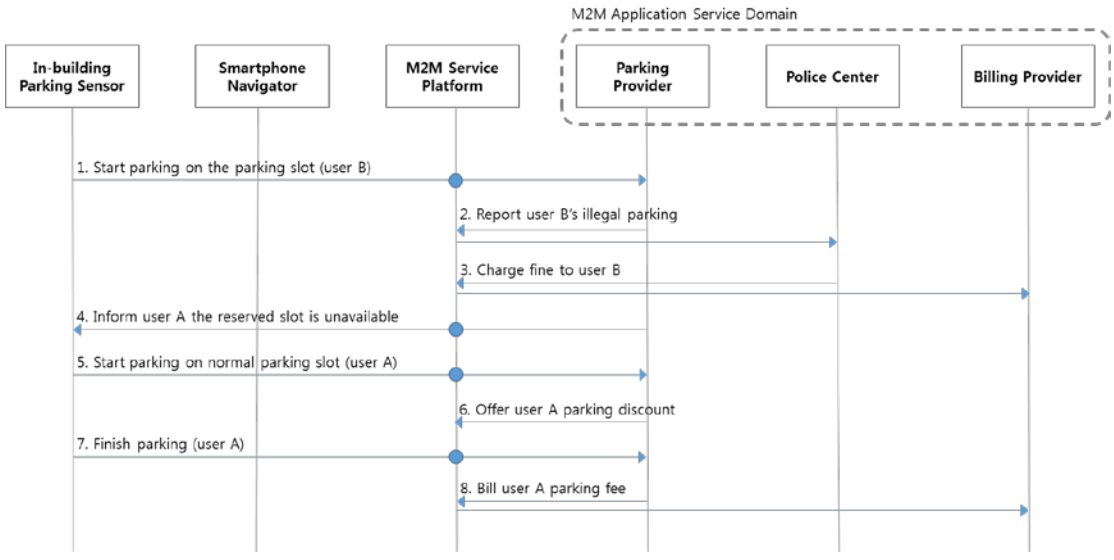

Figure 6.13.6-2: Normal Flow 1 - Finding Dedicated Parking Space

Normal Flow

1) The other user (user B) parks a car on the parking lot, which is already reserved by user A.

2) User B's illegal parking on the disabled-only parking area is reported to police center.

3) Police center charges fine on user B.

4) User A approaches the mall and notices that reserved parking space is taken and that only choice now is normal parking slots.

5) User A parks on a normal parking slot instead of the reserved one.

6) The parking provider offers parking discount coupon to the user A as a compensation.

7) After shopping, user A leaves the building and finish parking.

8) The parking provider bills parking fee for user A, applying the parking coupon.

6.13.7 Alternative Flow

Alternative Flow 1 - Dedicated Parking Space

Pre-condition

Before driving, the user (user A) makes a parking reservation for a slot in a shopping mall, which is especially for the disabled. It is normally assured because there will be fines for illegal parking on this dedicated parking space.

Triggers

None.

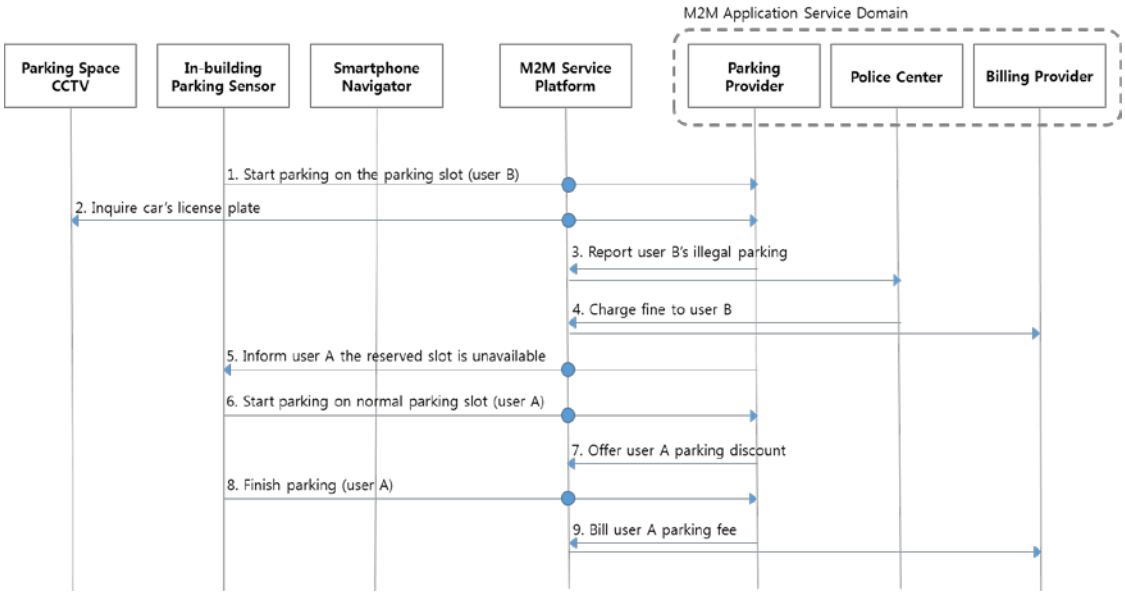

Figure 6.13.7-1 Alternative Flow 1 - Finding Dedicated Parking Space

1) The other user (user B) parks a car on the parking lot, which is already reserved by user A.

2) The mall parking provider inquires plate number of the car to a CCTV near the parking space.

3) User B's illegal parking on the disabled-only parking area is reported to police center.

4) Police center charges fine on user B.

5) User A approaches the mall and notices that reserved parking space is taken and that only choice now is normal parking slots.

6) User A parks on a normal parking slot instead of the reserved one.

7) The parking provider offers parking discount coupon to the user A as a compensation.

8) After shopping, user A leaves the building and finish parking.

9) The parking provider bills parking fee for user A, applying the parking coupon.

6.13.8 Post-conditions

None.

6.13.9 High Level Illustration

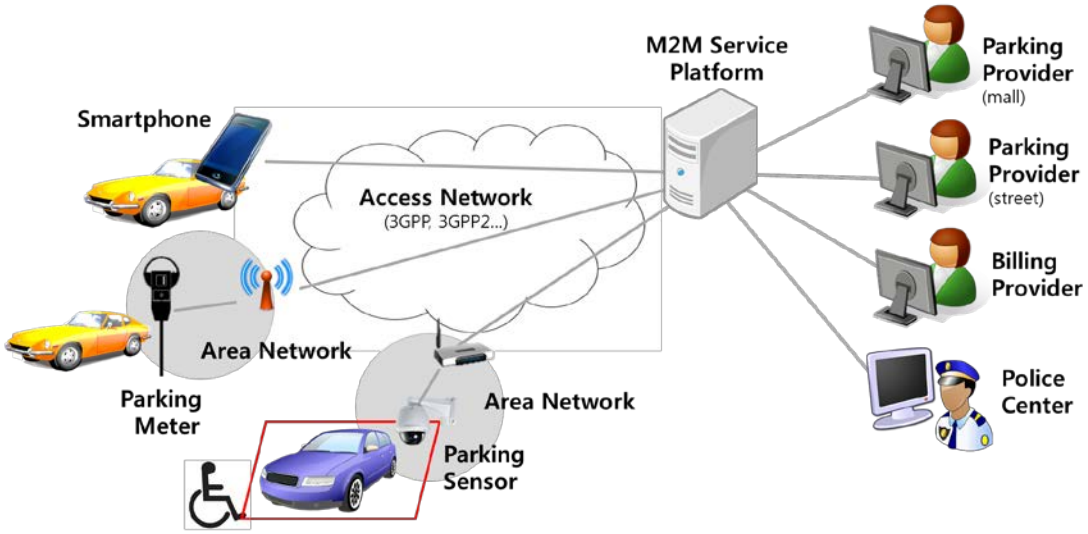

Figure 6.13.9-1: High Level Illustration of Smart Parking

6.13.10 Potential Requirements

1) The M2M System shall support mechanisms to correlate or compensate charging data/records from different M2M Application Service Providers.

2) The M2M System shall support triggering of M2M Devices to report collected data on-demand.

6.14 Vehicle Broadcasting without Registration

6.14.1 Description

This use case consists for any vehicle driving fast to signal its emergency state which can be for example hard braking state or abnormal vehicle state to other following vehicles or infrastructure, such as a road side unit in its vicinity.

By broadcasting time limited periodic messages which indicate vehicle's emergency state triggered by an event, the vehicle can warn unspecified entities of a sudden danger so limiting the risk of collision. The infrastructure can collect those information and relay them to other entities or servers in the distance.

Since a vehicle is a very quick-moving object and the transferred message is a time-critical information, the transmission speed is crucial for this use case.

6.14.2 Source

REQ-2016-0031R03 Vehicle Broadcasting without Registration TR-0026.

6.14.3 Actors

Source Vehicle

A Source Vehicle is a moving object which provides communications functions necessary to support connected vehicle operations. It can detect its emergency state and send warning message by broadcasting.

Target Vehicle

A Target Vehicle is a moving object which provides communications functions necessary to support connected vehicle operations for communicating with the Source Vehicle. After receiving warning message, it tries to avoid the potential danger.

Infrastructure Node

An Infrastructure Node is located by a road and provides communications functions necessary to support connected vehicle operations. It can collect information generated in a certain area and partially process it. It is also possible to convey it to the server or other vehicles in a far distance.

6.14.4 Pre-conditions

N/A.

6.14.5 Triggers

When a vehicle application detects its emergency state, it should trigger broadcasting specific information.

6.14.6 Normal Flow

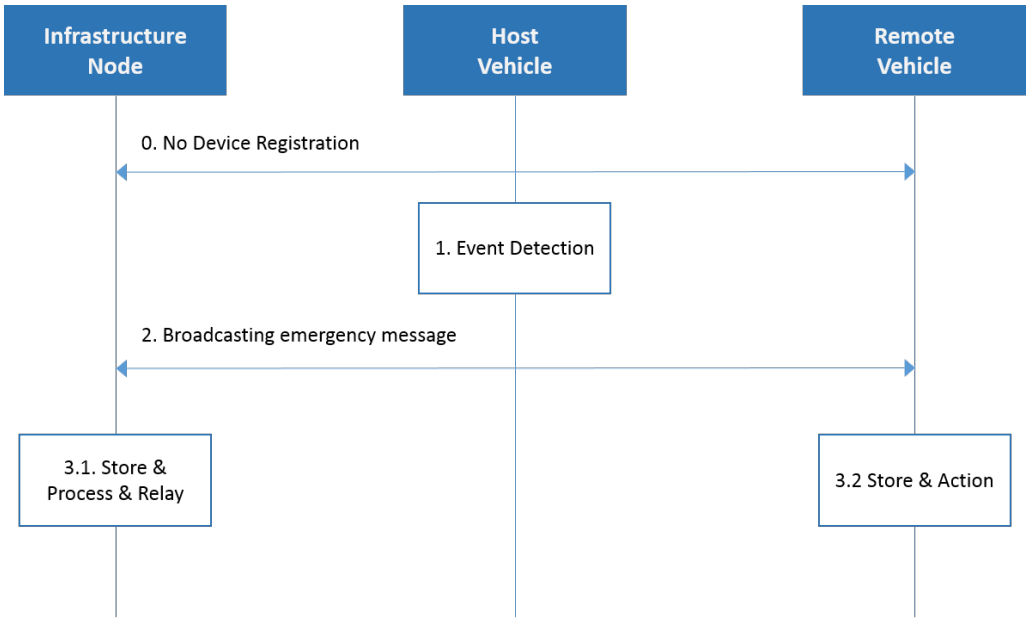

Figure 6.14.6-1 Normal Flow - Vehicle Broadcasting without Registration

0) Each Vehicle and Infrastructure Node do not register with each other.

1) Source Vehicle runs event detection engine and analyses whether the safety-related event has occurred or not.

2) Source Vehicle send the safety-related information to unspecified entities by broadcasting.

3.1) Infrastructure Node gathers the broadcasted information and locally processes and relays to other vehicles or servers in a distance if needed.

3.2) Target Vehicle receives the safety-related information and re-acts to avoid the potential collision.

6.14.7 Alternative Flow

N/A.

6.14.8 Post-conditions

N/A.

6.14.9 High Level Illustration

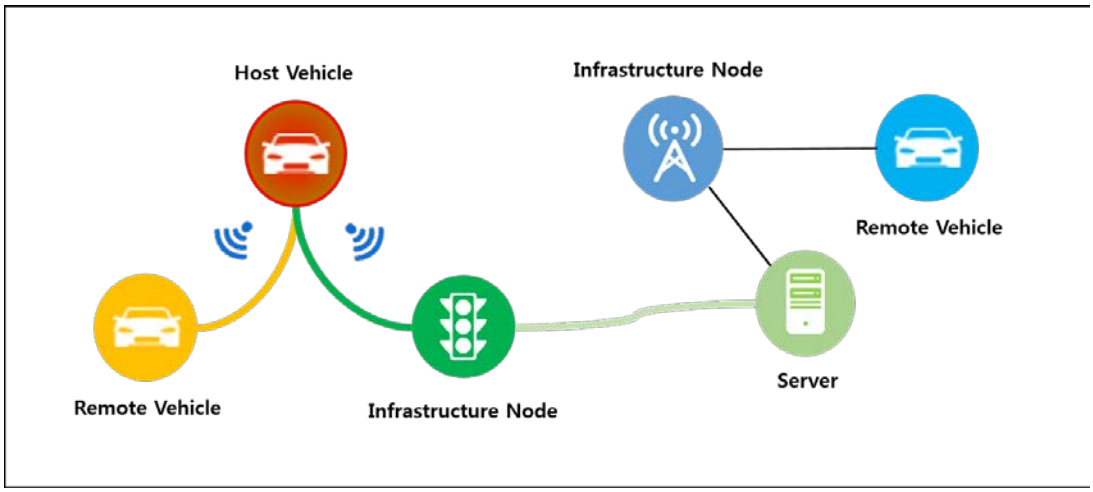

Figure 6.14.9-1 High Level Illustration - Vehicle Broadcasting without Registration

- Source Vehicle can send time-critical information to unspecified entities.
- Infrastructure Node can relay this information to the server which can also re-use that information.

6.14.10 Potential Requirements

The oneM2M system shall enable the M2M Infrastructure to facilitate direct communication between two or more different M2M devices without having registered with one another.

6.15 Vehicle location privacy protection

6.15.1 Description

The privacy protection is an issue that is considered for locating in human to human communications. For example, in 3GPP network, privacy protection is implemented by an independent entity such as a privacy profile register (PPR) or a gateway mobile location center (GMLC). In the vehicle domain, the location information of the vehicle represents the location of the vehicle user. Therefore an issue of privacy protection is also closely related to the vehicle location information. The privacy protection refers to that when a vehicle is connected to the M2M system, the user of the vehicle has permission to specify when and where a third party application is allowed to access to the vehicle location information.

After the M2M platform receives a message from a third party application, requesting for the location information of a certain vehicle, the M2M platform determines the entity that should performs the privacy inspection, according to the source of the location information. For example, if the location information of the vehicle is network based and the underlying network is 3GPP, then the entity that performs the privacy inspection should be the 3GPP location server. On the other hand, if the location information of the vehicle is device based, then it should be the M2M platform that performs the privacy inspection.

6.15.2 Source

REQ-2016-0040R02-TR-0026-Vehicle_location_privacy_protection.

6.15.3 Actors

- M2M Device: It is embedded in a vehicle, which is used to connect to the M2M platform.
- Vehicle: Vehicle connects to the M2M platform via the embedded M2M device and it is equipped with hardware devices for localization. (e.g. GPS, Cellular modem).
- Vehicle Owner: The owner of the vehicle. The owner has privacy agreement with the 3GPP network location server or the M2M platform.
- 3GPP network location server: When the vehicle connects to the M2M platform via the 3GPP underlying network, the 3GPP network location server can perform the location service and provide the location information to the M2M platform. In this case the 3GPP network location server also performs the privacy inspection function. The 3GPP network location server is configured with location privacy policies according to the privacy agreement with the vehicle owner and the network operator policy. The location privacy policies refer to when and where a third part application is allowed to access to the vehicle location information.
- M2M Platform: It connects, manages M2M devices and exposes location services to the M2M application. The M2M platform can collect the vehicle location information either from the underlying network, e.g. 3GPP network location server, or from the device itself, e.g. the vehicle is equipped with any location capable modules or technologies (e.g. GPS) and is able to position itself. The M2M platform is configured with location privacy policies according to the privacy agreement with the vehicle owner. The location privacy policies refer to when and where a M2M application is allowed to access to the vehicle location information.
- M2M Application server: The M2M application server requests for the location information of the vehicle for any possible application usage, e.g. fleet management or traffic monitoring.

6.15.4 Pre-conditions

- The vehicle owner already has privacy protection agreement with the operator of the 3GPP network or the M2M platform.

6.15.5 Triggers

The M2M application server requests to access to the location information of the vehicle.

6.15.6 Normal Flow

Scenario 1: The vehicle No.1 connects to the M2M platform via the 3GPP network and the location information is collected by the 3GPP network location server:

1) The M2M application sends message to the M2M platform to request for the location information of vehicle No.1.

2) The M2M platform checks the location information source of the vehicle. If the location information of the vehicle comes from the underlying network location server, e.g. 3GPP network location server, the M2M platform sends privacy inspection requests, including the M2M application identifier which can be recognized by the 3GPP network, to 3GPP network location server to verify whether the M2M application is allowed to access to the location information.

3) The 3GPP network location server is configured with the privacy policies of vehicle No.1. When the 3GPP network location server receives the access right verify request, it performs the privacy inspection, checking the privacy policies with the M2M application identifier and other information, e.g. the time and the lasted location of the vehicle. The 3GPP network location server returns the checking result to the M2M platform, that is whether the M2M application is allowed to access to the location information.

4) If the M2M application is allowed to access to the location information, after the M2M platform receives the responses from 3GPP network location server, it sends the location information request message to the 3GPP network location server.

5) The 3GPP network location server performs the location updating procedure inside the 3GPP network and collects the latest location information of the vehicle. Then the 3GPP network location server returns the location information to the M2M platform.

6) The M2M platform forwards the location information to the M2M application.

Scenario 2: The vehicle No.2 connects to the M2M platform via the 3GPP network. It is equipped with location capable modules or technologies (e.g. GPS) and is able to position itself. The M2M platform can collect the location information of the vehicle directly without the help of any other location server.

1) The M2M application sends message to the M2M platform to request for the location information of vehicle No.2.

2) The M2M platform checks the location information source of the vehicle. If the location information of the vehicle is device based and is collected by the M2M platform directly from the vehicle itself, then the M2M platform checks the privacy policies configured in the M2M platform.

3) If the M2M application is allowed to access to the location information, then the M2M platform returns the location information to the M2M application. If the M2M application is not allowed to access to the location information, then the M2M platform sends failure response to the M2M application with appropriate error code.

6.15.7 Alternative flow

N/A.

6.15.8 Post-conditions

N/A.

6.15.9 High Level Illustration

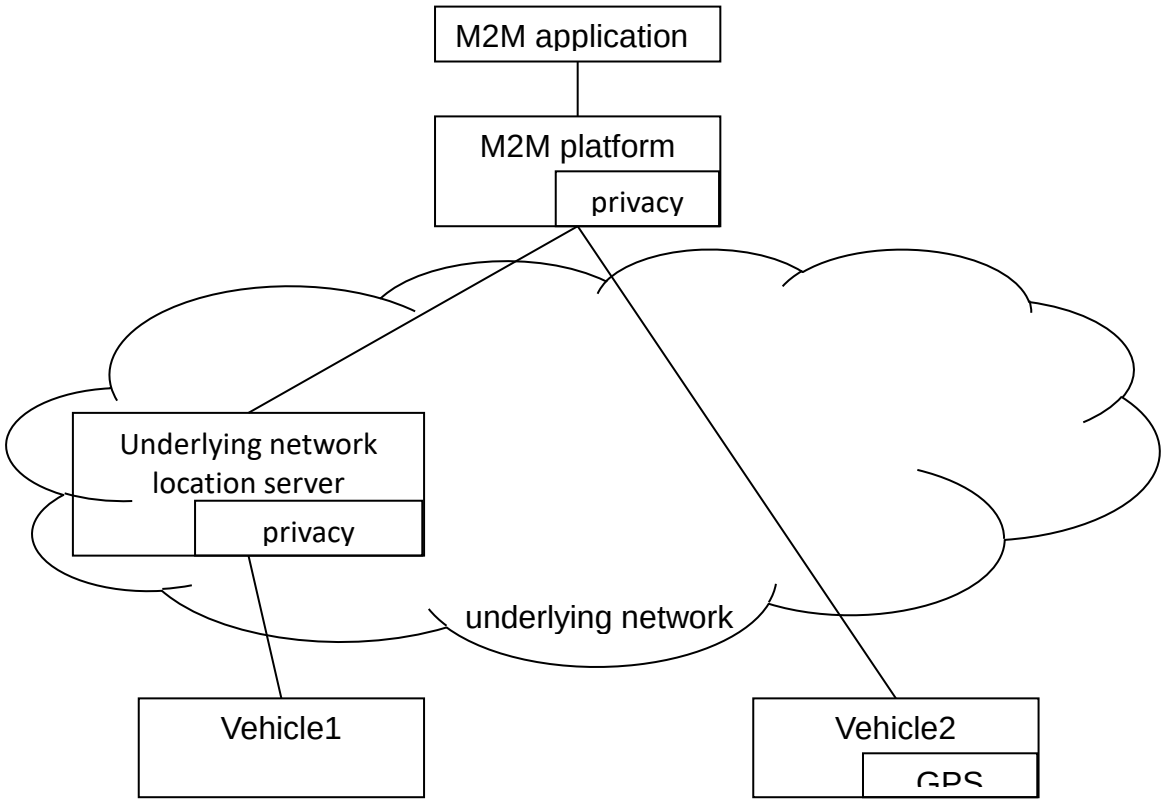

Figure 6.15.9-1 High Level Illustration - Vehicle Broadcasting without Registration

6.15.10 Potential requirements

1) The oneM2M system shall be able to support the enforcement and management (e.g. update) of the privacy policies for vehicle location information.

2) The oneM2M system shall be able to apply the privacy policies configured in the oneM2M system to those vehicles whose location information is obtained by the oneM2M system itself.

3) The oneM2M system shall be able to request the underlying network to perform, on behalf of the oneM2M system, the privacy policy decision concerning the location information if the location information is obtained by the underlying network location service ([i.2] SER-062 and SER-063).

6.16 Vehicle Domain service continuity

6.16.1 Description

Autonomous or self-driving cars have been gaining attention as early versions of such vehicles have become available. A gateway on such a vehicle, may be responsible for movement control (braking, turning, etc.) as well as delivering external services. Such a scenario requires real time monitoring of sensors and fast access to various actuators for accident avoidance. For example, time series data may be gathered which triggers specific actions when data is lost. In such cases service continuity and data transfer of time-critical information are essential.

6.16.2 Source

REQ-2016-0052R02-TR-0026_service_use_case.

6.16.3 Actors

Vehicle

A Source Vehicle is a moving object which provides support for services necessary for vehicle operations.

Road-Side Unit (RSU)

A Road-side unit is located along vehicular paths and provides localized support for services necessary for multiple vehicle operations. This support is provided for vehicles within the RSU's communication range.

Infrastructure Node

An Infrastructure Node provides centralized support for services provided by a Services Provider to a large number and variety of devices.

6.16.4 Pre-conditions

All RSUs are registered to the Infrastructure Node.

6.16.5 Triggers

When a vehicle moves it loses connectivity to one road-side unit, and then re-connects to different road-side units along the way.

6.16.6 Normal Flow

The Normal Flow depicts the case where a moving vehicle receives notifications.

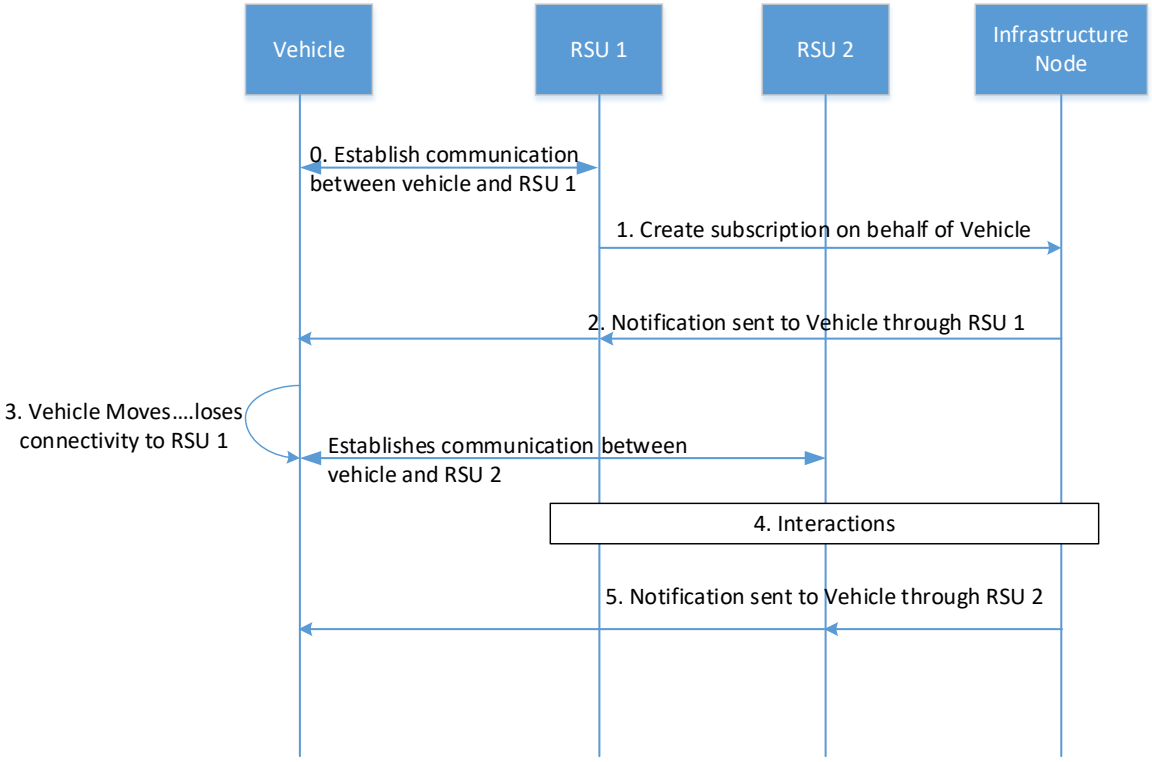

Figure 6.16.6-1: Normal Flow for Vehicular Domain Service Continuity

0) Vehicle establishes communication with RSU 1.

1) RSU 1 creates a subscription to a resource hosted on the Infrastructure Node on behalf of the Vehicle.

2) A notification corresponding to the subscription in step 1 is generated and sent to the Vehicle via RSU 1.

3) The vehicle moves, and establishes communication with RSU 2.

4) Supporting interactions between RSUs and Infrastructure Node.

5) A notification corresponding to the subscription in step 1 is generated. The notification is sent to the Vehicle via RSU 2.

6.16.7 Alternative flow

The Alternative Flow depicts the case where a moving vehicle reports observed road/traffic conditions (e.g. vehicles in proximity and their speed, distance to curb, etc.) and receives collision avoidance commands. The RSUs collect and store the information from the vehicles, and also exchange information with each other. This allows the RSUs to better identify the trajectory of vehicles and to help predict potential collisions. As collision avoidance decisions are extremely time sensitive, all decisions are made locally at the RSU. This avoids any unnecessary delays incurred with communication to the infrastructure node.

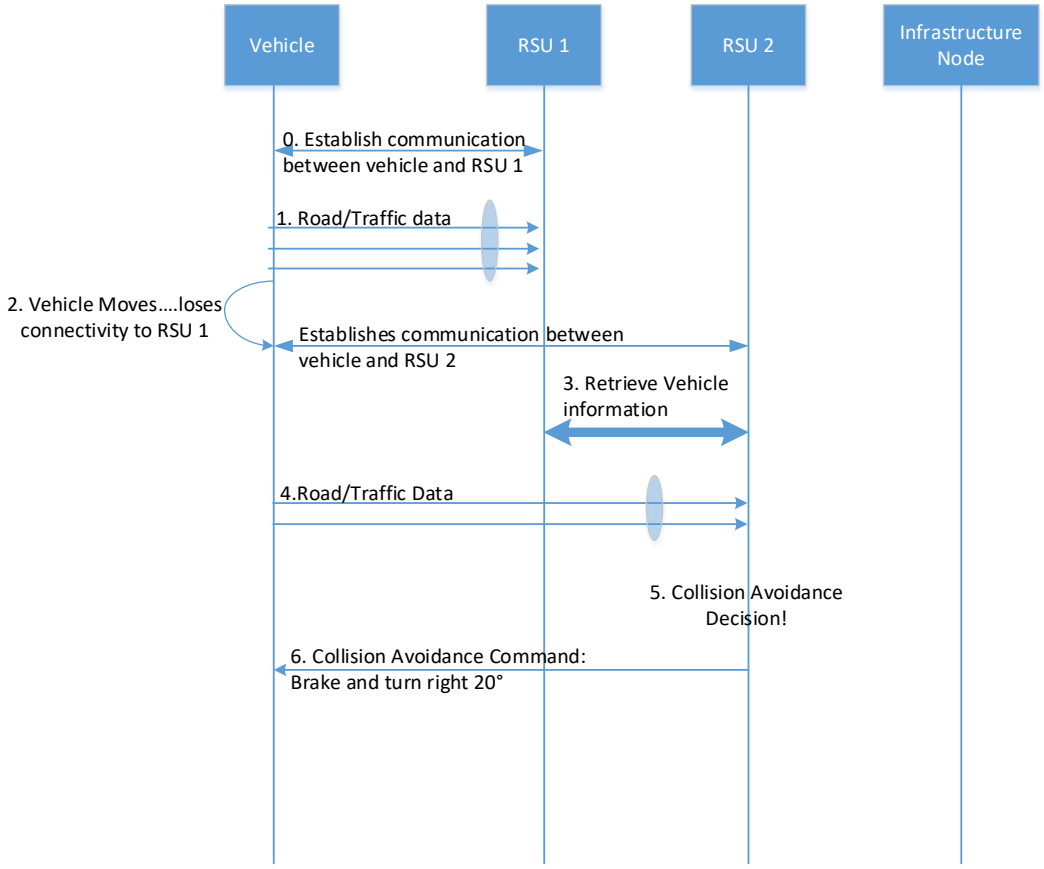

Figure 6.16.7-1: Alternative Flow for Vehicular Domain Service Continuity

0) Vehicle establishes communication with RSU 1.

1) Vehicle sends road/traffic data to RSU 1.

2) The vehicle moves, and establishes communication with RSU 2.

3) RSU 2 retrieves traffic data related to vehicle from RSU 1.

4) Vehicle sends road/traffic data to RSU 2.

5) Collision avoidance algorithm determines that a collision is imminent (based on vehicle data from RSU 2 and from RSU 1)

6) RSU 2 sends a collision avoidance command to the vehicle.

6.16.8 Post-conditions

N/A.

6.16.9 High Level Illustration

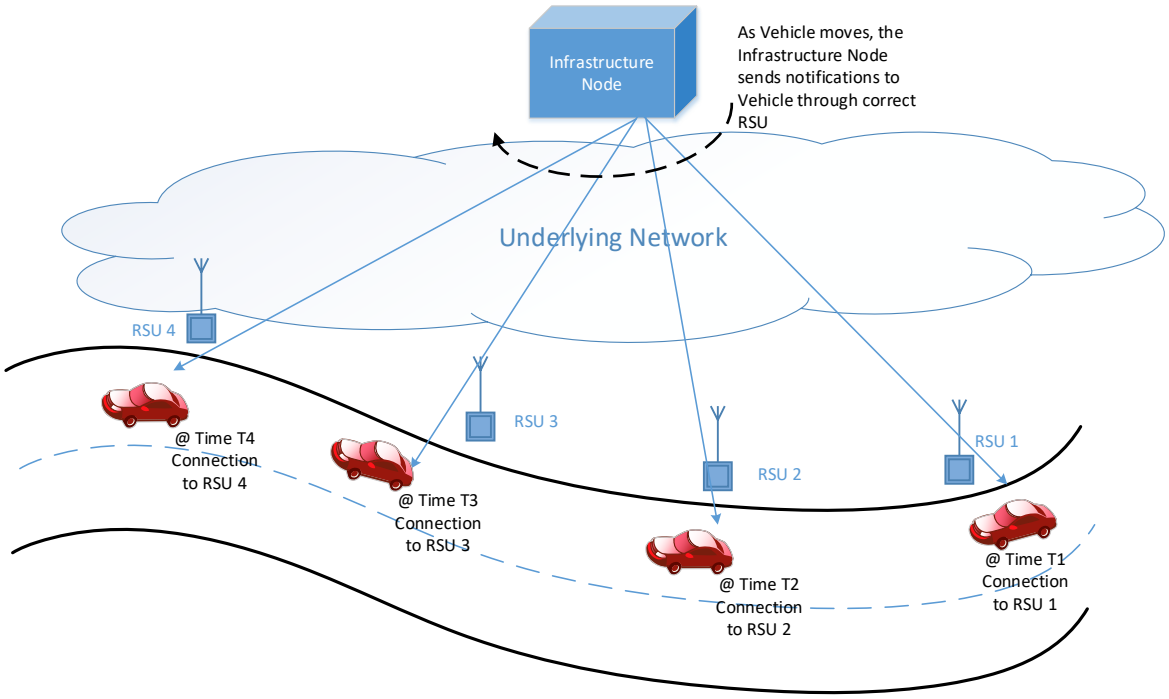

Figure 6.16.9-1: High Level Illustration Vehicular Domain Service Continuity

6.16.10 Potential requirements

1) The oneM2M system shall enable continuity of services to M2M devices as they move across various geographic points in the oneM2M system ([i.2] OSR-099).

6.17 Optimal Speed Recommendation

6.17.1 Description

The target vehicle is approaching the traffic light that is currently displaying red.

A local RSU station or the central server connected with local vehicles, traffic lights and other entities such as weather server, map server and remote vehicles collects the vehicle's precise location and the traffic light phase schedule as well as additional information such as weather, road shape, emergency situation etc. of the front road where the vehicle is heading to.

The local RSU station or the central server then calculates a specific optimal speed for the vehicle based on the collected information to reach the traffic light at the beginning of the next green phase and if necessary to avoid potential dangers.

The resulting speed is finally delivered to the driver who can avoid the unnecessary stop at the traffic light and unforeseeable hazards.

6.17.2 Source

REQ-2016-0067R02-new_Vehicle_usecase_TR-0026.DOC.

6.17.3 Actors

Target Vehicle

A Target Vehicle is a moving object which periodically provides its movement information such as time, location, velocity, heading, lateral and longitudinal acceleration to the central server so that it can utilize the information to calculate the optimal speed. After the calculation in the server, the Target Vehicle receives back the result directly from the server.

Traffic Light

A Traffic Light is located by a road and provides its phase schedule necessary for the server to calculate the optimal speed of the target vehicle.

Local RSU station or Central Server

A local RSU station or Central Server calculates the optimal speed for the target vehicle based on the gathered necessary local information from various sources including the target vehicle, the traffic light, other information servers and remote RSUs, central server, or vehicles.

Information Servers

There could be many kinds of additional Information Servers that can be connected with and utilized by the local RSU station or central server. For example, a weather server provides local weather data such as time, location, external air temperature, snow, rain and etc. and it is possible to detect potential road hazards via a map server.

Remote Vehicles

Remote vehicles automatically or manually report emergency situation to the local RSU station or server.

6.17.4 Pre-conditions

N/A.

6.17.5 Triggers

When the local RSU station or central server detects that the target vehicle has approached a few hundred meters (It should be more than 300 meters in order to safely adjust the current speed) in front of the traffic light, the server calculates the optimal speed and sends it to the target vehicle.

6.17.6 Normal Flow

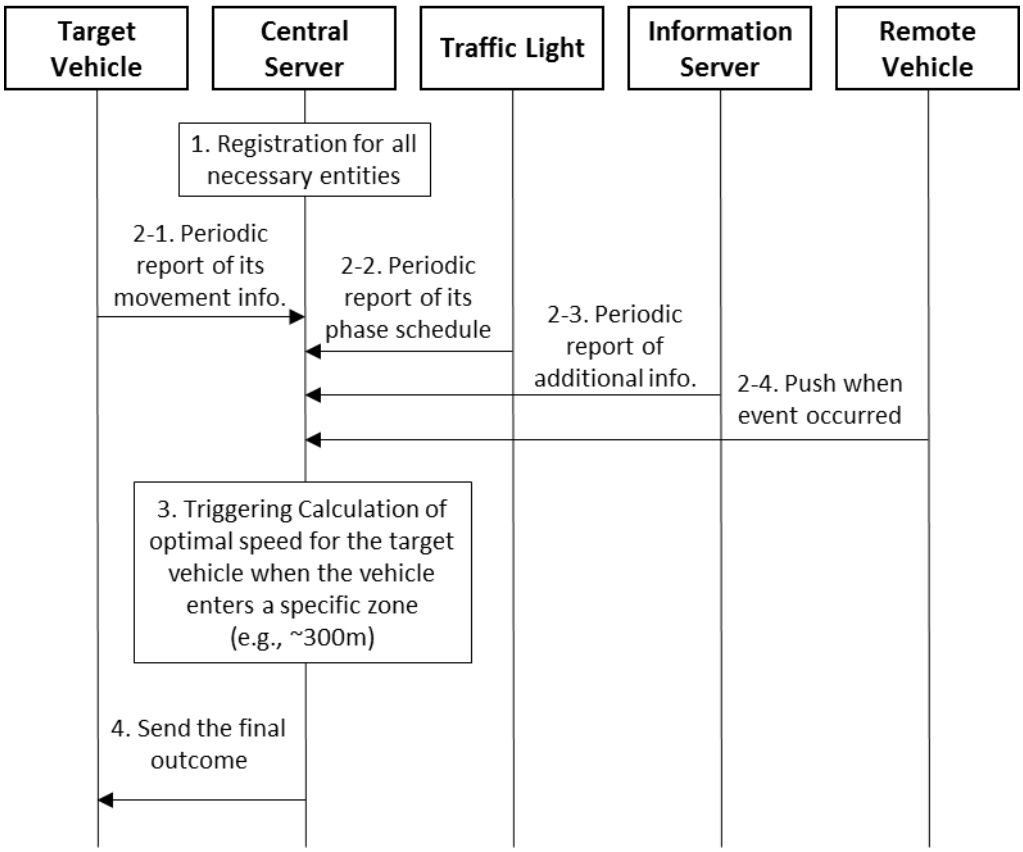

Figure 6.17.6-1 Normal Flow - Optimal Speed Recommendation

1) As the first step, associated entities register to the ITS system and communicate with the local RSU station or central server except remote vehicles which do not necessarily need to communicate because they are moving object and this scenario works based on local information. It is enough to know just the location of the remote vehicle and what happens around it when it is in the vicinity of the traffic light.

2) The local RSU station or central server gathers information to calculate the optimal speed of the target vehicle.

   2.1) The target vehicle periodically provides its movement information including heading, time, location, velocity, lateral and longitudinal acceleration.

   2.2) The traffic light periodically provides its phase schedule when to be red, yellow or green.

   2.3) Additional information server provides further useful data as requested by the local RSU station or central server. Information such as slippery zone, high accident frequency location, or sharp curve could be examples.

   2.4) Remote vehicles can also provide more information to the server in case an accident or something dangerous happens. This can be done automatically as programmed or manually by drivers. Note that the local RSU station or central server does not communicate with remote vehicles since it's impossible to predict which vehicles may have necessary information. Therefore remote vehicles should initiate communication with the local RSU station or central server only when necessary, typically when they reach the vicinity of the traffic light.

3) When the target vehicle has reached a specific area (It could be 300m in front of the traffic light) ahead of the traffic light, the local RSU station or central server starts calculating the optimal speed for the target vehicle to pass through the traffic light with no stop based on the collected information. If any additional information indicates there could be safety risk, the local RSU station or central server should consider them as well for the calculation.

4) The final outcome from the server is delivered to the target vehicle and shown to the driver who can follow the speed recommendation.

6.17.7 Alternative flow

N/A.

6.17.8 Post-conditions

N/A.

6.17.9 High Level Illustration

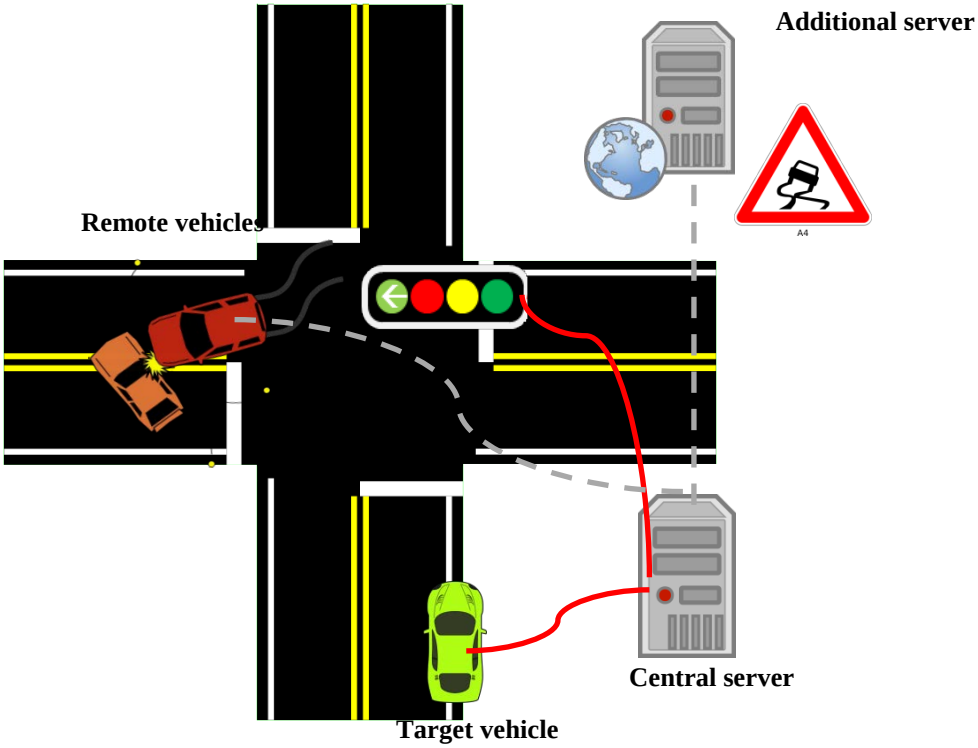

Figure 6.17.9-1 High Level Illustration - Optimal Speed Recommendation

- Solid lines in red are mandatory while dashed lines in grey are optional for this use case.

6.17.10 Potential requirements

1) The oneM2M system shall support real time data transmission for moving object to have enough time to react to the received information.

2) The oneM2M system shall be able to verify geographical location information from moving objects regardless of information accuracy.

3) The oneM2M system shall be able to verify time synchronization among multiple sources that provide inputs for a single output.

4) The oneM2M system shall be able to guarantee network stability and reliability for applications that can have safety impacts

5) The oneM2M system shall apply high-level security (e.g. protection from malicious hacker) for applications that can have safety impacts.

6) The oneM2M system shall provide privacy protection mechanism at the central server.

6.18 Autonomous driving

6.18.1 Description

Autonomous driving (AD) refers to the capability of the vehicle to drive from one location to another, without intervention from humans, and in a safe way, without incurring damage to surroundings (pedestrians, buildings, other vehicles) and to its (vehicle's) passengers.

Different levels of automation are shown in **Error! Reference source not found.**. From top to bottom we can see increasing levels of automation, with 'Full automation' (level 5) being autonomous driving.

For autonomous driving, it is essential for vehicle to have full information on its own state and state of its surroundings. In that respect data that is lost, or which arrives with large delay, is introducing uncertainty in the vehicle's model of the surroundings. This uncertainty will typically result in vehicle taking appropriate measures to deal with it, typically decreasing the speed of the vehicle, or stopping it altogether. Reasoning behind this is that for safety critical applications, safety is in the first place.Modern vehicles are equipped with number of sensors which are increasing comfort, fuel efficiency and safety of vehicles. That is done by using a variety of sensors that measure and collect data about vehicle (for example - ABS sensors, brake switch, speed, location, etc.), but are also equipped with sensors to observe state of its surroundings - using radar, camera, short range distance measurement sensors, LIDAR, etc.

Further, the roads and surrounding infrastructure are becoming more instrumented with sensors and are able to communicate. The possibility of interconnecting infrastructure sensors (cameras, traffic light radars, road sensors , etc.) and thus exchanging data with vehicles may lead to new ways to design autonomous vehicle systems, thereby reducing cost, while increasing robustness and reliability of autonomous driving.

Other connected objects (pedestrian's smartphones, for example) may also act as additional sources of data for autonomous driving vehicle, thereby contributing to improved efficiency, accuracy and safety of the Autonomous driving functions.

System supporting combining all these sources of data will enable pushing the driving automation to the higher levels of automation (as described in clause 5.4), ultimately to one where the driver is out of the (control/driving) loop. Furthermore, by making autonomous cars an IoT entity, this will enable larger groups of developers to create IoT/AD services.

6.18.2 Source

REQ-2017-0001R03-Autonomous_driving.

6.18.3 Actors

Vehicle Driver:

Driver sits in its normal (driving) position, and is in position to take corrective action (level 3, level 4) when prompted to do so by the autonomous driving system. There is no vehicle driver at level 5 of automation.

IoT platform provider

It operates an IoT platform which is collecting data from vehicles, other participants in traffic (pedestrians, cyclists), from roads and associated infrastructure (traffic lights, cameras, etc.).

Autonomous driving application provider

Party that is providing Autonomous driving application (ADApp) which runs on AS (Application Servers). AS connects to the IoT platform, and from there it collects relevant data needed to run an ADApp - for example LDM (Local Dynamic Map).

Communication Network provider/operator

Provides and facilitates connectivity between vehicles, roads and associated infrastructure. This covers both wireless (for example LTE, LTE-V2X, ITS-G5 and WAVE) and fixed connections. The Network supports receiving requests for data transfers with required latency and with required packet losses. It is not expected or mandated that single network operator provides all of connectivity.

IoT Device

IoT devices are embedded in a vehicles, roads and associated infrastructure, as well as in devices used by other participants in traffic (pedestrians, cyclists). Each IoT device collects and sends data to IoT platform, and can receive data from platform.

6.18.4 Pre-conditions

The vehicle supports autonomous driving - meaning it is capable of transmitting and receiving data from other vehicles, road and other infrastructure, other participants in traffic (pedestrians, cyclists), and based on own collected data and received data from ADApp on making decision and acting upon it.

6.18.5 Triggers

Autonomous driving mode can be activated by vehicle driver, or it can be activated by default. How the HMI (Human Machine Interface) works and which commands are given to the vehicle is out of scope of the present document.

6.18.6 Normal Flow

1) The vehicle continuously collects data from sensors within the vehicle and sends it to the IoT platform. The collected data includes information from internal state of the vehicle (ABS status, brake switch, accelerator pedal, etc.) as well as data on surroundings of the vehicle (radar, LIDAR, cameras, etc.).

2) Road infrastructure (roads, traffic lights, cameras, etc.) continuously collects data from its sensors and sends it to the IoT platform.

3) Devices belonging to other participants in traffic (pedestrians, cyclists, etc.) continuously collect data from its sensors and send it to the IoT platform.

4) The IoT platform hosts the collected data from abovementioned sources, and upon request it will send it to AS (Application Server) where Autonomous driving application (ADApp) is running. ADApp is subscribed to the IoT data from all participants in traffic. Note that in order to protect privacy of participants in traffic, the accessibility of the collected data has to be sufficiently restricted and access to collected data is covered by regional regulatory obligations.

5) Processed information from ADApp is sent back to IoT platform, and is made available to all ADApp subscribed participants in traffic.

6) Each participant in traffic is responsible for interpretation and action based on received ADApp data.

6.18.7 Alternative Flow

None.

6.18.8 Post-conditions

Vehicle stays in autonomous driving mode until removed from it by driver.

6.18.9 High Level Illustration

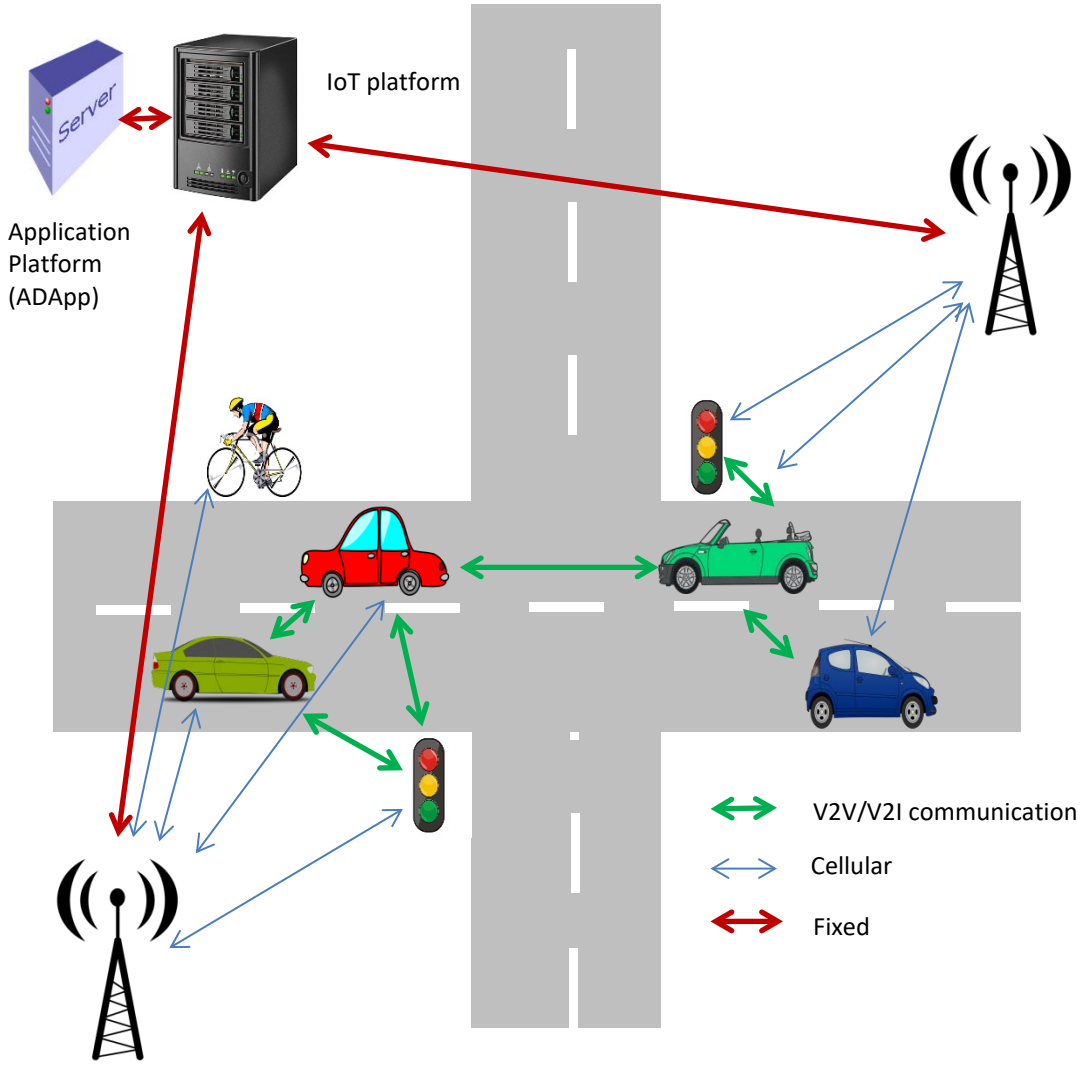

Figure 6.18.9-1: Example of IoT data streams and corresponding communication networks

6.18.10 Potential Requirements

1) The M2M system shall support the data to be transmitted to IoT platform with strict timing and packet loss requirements, determined by application ([i.2] CMR-016).

2) The M2M system shall support the data to be transmitted from IoT platform to subscribed devices with highest priority, with strict timing and packet loss requirements, determined by application ([i.2] CMR-017).

3) For each source of relevant autonomous driving data which is sent to the IoT platform, the oneM2M System shall be able to detect and report the missing data in time series. For each vehicle receiving data from ADApp, the oneM2M System shall be able to detect and report the missing data in time series ([i.2] CMR-018 and CMR-019).

7 Overview of Potential Requirements

Potential requirements from all vehicular domain use cases collected in this technical report are summarized as follows,

1) The M2M application System shall enable the M2M Devices to exchange M2M application to diagnostic data periodically with the M2M Application in the network domain ([i.2] OSR-118).

NOTE 1: This requirement addresses the use case 6.1.

2) The M2M System shall enable the M2M Application to configure the notification interval in the M2M Devices ([i.2] OSR-112).

NOTE 2: This requirement addresses the use case 6.1.

3) The M2M system shall support a mechanism to describe the syntax and semantics format of the M2M application diagnostics data exchanged between the M2M Devices and the M2M Application in the network domain ([i.2] OSR-119).

NOTE 3: This requirement addresses the use case 6.1.

4) The M2M service shall be able to provide the mechanism for authorization for integrity-checking and installing processes of software/hardware/firmware component(s) on M2M Device(s) ([i.2] SER-013).

NOTE 4: This requirement addresses the use case 6.2.

5) The M2M system shall be able to support authentication using device key on the integrity check for M2M Device(s) ([i.2] SER-073).

NOTE 5: This requirement addresses the use case 6.2.

6) The M2M Device shall be able to support HSM (Hardware Security Module) to protect its integrity depending on the security level requirement ([i.2] SER-023).

NOTE 6: This requirement addresses the use case 6.2.

7) A M2M System shall support communication between M2M Platform and a M2M device either directly or via a gateway ([i.2] OSR-113).

NOTE 7: This requirement addresses the use case 6.3.

8) A M2M System shall be able to exchange information between M2M applications via M2M Platform ([i.2] OSR-114).

NOTE 8: This requirement addresses the use case 6.3.

9) A M2M System shall be able to take actions according to the received service requests from M2M Applications.

NOTE 9: This requirement addresses the use case 6.3.

10) A M2M system shall be able to support service requests from M2M applications for communication with QoS requirement, such as, higher delivery priority, reliable delivery, etc. ([i.2] OSR-115).

NOTE 10:This requirement addresses the use case 6.3.

11) A M2M System shall support mutual-authentication among M2M device, M2M gateway, M2M platform and M2M Application ([i.2] SER-040).

NOTE 11:This requirement addresses the use case 6.3.

12) The information sent by a M2M device or the M2M platform or a M2M application shall use cryptographic technology to ensure information authentication and information integrity ([i.2] SER-068).

NOTE 12:This requirement addresses the use case 6.3.

13) A M2M system shall permit information being provided in anonymous way ([i.2] SER-074).

NOTE 13: This requirement addresses the use case 6.3.

14) A command issued by a M2M System shall be able to have time expiration or geography restriction ([i.2] OSR-116).

NOTE 14: This requirement addresses the use case 6.3.

15) Provisioning, installation, configuration and registration method of terminal system:

- Especially for the case of overlapping two different system for DTG management system (owns and manages the device) and the application system using DTG data (utilizing the data from the device) ([i.2] OSR-134).

NOTE 15: This requirement addresses the use case 6.4.

16) DTG/FMS data storing method and delivery protocol:

- There is no dominant standard specifying data formats and protocols for vehicle related applications.

NOTE 16: This requirement addresses the use case 6.4.

17) Vehicle location based service method:

- M2M service platform is expected to provide the service capability supporting location based service ([i.2] OSR-120).

NOTE 17: This requirement addresses the use case 6.4.

18) Control, configuration, error logging, and management method for the terminal system Over-The-Air:

- M2M service platform is expected to provide the service capability supporting the Over-The-Air management ([i.2] OSR-121).

NOTE 18: This requirement addresses the use case 6.4.

19) The M2M system shall provide the capability for an M2M device to maintain registration with multiple entities simultaneously ([i.2] OSR-122).

NOTE 19: This requirement addresses the use case 6.5.

20) The registration shall be able to include information that identify the peer entity, and other information such as its management privilege, subscription etc., that are necessary for the conduct of the respective peer relationships([i.2] OSR-135).

NOTE 20: This requirement addresses the use case 6.5.

21) It shall be possible for some registrations to hold the complete set of information context about the peer entity. This is referred to as "full registration"([i.2] OSR-136).

NOTE 21: This requirement addresses the use case 6.5.

22) It shall be possible for some registrations to hold only a subset of information context about the peer entity. This is referred to as "lightweight registration"([i.2] OSR-137).

NOTE 22: This requirement addresses the use case 6.5.

23) It shall be possible for "lightweight registration" at different entities that pertain to a common peer entity, to hold different sets of information, if needed, about the common peer entity([i.2] OSR-138).

NOTE 23: This requirement addresses the use case 6.5.

24) It shall be possible to correlate the "full registration" and the "lightweight registration" that pertain to a common peer entity([i.2] OSR-139).

NOTE 24: This requirement addresses the use case 6.5.

25) It shall be possible to distinguish the "full registrations" and the "lightweight registrations" that pertain to a common peer entity([i.2] OSR-140).

NOTE 25:This requirement addresses the use case 6.5.

26) The M2M service platform shall be able to support the time-based policies to access the Underlying network ([i.2] CMR-014).

NOTE 26:This requirement addresses the use case 6.6 "Use cases for Taxi Advertisement".

27) The oneM2M System shall enable discovery of M2M Application Servers, M2M Management Servers and M2M Devices available to an M2M Gateway for data exchange ([i.2] OSR-086).

NOTE 27:This requirement addresses the use case 6.7.

28) The oneM2M System shall enable discovery of M2M Gateways available to a M2M Management Server and an M2M Device for data exchange ([i.2] OSR-087).

NOTE 28:This requirement addresses the use case 6.7.

29) The oneM2M System shall be able to support the capabilities for data repository (i.e. to collect/store) and for data transfer from one or more M2M Devices or M2M Gateways, for delivery to one or more M2M Gateways via M2M Area Network without any assistance or instruction of M2M Management Servers and M2M Application Servers ([i.2] OSR-088).

NOTE 29:This requirement addresses the use case 6.7.

30) Upon request from M2M Application Server, an M2M Gateway or M2M device shall enable functions that pre-process (e.g. average) M2M data before providing them to the recipient ([i.2] 0SR-104).

NOTE 30:This requirement addresses the use case 6.7.

31) Upon request, an M2M Gateway or M2M device shall enable functions that erase M2M data (e.g. that have been sent or could not be sent to the recipient within a certain time) based on criteria from an M2M Application ([i.2] OSR-105).

NOTE 31:This requirement addresses the use case 6.7.

32) An M2M Gateway and/or an M2M Device shall be able to broadcast to all M2M Devices and/or M2M Gateways in the vicinity its need to receive/deliver specific data ([i.2] OSR-106).

NOTE 32:This requirement addresses the use case 6.7.

33) M2M Gateway and/or M2M Device shall be able to establish a connection to each other if it is able to receive/deliver the required specific data ([i.2] OSR-107).

NOTE 33:This requirement addresses the use case 6.7.

34) The oneM2M System shall enable M2M Gateways to set conditions used for processing jointly data subscriptions and distribute the resulting notifications according to the set conditions ([i.2] OSR-108).

NOTE 34:This requirement addresses the use case 6.7.

35) The oneM2M System shall enable subscriptions to changes to multiple resources which aim to generate notifications if and only if the expected changes to those resources occur concurrently ([i.2] OSR-109 and OSR-110).

NOTE 35:This requirement addresses the use case 6.7.

36) OneM2M System shall be able to send the information to intended vehicles by unicast, multicast and/or broadcast ([i.2] OSR-123).

NOTE 36:This requirement addresses the use case 6.8.

37) oneM2M System shall be able to securely transfer the information by using an appropriate method such as digital signature ([i.2] SER-069).

NOTE 37: This requirement addresses the use case 6.8.

38) oneM2M System shall be able to transfer the information on real-time basis for feeding back current road states to automatic driving control. The feedback time should be less than a few seconds (the distance between vehicles normally corresponds to a few seconds) to avoid unnecessary speed down/stop of following vehicles ([i.2] OSR-124).

NOTE 38: This requirement addresses the use case 6.8.

39) OneM2M system shall be able to guarantee its reliability in order to receive/feedback massages from/to related vehicles ([i.2] OSR-125).

NOTE 39: This requirement addresses the use case 6.8.

40) oneM2M System shall enable sharing of service information between devices/GWs based on proximity ([i.2] OSR-126).

NOTE 40: This requirement addresses the use case 6.8.

41) oneM2M System shall enable sending and receiving of service information between devices/GWs with minimized interruption ([i.2] OSR-127).

NOTE 41: This requirement addresses the use case 6.8.

42) The oneM2M System shall enable the cancellation of continuous data collection and/or the deletion of collected data when pre-defined conditions are met ([i.2] OSR-089).

NOTE 42: This requirement addresses the use case 6.9 "Use Case on Vehicle Data Wipe Service".

43) The oneM2M System shall enable pre-defined conditions to be protected from unauthorized modification ([i.2] SER-050).

NOTE 43: This requirement addresses the use case 6.9.

44) The oneM2M System shall enable the deletion of M2M data produced/stored by the M2M Devices/Gateways based on request from an authorized entity ([i.2] SER-051).

NOTE 44: This requirement addresses the use case 6.9.

45) The oneM2M System shall support reporting of Geo-Fence based Location event of the target M2M Device to the M2M Application based on the Application's configuration. (i.2 OSR-047)

NOTE 45: This requirement addresses the use case 6.10 "Vehicle Management based on Geo-Fence".

46) The oneM2M System shall support the M2M Application setting the configuration for Geo-Fence based location service ([i.2] OSR-117).

NOTE 46: This requirement addresses the use case 6.10.

47) The oneM2M System shall be able to prevent unauthorized modification of the firmware of M2M Device ([i.2] SER-064).

NOTE 47: This requirement addresses the use case 6.11.

48) The oneM2M System shall be able to detect unauthorized modification of the firmware of M2M Device ([i.2] SER-065).

NOTE 48: This requirement addresses the use case 6.11.

49) The oneM2M System shall be able to stop operation of M2M device when it is updated with wrong firmware ([i.2] SER-066).

NOTE 49: This requirement addresses the use case 6.11.

50) The oneM2M System shall be able to support security mechanisms to protect their cryptographic keys and cryptographic operations by using tamper resistant elements such as TPM (Trusted Platform Module), HSM (Hardware Security Module) and SIM (Subscriber Identity Module) ([i.2] SER-070).

NOTE 50: This requirement addresses the use case 6.11.

51) The oneM2M System shall be able to prevent malfunction of M2M Device caused by receiving unsolicited messages or information ([i.2] SER-067).

NOTE 51: This requirement addresses the use case 6.11.

52) The M2M System shall support mobile/portable M2M Gateway and/or Device ([i.2] OSR-128).

NOTE 52: This requirement addresses the use case 6.12.

53) The M2M System shall support to distinguish the event levels for reporting and handle it differentially.

EXAMPLE: The event levels may be divided into normal and urgent event ([i.2] OSR-032).

NOTE 53: This requirement addresses the use case 6.12.

54) Based on the condition of the M2M Gateway and/or Device, the M2M System shall change the reporting (or subscription) mechanisms and/or configurations related to a service ([i.2] OSR-033).

NOTE 54: This requirement addresses the use case 6.12.

55) The M2M System shall support to process access right requests of a resource and grant the requests if the required conditions are met ([i.2] SER-071).

NOTE 55: This requirement addresses the use case 6.12.

56) The M2M System shall support mechanisms to correlate charging data/records from different M2M Application Service Providers ([i.2] CHG-002 and CHG-007).

NOTE 56: This requirement addresses the use case 6.13.

57) The M2M System shall support triggering M2M Devices to report on-demand regarding collected data from other M2M Devices ([i.2] OSR-072).

NOTE 57: This requirement addresses the use case 6.13.

58) The oneM2M system shall enable the M2M Infrastructure to facilitate direct communication between two or more different M2M devices without having registered with one another ([i.2] OSR-130).

NOTE 58: This requirement addresses the use case 6.14.

59) The oneM2M system shall be able to support the enforcement and management (e.g. update) of the privacy policies for vehicle location information ([i.2] SER-026).

NOTE 59: This requirement addresses the use case 6.15.

60) The oneM2M system shall be able to apply the privacy policies configured in the oneM2M system to those vehicles whose location information is obtained by the oneM2M system itself ([i.2] SER-026).

NOTE 60: This requirement addresses the use case 6.15

61) The oneM2M system shall be able to request the underlying network to perform, on behalf of the oneM2M system, the privacy policy decision concerning the location information if the location information is obtained by the underlying network location service ([i.2] SER-062 and SER-063).

NOTE 61: This requirement addresses the use case 6.15.

62) The oneM2M system shall enable continuity of services to M2M devices as they move across various geographic points in the oneM2M system ([i.2] OSR-099).

NOTE 62: This requirement addresses the use case 6.16.

63) The oneM2M system shall support real time data transmission for moving object to have enough time to react to the received information.

NOTE 63:This requirement addresses the use case 6.17.

64) The oneM2M system shall be able to verify geographical location information from moving objects regardless of information accuracy ([i.2] OSR-131).

NOTE 64:This requirement addresses the use case 6.17.

65) The oneM2M system shall be able to verify time synchronization among multiple sources that provide inputs for a single output ([i.2] OSR-132).

NOTE 65:This requirement addresses the use case 6.17.

66) The oneM2M system shall be able to guarantee network stability and reliability for applications that can have safety impacts ([i.2] OSR-133).

NOTE 66:This requirement addresses the use case 6.17.

67) The oneM2M system shall apply high-level security (e.g. protection from malicious hacker) for applications that can have safety impacts ([i.2] SER-075).

NOTE 67:This requirement addresses the use case 6.17.

68) The oneM2M system shall provide privacy protection mechanism at the central server ([i.2] SER-072).

NOTE 68:This requirement addresses the use case 6.17.

69) The M2M system shall support the data to be transmitted to IoT platform with strict timing and packet loss requirements, with requirements defined by application determined by application ([i.2] CMR-016).

NOTE 69:This requirement addresses the use case 6.18.

70) The M2M system shall support the data to be transmitted from IoT platform to subscribed devices with highest priority, with strict timing and packet loss requirements, determined by application ([i.2] CMR-017).

NOTE 70:This requirement addresses the use case 6.18.

71) For each source of relevant autonomous driving data which is sent to the IoT platform, the oneM2M System shall be able to detect and report the missing data in time series. For each vehicle receiving data from the Autonomous Driving Application, the oneM2M System shall be able to detect and report the missing data in time series ([i.2] CMR-018 and CMR-019).

NOTE 71:This requirement addresses the use case 6.18.

8 High Level Architecture

8.1 Introduction

The use cases in the vehicular domain discussed in the present document are listed in Table 8.1-1.

Table 8.1-1: Use cases in the vehicular domain

Use case No.	Title	Description
1	Vehicular Diagnostic & Maintenance Report	See clause 6.1
2	Use Case on Remote Maintenance Services	See clause 6.2
3	Traffic Accident Information Collection	See clause 6.3
4	Fleet Management Service using DTG (Digital Tachograph)	See clause 6.4
5	Use cases for Electronic Toll Collection (ETC) service	See clause 6.5
6	Use cases for Taxi Advertisement	See clause 6.6
7	Use Case on Vehicle Data Service	See clause 6.7
8	Smart Automatic Driving	See clause 6.8
9	Use Case on Vehicle Data Wipe Service	See clause 6.9
10	Vehicle Management based on Geo-Fence	See clause **Error! Reference source not found.**
11	Use Case on Secure Over-The-Air Firmware Update for Automotive ECUs	See clause 6.11
12	Car/Bicycle Sharing Services	See clause 6.12
13	Smart Parking	See clause 6.13
14	Vehicle Broadcasting without Registration	See clause 6.14
15	Vehicle location privacy protection	See clause 6.15
16	Vehicle Domain service continuity	See clause 6.16
17	Optimal Speed Recommendation	See clause 6.17
18	Autonomous driving	See clause 6.18

Though these use cases are focusing on the vehicular domain, the ways of functional deployment are different. The following clauses provide three types of high level oneM2M architecture mapping for these use cases.

8.2 Vehicular Architecture Type 1

Figure 8.2-1 illustrates the first type of high level oneM2M architecture in vehicular domain. In this type, vehicular domain applications and the M2M service platform described in each use case are mapped to the IN (Infrastructure Node). Each vehicle is equipped with a communication module which supports cellular network or other wireless communication technologies, and the vehicle can connect to a service platform via the communication module. Alternatively, an external unit such as the smartphone acts as both a communication module and a GW. In this case, the external unit would connect to in-vehicle network via some network technologies such as Bluetooth.

The GW which provides some functions defined in oneM2M acts as the MN or the ASN. The GW would be the in-vehicle unit such as an ECU (Electronic Control Unit) or the external unit such as the smartphone. The GW is connecting to in-vehicle units and external sensors via an in-vehicle network such as CAN (Controller Area Network) or other communication technology such as Bluetooth.

The in-vehicle units such as ECUs or external sensors act as the ADN or the non-oneM2M device node.

The use case 1 to 3, 6 to 11, 13, 15 and 18 are categorized into this type.

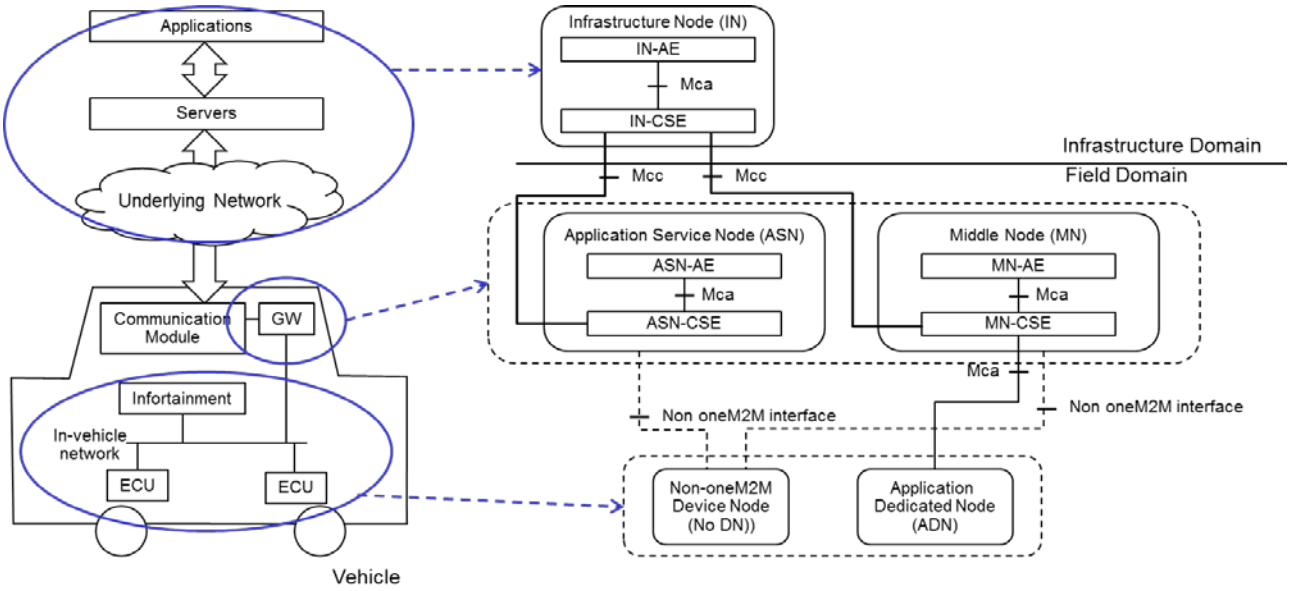

Figure 8.2-1: High Level Architecture Type 1 in Vehicular Domain

8.3 Vehicular Architecture Type 2

Figure 8.3-1 illustrates the second type architecture. Similar to the first type, vehicular domain applications and the M2M service platform described in each use case are mapped to the IN. Each vehicle is equipped with a communication module to connect to a road side unit.

The road side unit which provides some functions defined in oneM2M acts as the MN.

In this type, the GW in vehicle acts as the MN or the ASN. The in-vehicle units such as ECUs or external sensors act as the ADN or the non-oneM2M device node.

The use cases 3, 4, 5, 12 and 18 are categorized into this type.

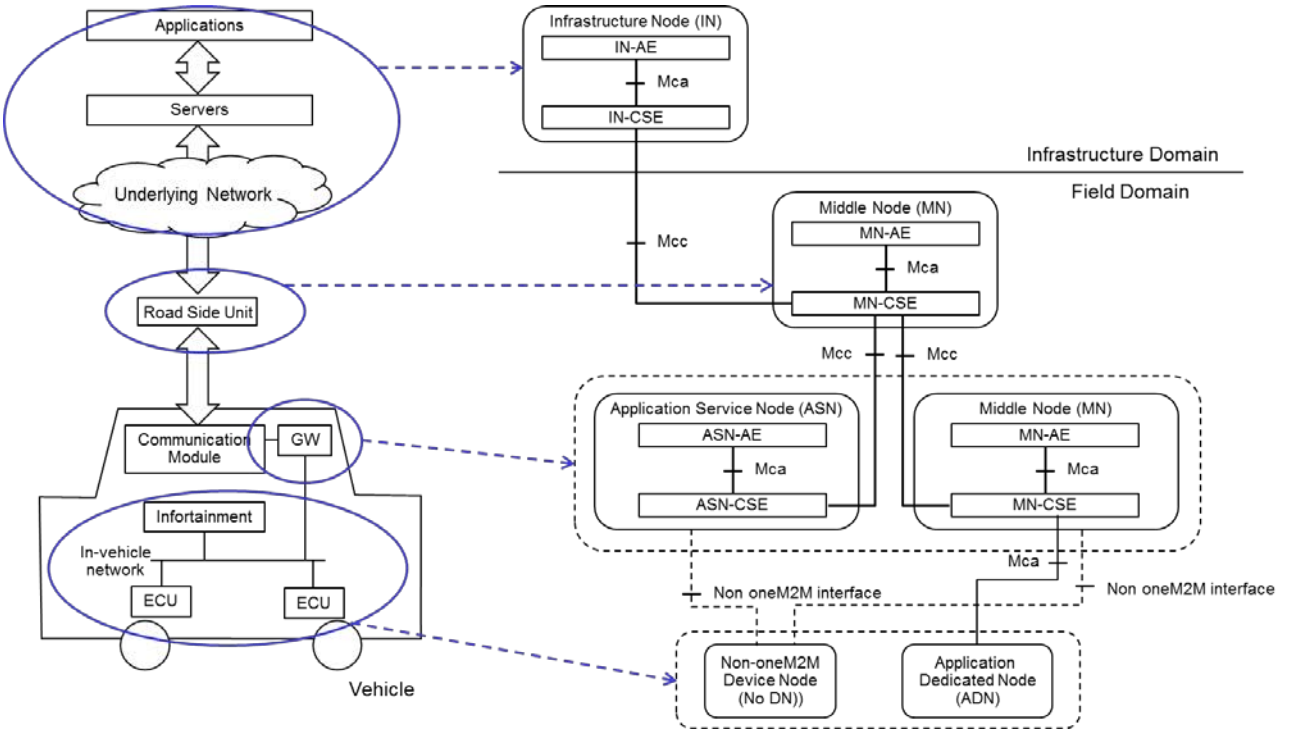

Figure 8.3-1: High Level Architecture Type 2 in Vehicular Domain

8.4 Vehicular Architecture Type 3

Figure 8.4-1 illustrates the third type. In this type, a vehicle to vehicle communication technology is supported in each vehicle. To support the vehicle to vehicle communication in the oneM2M architecture, the GW acts as the MN.

The use case 3, 14 and 18 are categorized into this type.

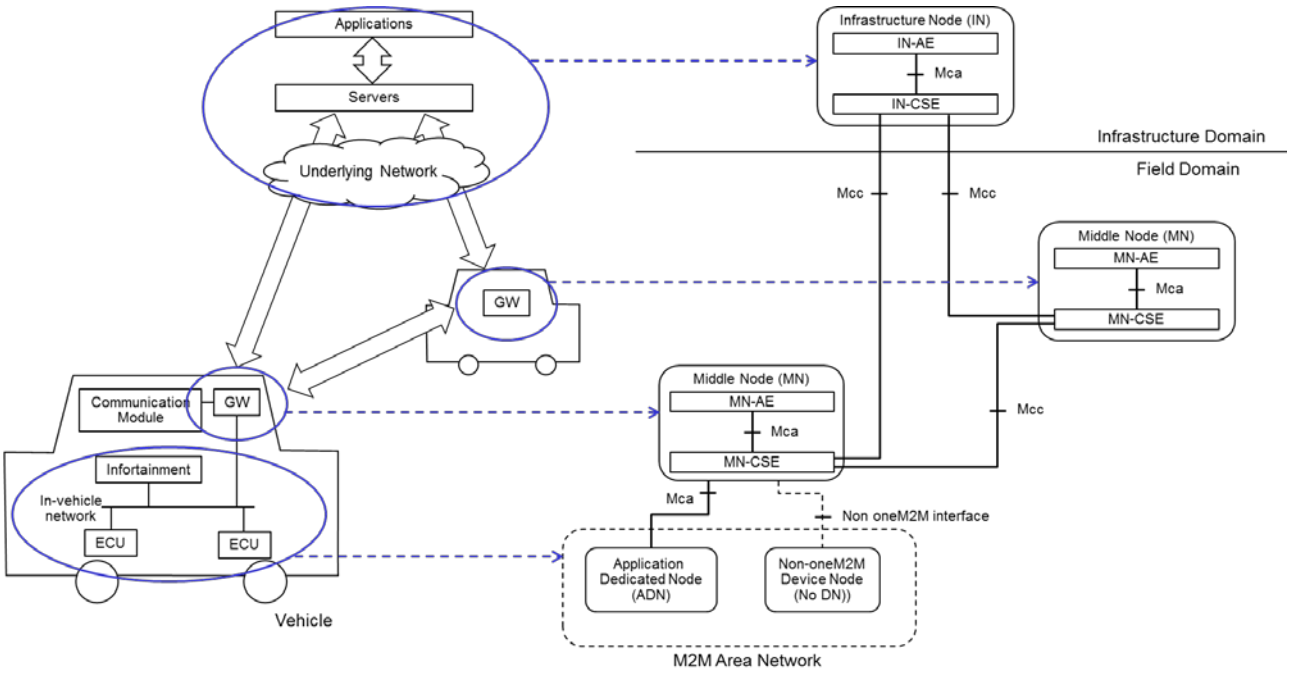

Figure 8.4-1: High Level Architecture Type 3 in Vehicular Domain

8.5 Vehicular Architecture Type 4

Figure 8.5-1 illustrates the fourth type of architecture - which is a variation to Architecture Type 2. The main difference is that the vehicle has no GW functionality providing oneM2M services to the in-vehicle units. However, the vehicle still has a communication module to communicate with RSUs.

In this architecture, the in-vehicle units, such as ECUs or external sensors, act as ADNs or non-oneM2M device nodes. The RSU, which provides functions defined in oneM2M, acts as an MN.

The use case 5, 16 and 18 are categorized into this type.

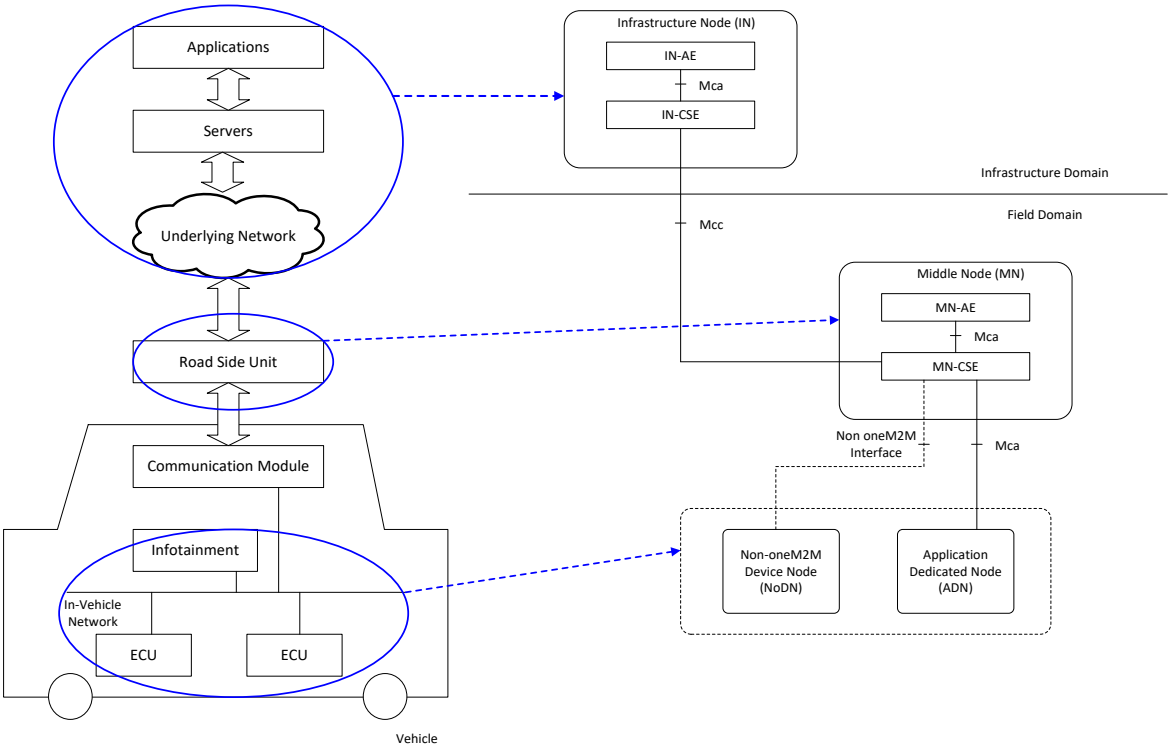

Figure 8.5-1: High Level Architecture Type 4 in Vehicular Domain

9 Key Issues for Enablement of Vehicular Domain

This clause summarizes key issues resulting from the use cases and requirements identified. Each key issue is captured in a separate clause.

9.1 Key Issues 1: Location

Geographical location information is important in vehicular services, because a node in vehicular domain moves dynamically. As shown in use cases in clause 6, location is the key factor of vehicular services. For instance, the procedures of vehicular services could be triggered by the change of location. In other cases, the vehicular service would be provide to the vehicles located in its dedicated area.

In oneM2M Release 2, Location CSF (LOC CSF) gives the following location functions:

- A location server in the Underlying Network.
- A GPS module in an M2M device.
- Information for inferring location stored in other Nodes, such as short-range communication establishment with a gateway.

Through the analysis of existing use cases in this technical report, the following sub-clauses contain consideration points about location for the vehicular domain.

9.1.1 Accuracy of geographic location

In the vehicular domain, each vehicle would generally be equipped with a device to obtain geographical location (geo-location) information. Through the location device, a CSE in a vehicle periodically updates its geo-location and forward it to the oneM2M platform. The accuracy of geo-location information and the applicable use cases are dependent on the capability of the geo-location information source. Table 9.4-1 summarizes the required accuracy of the geo-location information, and the methods to obtain the required information accurately.

Table 9.1-1: Required position accuracy of location

Title	AREA	ROAD	LANE
Accuracy of geo-location (margin of error)	Less than dozens of meters	Less than a few meters	Less than dozens of centimetres
Example of methods	An Underlying Network (e.g. Cell-ID from 3GPP, ITS-G5) Information for inferring location stored in other Nodes	GPS Assisted GPS Underlying Network (e. g. Cell-ID from RSUs) Information for inferring location stored in other Nodes	RTK-GPS Recent GNSS/RNSS
Use cases	6.8, 6.10	6.4, 6.12, 6.13	6.x

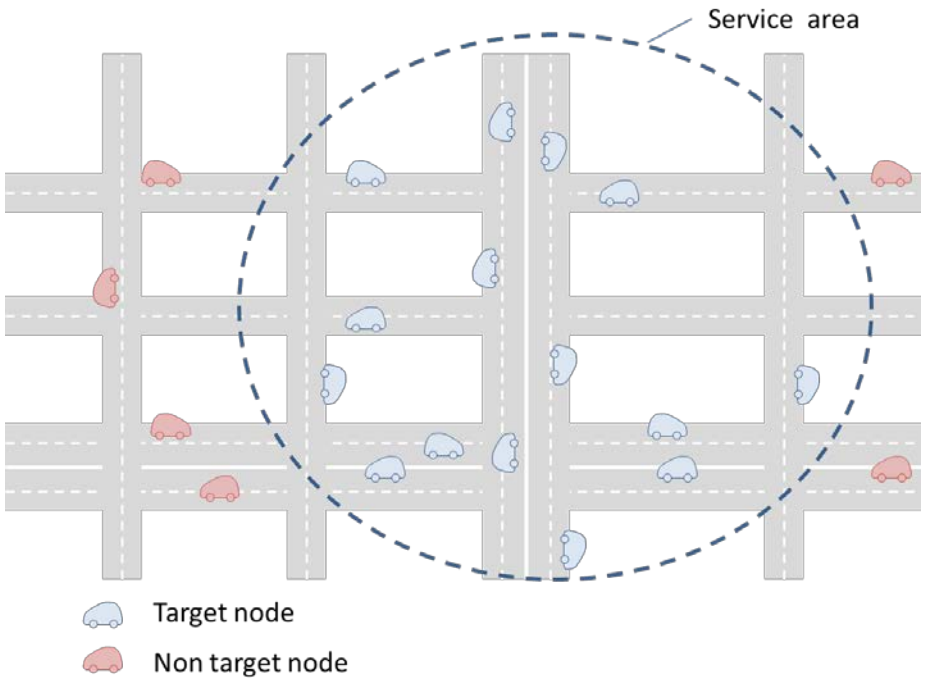

Figure 9.1.1-1: AREA

To deliver a content to vehicles located in a dedicated area (a few hundred meters), the "AREA level" accuracy of geo-location would be required. Figure 9.1.1-1 illustrates the concept of "AREA" use case. In this case, an acceptable error range should be less than dozens of meters. For instance, an underlying network can provide location information with the required accuracy of geo-location based on Cell-ID.

Geo-Fence is also applicable for the AREA level accuracy of geo-location. In oneM2M Release 2, Geo-Fence is usable as the resource type <*locationPolicy*>. For instance, in use case 6.8, when a traffic accident occurs, oneM2M platform sets Geo-Fence around the accident point. Then the oneM2M platform can provide the content to the vehicles in the Geo-Fence.

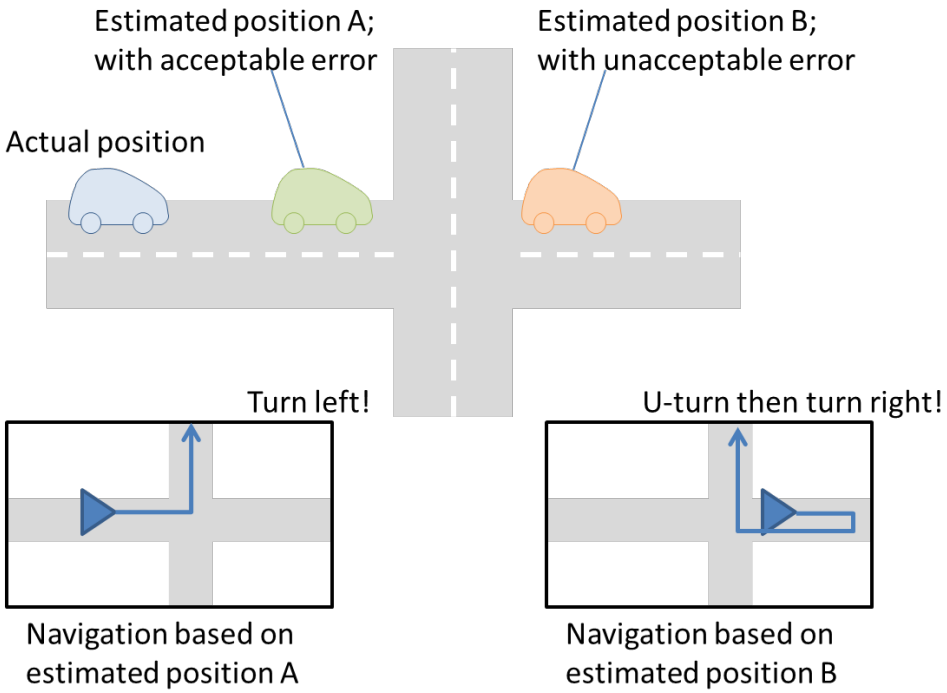

Figure 9.1.1-2: ROAD

To deliver content to vehicles located in a dedicated road, the "ROAD level" accuracy of geo-location would be required. In this case, navigation systems or services can be used and the acceptable error range should be less than a few meters. Some use cases provide route navigation for vehicles based on their location. Figure 9.1.1-2 illustrates the reason why "ROAD" accuracy of geo-location is required. As shown in the figure, the service cannot provide accurate route navigation when the location contains larger error. To obtain the location information with the required accuracy of geo-location level, GPS (Global Positioning System) or Assisted GPS are applicable. Furthermore, when the underlying network consists of RSUs, if the RSUs are located every a few meters to dozens of meters, the oneM2M platform could obtain ROAD level accuracy of geo-location.

To deliver content to vehicles located in a dedicated lane, the "LANE level" accuracy of geo-location would be required. At this time, no use case needs this level of accuracy, but future use cases will require this accuracy. In this case, an acceptable error range should be less than dozens of centimetres. To obtain the location information with the required accuracy level, a device in a vehicle and an infrastructure should support the current positioning systems such as RTK-GPS (Real Time Kinematic GPS) or the latest GNSS (Global Navigation Satellite System) /RNSS (Radio Navigation Satellite System).

9.1.2 Latency

Since a node in vehicular domain moves fast and dynamically, the latency (e. g., network latency) causes a difference between the actual vehicle position and the assumed position in application implemented in the infrastructure. For instance, one second of latency causes dozens of meters of difference. Therefore, if the service requires higher position accuracy and time-sensitive operation, the service should take the effect of latency into consideration (for instance, extrapolating the location considering the latency).

9.2 Key Issue 2: Maintaining AE contact information

Clauses 6.3, 6.5, 6.8 and 6.16, as well as requirement 1 in clause require that a moving M2M device uses the remote services offered by an M2M platform as well as the local services offered by a local M2M gateway. As a device AE moves along its path, it moves out of communication range of one MN-CSE and enters the communication range of another MN-CSE.

Assuming a Release 2 implementation, if the AE wishes to use the services provided by these MN-CSEs, the AE will perform a re-registration. At each (re)registration, a new <AE> resource is created at the new MN-CSE.

AEs are reached through their registrar CSE, by contacting the <AE> resource created during registration. This is referred to as the AE contact information. A number of oneM2M services rely on having up-to-date AE contact information in order to function properly. For these services, the remote CSE providing this service stores the URI of the <AE> resource as an attribute. A quick search of oneM2M TS-0004 [i.16] reveals that an <AE> resource URI can be used in the following services: notification list of subscribe/notify, link associated with an announced resource, notification list of a non-blocking asynchronous request, and member list of a group resource.

This results in the following issue:

- If an AE changes its registration point, and by consequence its AE contact information, the oneM2M services highlighted above, will have URIs that point to the wrong Registrar CSE - that is they have stale AE contact information.

9.3 Key Issue 3: Registration management

Clauses 6.3, 6.5, 6.8 and 6.16, as well as requirement 1 in clause 6.16.10 require that a moving M2M device uses the remote services offered by an M2M platform as well as the local services offered by a local M2M gateway. As a device AE moves along its path, it moves out of communication range of one MN-CSE and enters the communication range of another MN-CSE.

Assuming a Release 2 implementation, if the AE wishes to use the services provided by these MN-CSEs, the AE is required to perform a re-registration to every new MN-CSE it encounters. However, as the AE moves out of communication range of an 'old' MN-CSE, it will not be able to delete its registration hosted on that CSE.

This results in the following potential issue:

- The AE might end up with multiple registrations in the oneM2M system. Although the old registration will eventually expire and be removed, there is a period where both registrations are active. In fact, the duration of this period may be quite long depending on the expiration time associated with the registration.
- During this period, the old MN-CSE may consume processing and storage resources to manage the old AE registration. For example, the old MN-CSE may be monitoring a <timeSeries> resource associated with the AE, in order to track, detect and report missing data in a Time Series. Or, the old MN-CSE may receive a discovery request and it will search under the old registration <AE> resource, for resources matching the filter criteria.

9.4 Key Issue 4: Security

Due to the dynamic nature of vehicles, any service that intends to run in the vehicle domain will have to contend with security within the physical constraints of the vehicle as well as security when attempting communication with other vehicles and with external services. As has been shown throughout the use cases, there is a lot of sensitive data within vehicles to be secured. There are also a great number of use cases that require using this sensitive data for decision making to be transmitted in a secure manner. Furthermore, many decisions (or late decisions) by the vehicle components may impact human safety, so security and reactivity of the entire data acquisition, processing and decision chain is especially critical.

In oneM2M Release 2, Security CSF (SEC CSF) comprises the following functionalities:

- Sensitive data handling.
- Security administration.
- Security association establishment.
- Access control including identification, authentication and authorization.
- Identity management.

Within security, we believe that there has been one particular area that has not been clearly defined, which we label secure communication.

9.4.1 Secure communication

The vehicular domain has been traditionally the most rapidly networked domain. With the introduction of more powerful Electronic Control Units (ECUs) and a number of high bandwidth networking technologies, such as MOST, Automotive Ethernet and LTE-A, it has become imperative that vehicle domain communications be as secure as possible. Security for the vehicle domain is realized without impacting vehicle functionality or imposing taxing resource acquisition and allocation.

In the M2M System functional architecture, there exists the scopes of Intra-M2M Service Provider (SP) and Inter-M2M SP communication. We propose creating a similar classification in order to define secure communication in the vehicle domain. The classifications we recommend are described in Table 9.4-1.

Table 9.4-1 Communication classifications

Classification name	Description
External communication (EC)	Communication between a vehicle and broader network operator related services. This includes, but is not limited to Internet based services. These services may or may not be included as a part of M2M systems
Inter-vehicle communication (InterVC)	Communication between two or many vehicles. These vehicles may or may not be included in a M2M system
Intra-vehicle communication (IntraVC)	Communication within the physical boundaries of one vehicle in the M2M system. This classification extends to wireless based communication that has a source and destination within the same vehicle

It is quickly apparent that the three classifications we have described have different requirements. We also propose that there are two major categories within these classifications. These are detailed in Table 9.4-2.

Table 9.4-2 Classification categories

Classification category	Description
Software based secure communication (SBSC)	Secure communication created by using software running on any type of general purpose CPU in the system
Hardware based secure communication (HBSC)	Secure communication created by using specialized hardware based functions available to the system

There is a need to consider SBSC from a standards point of view, as there are stricter resource requirements in the InterVC and IntraVC classifications. We assume that that EC will be using much more flexible and powerful vehicle gateway-like devices which will lean towards the use of HBSC, meaning that already standardized algorithms can be used. Since in InterVC and IntraVC, the devices used vary greatly in available resources and functionalities, we propose to include SBSC that can be further classified by the criteria seen in Table 9.4-3.

Table 9.4-3 SBSC recommended criteria

Criteria	Description
Cryptographic method type	Symmetric or asymmetric cryptographic method
Program size	The non-volatile memory size of the program securing the communication (typically encrypting/decrypting messages)
Average CPU time per byte	The amount of time a general purpose CPU would need to successfully encrypt/decrypt a byte of a message
Memory use	The size of the volatile memory footprint for the method to successfully encrypt/decrypt a message

Whenever possible, the selection of a particular encryption algorithm, be it SBSC or HBSC based, for communication is realized by the security mechanisms defined in the oneM2M release. When dealing with systems that are not part of the M2M system, the algorithm to use must be selected using other standardized security mechanisms available to both parties.

9.4.2 Lightweight Encryption

This type of encryption, discussed in Recommendation ITU-T X.1362 [i.5], significantly reduces the time consumed by a general purpose CPU for encryption, while permitting the system to pick a suitable level of protection for data confidentiality and integrity. Particularly for SBSC communications, these types of light-weight encryption algorithms shall be considered for resource constrained devices.

9.4.3 Security for credential

In almost all of the use cases described in clause 6, it is necessary to authenticate between entities before a service starts. In vehicle domain, there are many use cases that impact safety (particularly use cases at clauses 6.2, 6.8 and 6.11), so that a trusted credential should be used for authentication. shows examples of authentication between infrastructure node and communication module. If a credential in communication module is compromised, malicious node can connect with communication module. After establishment of connection, malicious node may be able to receive vehicle information such as version of installed software and may be able to send malicious message.

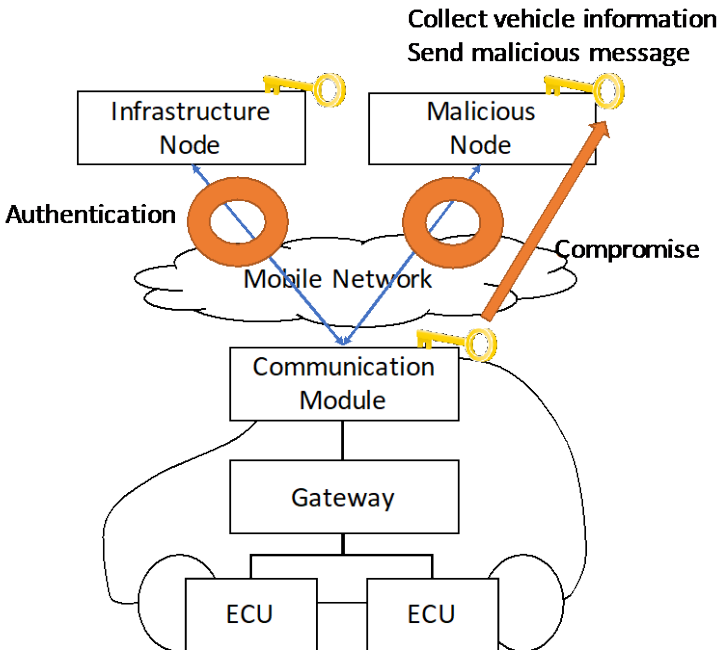

Figure 9.4.3-1: Example of authentication between infrastructure node and communication module

Figure 9.4.3-2 shows an example of authentication between gateway and ECU. A credential is used for authentication between gateway and ECU. If the credential in gateway or ECU is compromised, unauthorized ECU can be installed.

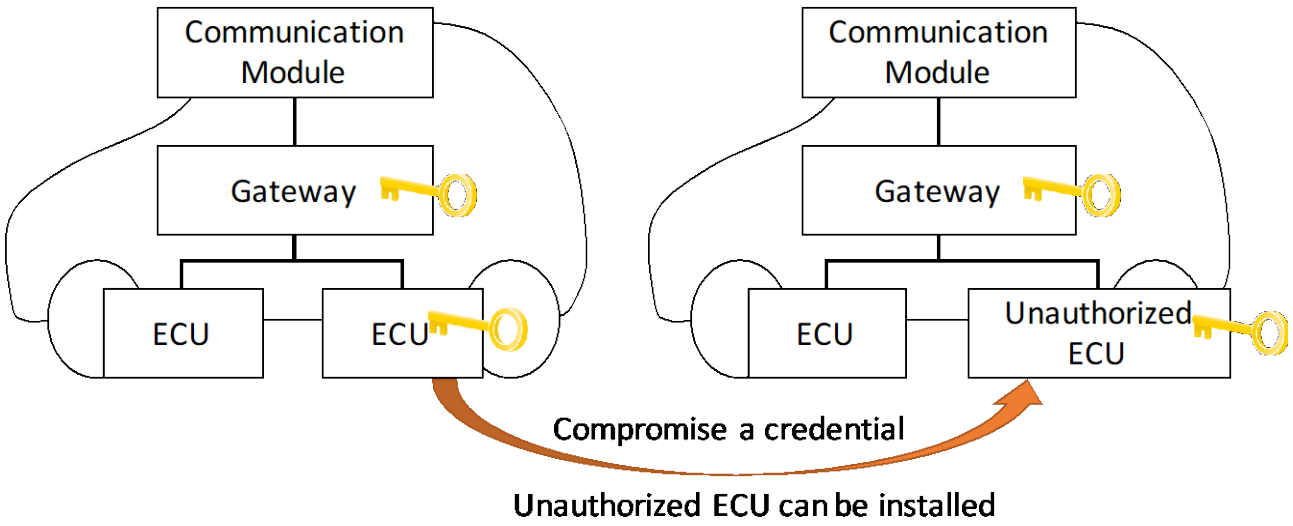

Figure 9.4.3-2: Example of authentication between Gateway and ECU

In addition to authentication, some data such as ECU software/middleware/firmware also impact the in-vehicle system, so the data should be verified for integrity using a trusted credential. Figure 9.4.3-3 shows an example of integrity check of software/middleware/firmware on gateway. If boot symmetric key and boot digest are compromised, unauthorized software/middleware/firmware can be installed.

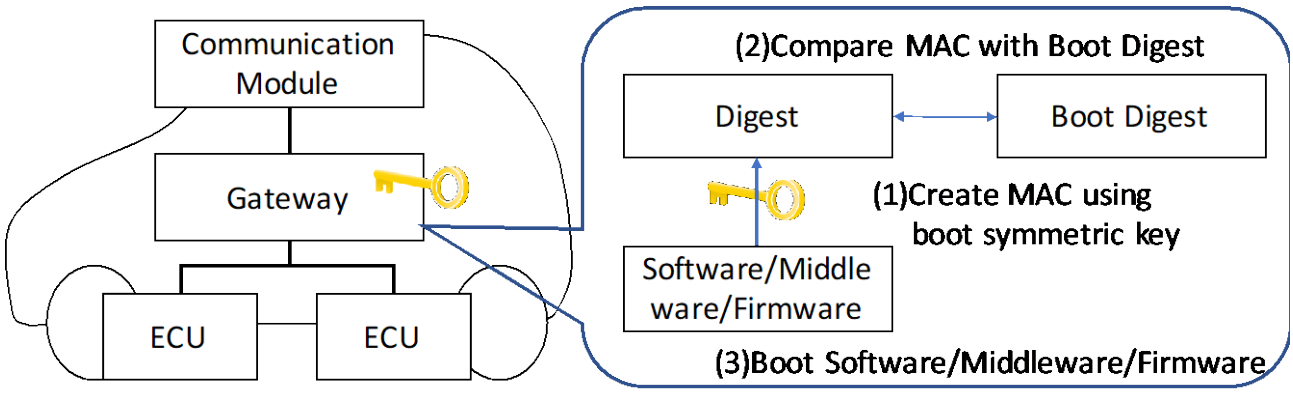

Figure 9.4.3-3: Example of integrity check of software/middleware/firmware on gateway

Figure 9.4.3-3 shows an example of verification of ECU integrity. ECU creates digests of software/middleware/firmware and digital signatures of each digests using ECU private key, then ECU send the digests and signatures to gateway. Gateway verifies the digital signatures using ECU public key. If ECU private key is compromised, unauthorized ECU can be installed.

9.5 Key Issue 5: Cross-Resource Subscription

Clause requires that oneM2M System shall enable subscriptions to changes to multiple resources which aim to generate notifications if and only if the expected changes to those resources occur concurrently.

Assuming the scenario in clause 6.7 where concurrent changes to several resources need to be identified for a single event detection. A Release 2 implementation would require the subscriber to create several subscriptions. Both the host and the subscriber will process and monitor a potentially high number of individual notifications, although the concurrent change might be very rare. In order to prevent the loss of individual notifications to affect event evaluation, the subscriber would need to implement potentially sophisticated logic without event generation knowledge available only at the host.

This results the following issue:

- Subscriptions to changes to multiple resources using separate subscriptions require a messaging overhead in processing subscription requests by the host and processing un-actionable notifications by the subscriber.
- In addition transmission issues due to vehicle mobility which affect individual notifications create undue burdens on the subscriber/receiver in order to evaluate the occurrence of the event of interest.

9.6 Key Issue 6: Subscription Aggregation

Clause 6.7 describes two requirements related to data subscription grouping or aggregation:

1) the oneM2M System shall enable M2M Gateways to group/aggregate data subscriptions to reduce the number of messages to M2M Devices; and
2) the oneM2M System shall enable M2M Gateways to distribute notifications according to how data subscriptions have been grouped/aggregated.

Assuming the scenario in clause 6.7 where several remote subscribers create identical subscriptions to resources on an end M2M Device (e.g. sensor) behind an M2M gateway (e.g. the smart vehicle). A Release 2 implementation would require each subscriber to issue separate subscription request to the M2M Device via the M2M Gateway. This produces a messaging overhead between the M2M Gateway and the M2M Device due to the transmission of multiple subscription requests which could be the same. The M2M Device needs to generate independent notifications (e.g. one for each subscriber) when the event notification criteria is satisfied, which will be transmitted via the M2M Gateway separately. In addition, the M2M Device may have memory and processing constraints designed to support a limited number of resources. In this case the burden of monitoring each subscription separately, creates a processing overhead proportional with the number of subscribers. For some constrained devices, it is not feasible to design based on service demands (i.e. how many entities are interested in subscribing) rather than on the resources supported.

This results in the following issue:

- Handling of multiple subscription requests with the same criteria introduces request and notification messaging overhead which is duplicated for M2M Devices behind an M2M Gateway.

9.7 Key Issue 7: Time synchronization

Clause 6.17 provides a requirement related to time synchronization:

1) The oneM2M System shall be able to verify time synchronization among multiple sources that provide inputs for a single output.
2) Other use cases such as the one in clause 6.18 rely on strict timing requirements being able to be enforced in the system.

Due to their resource constrained nature, some devices and networks lack support for a reference time or clock that can be efficiently and effectively propagated and synchronized to all the entities, in an end-to-end manner. Time synchronization mechanisms such as NTP, PTP, and GPS are not always available and /or best-suited for all types of IoT deployments.

Without adequate time synchronization at the device level, applications hosted on different devices are unable to maintain synchronization with applications hosted on other devices in the network. For example, a lack of synchronization between the applications can result in one application not sending a message to another application in the proper scheduled time window or waking up at the proper time in order to receive a message. In another example, applications timestamping data (e.g. sensor measurements) which they share with other applications may provide information which is incorrect or not being able to be interpreted correctly.

In release 3 implementations the existence or lack of synchronization mechanisms cannot be detected. Moreover, lack of synchronization between various entities cannot be detected and corrected.

This results in the following issue:

- Services requiring time-sensitive operations rely on the various Service Layer instances being time-synchronized.

10 Potential Solutions for the Key Issues

10.1 Solution A: Maintaining AE contact information - IN-CSE Notifies all CSEs

10.1.1 Solution Description

In order to address the Key Issues 2 and 3 we can rely on the IN-CSE to propagate the updates throughout the oneM2M system and to proactively delete the AE registration in the old MN-CSE. The basic procedure is described below:

Step 1: The IN-CSE determines that an AE mobility event has occurred. An AE mobility event occurs when an AE has moved and has re-registered to a new ASN-CSE or MN-CSE.

For cases where the AE-ID is assigned by the IN-CSE (AE-ID-Stem starts with the letter 'S') the IN-CSE may examine the re-registration <AE> announcement. If this announcement is using a previously allocated AE-ID, but coming from a different ASN-CSE or MN-CSE, the IN-CSE can make the determination that this registration is as a result of an AE mobility event.

For cases where the AE-ID is assigned by a local CSE (an ASN-CSE or MN-CSE), then the AE-ID-Stem starts with the letter 'C'. The local CSE may examine the registration request. If the AE supplies an AE-ID-Stem and the local CSE is unaware of this AE, then the CSE can make the determination that this registration is as a result of an AE mobility event. Once this determination is made, the local CSE needs to inform the IN-CSE. This can be achieved through a Notify request, which includes in its *Content* the old AE contact information (old URI) and new AE contact information (new URI).

Step 2: The IN-CSE examines all its services that have the old AE contact information, and it updates these with the new AE contact information. Basically this would involve looking in the attributes of all <group>, <subscription>, and <announce> resources, and replacing the old URI with the new URI.

Step 3: The IN-CSE informs MN-CSEs and ASN-CSEs that there has been an AE mobility event for this AE. This can be achieved through a Notify request, which includes in its *Content* the old AE contact information (old URI) and new AE contact information (new URI).

Step 4: The MN-CSEs and ASN-CSEs examine their services to determine if they are impacted by the AE mobility event. If so, they update the old AE contact information with the new AE contact information. The Old MN-CSE will use this as an indication that it may delete the old <AE> registration.

10.1.2 Solution Applicability

This solution applies to Key Issue 2 and Key Issue 3.

10.2 Solution B: Maintaining AE contact information - IN-CSE Notifies only impacted CSEs

10.2.1 Solution Description

This solution addresses Key Issue 2, and avoids the need for the IN-CSE to contact all ASN/MN-CSEs in the oneM2M System, to inform them about an AE mobility event. The solution relies on the IN-CSE being aware of the ASN/MN-CSEs that have oneM2M services that use AE contact information (namely notification list of subscribe/notify, link associated with an announced resource, notification list of a non-blocking asynchronous request, and member list of a group resource). The IN-CSE keeps a master list of AE contact information for each ASN/MN-CSE in the oneM2M System.

The operation involves two processes:

1) Keeping the IN-CSE in sync with all ASN/MN-CSEs that have resources that contain an <AE> resource URI.

2) Once an AE mobility event is detected, the IN-CSE determining the impacted ASN/MN-CSEs, and only updating these with the new AE contact information.

Process 1: Keeping IN-CSE in sync

Step 1: IN-CSE creates a resource to store AE contact information for each ASN/MN-CSE. This may be a simple list of <AE> resource URIs associated with each ASN/MN-CSE. For example, these may be stored in a <*masterAEContactList*> resource.

Step 2: When a service using AE contact information is created/updated/deleted at an ASN/MN-CSE, the ASN/MN-CSE notifies the IN-CSE. The notification includes the AE contact information.

Step 3: Upon reception of the notification, the IN-CSE updates the <*masterAEContactList*> accordingly. Either adding or deleting the entry for the impacted CSE.

Process 2: Updating AE contact information in oneM2M system

Step 1: The IN-CSE determines that an AE mobility event has occurred. An AE mobility event occurs when an AE has moved and has re-registered to a new ASN-CSE or MN-CSE.

- For cases where the AE-ID is assigned by the IN-CSE (AE-ID-Stem starts with the letter 'S') the IN-CSE may examine the re-registration <AE> announcement. If this announcement is using a previously allocated AE-ID, but coming from a different ASN-CSE or MN-CSE, the IN-CSE can make the determination that this registration is as a result of an AE mobility event.
- For cases where the AE-ID is assigned by a local CSE (an ASN-CSE or MN-CSE), then the AE-ID-Stem starts with the letter 'C'. The local CSE may examine the registration request. If the AE supplies an AE-ID-Stem and the local CSE is unaware of this AE, then the CSE can make the determination that this registration is as a result of an AE mobility event. Once this determination is made, the local CSE needs to inform the IN-CSE. This can be achieved through a Notify request, which includes in its *Content* the old AE contact information (old URI) and new AE contact information (new URI).

Step 2: The IN-CSE examines all its services that have the old AE contact information, and it updates these with the new AE contact information. Basically this would involve looking in the attributes of all <group>, <subscription>, and <announce> resources, and replacing the old URI with the new URI.

Step 3: The IN-CSE searches in the <*masterAEContactList*> to find all CSEs that are impacted as a result of the AE mobility event. It then informs these MN-CSEs and ASN-CSEs that there has been an AE mobility event for this AE. This can be achieved through a Notify request, which includes in its *Content* the old AE contact information (old URI) and new AE contact information (new URI).

Step 4: The MN-CSEs and ASN-CSEs examine their services to determine if they are impacted by the AE mobility event. If so, they update the old AE contact information with the new AE contact information. The Old MN-CSE will use this as an indication that it may delete the old <AE> registration.

10.2.2 Solution Applicability

This solution applies to Key Issue 2.

10.2.3 Solution Details

10.2.3.1 Impacted Resources and Attributes

10.2.3.1.1 Overview

To implement this solution, the <*AE*> resource and the <*AEAnnc*> resource need to be modified, and two new resource types need to be defined (<*AEContactList*> and <*AEContactListPerCSE*). The <*AEContactList*> resource maintains a list of CSEs that may be impacted by a change in an AE registration point. For each CSE in this list, the <*AEContactListPerCSE*> resource includes a list of all "references" to Application Entity resource identifiers that are included in this CSE. These "references" to Application Entity resource identifiers may occur in announcement links, notification targets, and in group member IDs. These are further detailed in the following clauses.

10.2.3.1.2 Modified <*AE*> resource

New attributes as shown in Table 10.2-1. The *status* attribute allows to transition an <*AE*> resource to an INACTIVE status. While INACTIVE, the resource is not discoverable. In addition as part of the registration procedure, an Application Entity may request (through the localRegistration attribute) that the service provider track the AE as it changes registration points. If set to TRUE, the service provider will track the Application Entity, essentially keeping track of its Registrar CSE. If set to FALSE, the service provider will not track the Application Entity.

Table 10.2-1: New attributes of <*AE*> resource

Attributes of <*AE*>	Multiplicity	RW/ RO/ WO	Description	<*AEAnnc*> Attributes
status	0..1	RW	Denotes status of the AE registration. If ACTIVE, the <AE> resource and all its child resources may be discoverable. If INACTIVE, the <AE> resource and all its child resources shall not be discoverable.	OA
localRegistration	0..1	RW	Denotes if the Application Entity requests that its Registration Points be tracked. If TRUE, AE requests to be tracked as it changes its Registration Point. If FALSE, the AE requests not to be tracked as it changes its Registration Point.	OA

10.2.3.1.3 Modified <*AEAnnc*> resource

New common attribute to allow transitioning the <*AEAnnc*> resource to INACTIVE status. While INACTIVE, the <*AEAnnc*> resource at the IN-CSE is not discoverable (see Table 10.2-2).

Table 10.2-2: New Commonly Used Attributes for Announced Resources

Attribute Name	Mandatory/Optional	Description
status	Optional	Only applicable to announced <AE> resource. Denotes status of the announced AE registration. If ACTIVE, the announced <AE> resource and all its child resources may be discoverable. If INACTIVE, the announced <AE> registration and all its child resources shall not be discoverable.

10.2.3.1.4 New Resource Type: AEContactList

An <*AEContactList*> resource shall contain <*AEContactListPerCSE*> child resources, one for each CSE that has sent a NOTIFY request to the CSE about the creation, update, or deletion of a resource that references an Application Entity resource identifier. The <*AEContactList*> resource shall only be created in the IN-CSE.

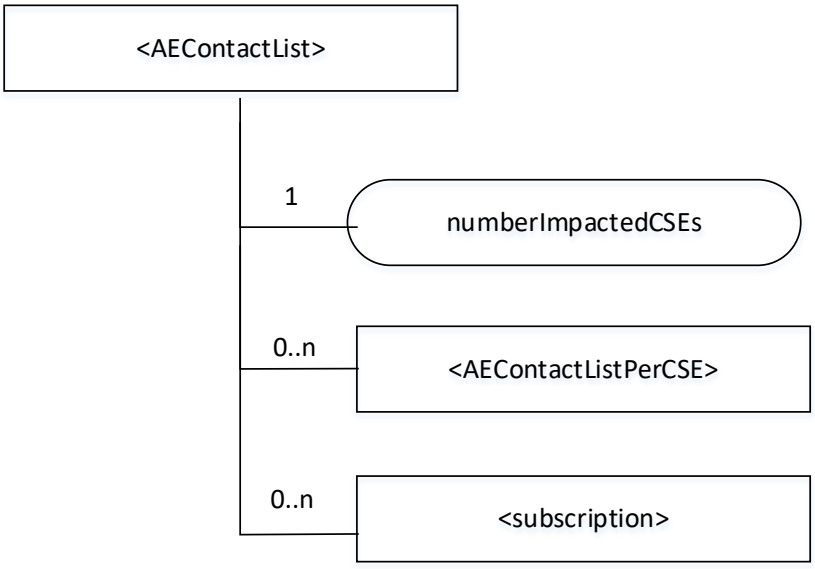

Figure 10.2.3-1: Structure of <*AEContactList*> resource

The <*AEContactList*> resource shall contain the child resources specified in Table 10.2-3.

Table 10.2-3: Child resources of <*AEContactList*> resource

Child Resources of <*AEContactList*>	Child Resource Type	Multiplicity	Description
[variable]	<*subscription*>	0..n	
[variable]	<*AEContactListPerCSE*>	0..n	

The <*AEContactList*> resource shall contain the attributes specified in Table 10.2-4 .

Table 10.2-4: Attributes of <*AEContactList*> resource

Attributes of < *AEContactList* >	Multiplicity	RW/ RO/ WO	Description
resourceType	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
resourceID	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
resourceName	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
parentID	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
expirationTime	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
accessControlPolicyIDs	0..1 (L)	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
creationTime	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
lastModifiedTime	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
Labels	0..1 (L)	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
numberImpactedCSEs	1	RO	The number of Hosting CSEs that have reported that they have a reference to an Application Entity resource identifier.

10.2.3.1.5 New Resource Type: AEContactListPerCSE

An <*AEContactListPerCSE*> resource shall include a list Application Entity resource identifiers (SP-relative-Resource-IDs of an AE). For example, these Application Entity resource identifiers may occur in announcement links, notification targets, and in group member IDs. The <*AEContactListPerCSE*> resource shall only be created in the IN-CSE.

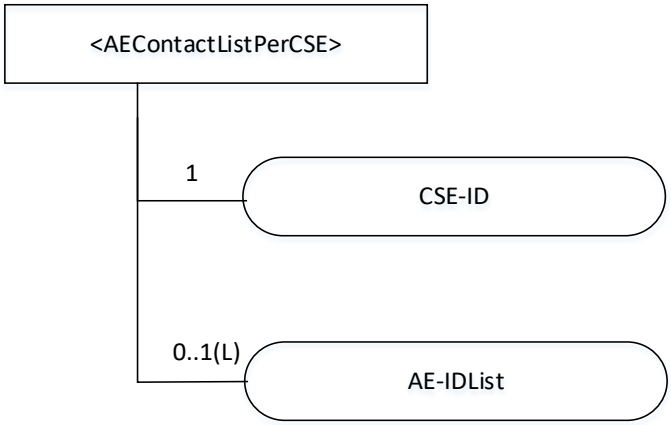

Figure 10.2.3-2 Structure of *<AEContactListPerCSE>* resource

The <*AEContactListPerCSE*> resource shall contain the attributes specified in Table 10.2-5.

Table 10.2-5: Attributes of *<AEContactListPerCSE>* resource

Attributes of *<AContactListPerCSE>*	Multiplicity	RW/ RO/ WO	Description
resourceType	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
resourceID	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
resourceName	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
parentID	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
expirationTime	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
accessControlPolicyIDs	0..1 (L)	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
creationTime	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
lastModifiedTime	1	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
Labels	0..1 (L)	RO	See clause 9.6.1.3 in oneM2M TS-0001 [i.3].
CSE-ID		RO	The identifier of the Hosting CSE which has a reference to an Application Entity resource identifier (SP-relative-Resource-ID that points to an AE). Hosting CSEs notify the IN-CSE when they have a reference to an <AE> resource through e.g. announcements, notification targets, group member IDs.
AE-IDList	0..1(L)	RO	List of Application Entity resource identifiers hosted on CSE with identifier CSE-ID.

10.2.3.2 Impacted Information Flows

10.2.3.2.1 Overview

To implement this solution, the new procedure to manage change in AE Registration Point is detailed in the following clause

Other oneM2M TS-0001 [i.3] information flows/procedures will be affected as follows:

- Application Entity registration procedure (oneM2M TS-0001 [i.3], clause 10.1.1.2.2): The existing 5 cases are shown to be valid for initial registration or re-registration to the same Registrar CSE. Two new cases are added, for re-registration to a new Registrar CSE: when the AE-ID-Stem starts with 'S' and AE includes an AE-ID-Stem and when AE-ID-Stem starts with 'C' and AE includes an AE-ID-Stem

- The notification procedures overview (oneM2M TS-0001[i.3], clause 10.3.1) update includes in the listing the new notifications of registration point changes and notifications to the IN-CSE for new/updated reference to an Application Entity Resource identifier

- Modifications to the basic CREATE, UPDATE and DELETE procedures (oneM2M TS-0001[i.3], clause 10.1) to show that notifications are sent by Receivers when the operation changes a reference to an Application Entity Resource Identifier, These notifications allow adding, updating, or removing entries from the <*AEContactList*> resource.

- Modification of CSE processing at expiry of resource *expirationTime* (oneM2M TS-0001[i.3], clause 9.6.1.3.1) to indicate that if the 'obsolete' resource had a reference to an Application Entity Resource ID, the Hosting CSE shall send a NOTIFY request to the IN-CSE, allowing removing the entry from the <*AEContactList*> resource.

10.2.3.2.2 Procedure for Managing Change in AE Registration Point

10.2.3.2.2.1 Procedure at IN-CSE

The IN-CSE may determine that an AE has changed registration point either by:

- Observing the creation on an <*AEAnnc*> resource with an **AE-ID-Stem** that it had previously assigned for a different Registrar CSE

- Receiving a NOTIFY request from a Registrar CSE whose content includes the SP-relative-Resource-ID before and after the change in registration point

In both cases, the IN-CSE shall send a NOTIFY request to the CSEs, so that these may update the references to the <*AE*> resources for the AE that has changed its registration point. If the IN-CSE maintains an <*AEContactList*> resource, the IN-CSE shall determine which CSEs are effected, and shall send the NOTIFY request only to these. If the IN-CSE does not maintain an <*AEContactList*> resource, the IN-CSE shall send the NOTIFY request to all CSEs. The ***Content*** parameter of the NOTIFY request shall contain the SP-relative-Resource-ID at the prior registration point and the SP-relative-Resource-ID at the new registration point

10.2.3.2.2.2 Procedure at any CSE

Upon receiving a NOTIFY request regarding a change in AE registration point, the receiving CSE shall update all references to the SP-Relative-Resource-ID (e.g. in Announce links, Notification targets, group Member IDs) tied to the prior AE registration point, so that these refer to the new AE registration point.

10.3 Solution C: Cross-Resource Subscription

10.3.1 Solution Description

In order to address the Key Issue 5, a subscriber AE/CSE creates resource subscriptions where automatic notifications depend on two or more resources, not a single resource. In other words, notifications are generated when changes to multiple resources occur concurrently. This subscription type (termed here Cross-Resource Subscription) involves multiple resources which do not necessarily have parent-child relationship. The corresponding notifications are termed Cross-Resource Notifications.

The subscriber indicates specific requirements when issuing a Cross-Resource Subscription request to the resource host. The basic procedure is described below:

Step 1: The subscriber sends a Cross-Resource Subscription request to the resource host. This request message informs the resource host that the subscriber is interested in concurrent changes of multiple resources and expects to receive a single notification if all changes take place. For this purpose, the message may include parameters such as:

- target resources (targetResourcesList): the list of target resources for the Cross-Resource subscription.

- event notification criteria (eventNotificationCriteriaList): the list of event notification criteria (as defined in oneM2M Release 2) for all target resources. One event notification criteria could apply to each individual resource in listOfTargetResources.

- time window for evaluation: Includes information about window type (windowType), e.g. periodic, sliding, etc. and a window duration (windowDuration). The time window is used by the resource host to determine if the expected changes to the target resources should generate Cross-Resource Notifications.

Step 2: The resource host processes the received Cross-Resource Subscription request. If the request is approved, the resource host creates a local subscription resource to maintain the subscription request.

Step 3: The resource host sends a response to the subscriber. The response may include a Uniform Resource Identifier (URI) of the local subscription resource created in Step 2.

Step 4: The resource host observes that events on target resources take place.

Step 5: The resource host uses the time window mechanism to determine whether a notification should be generated. The time window mechanism implemented in the resource host enables the host to relieve the subscriber of the processing burden which would be required if using existing single-resource subscription mechanisms in oneM2M Release 2, as discussed in clause 9.5.

Step 6: The resource host sends a notification to the subscriber (and/or other entities if specified by the notificationURIs in Step 1) assuming the decision from Step 5 is yes.

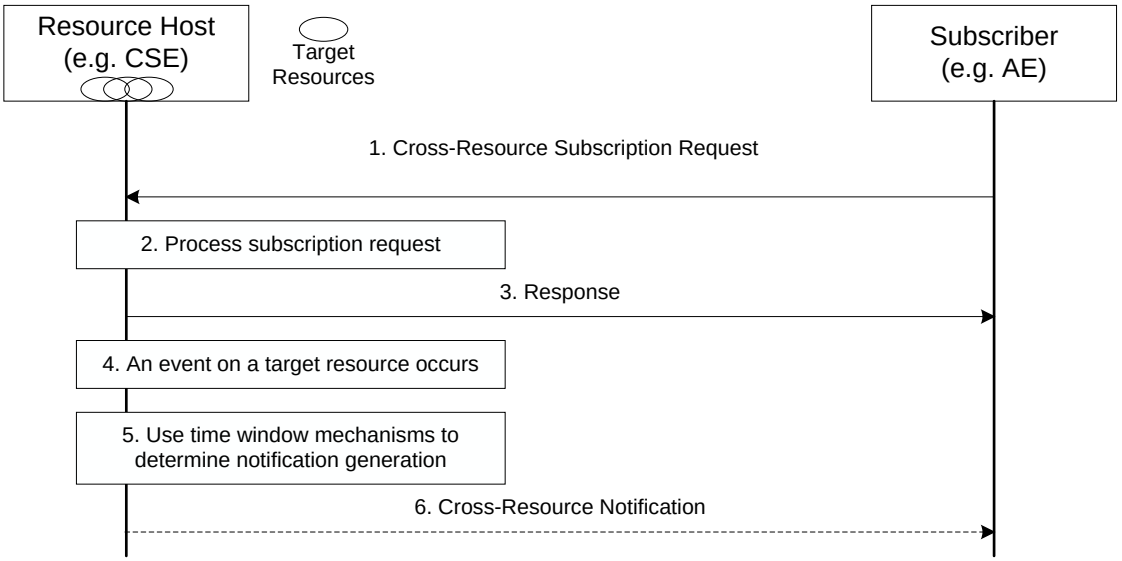

Figure 10.3.1-1. Solution Procedures for Cross-Resource Subscription

As illustrated in Figure 10.3.1-1 , the proposed solution procedures use Cross-Resource Subscription and Cross-Resource Notification, which removes the messaging overhead in processing separate subscription requests by the Resource Host and processing un-actionable notifications by the Subscriber.

Also, in the proposed solution procedures in Figure 10.3.1-1 , the Resource Host is responsible for evaluating the occurrence of the event of interest , which removes the undue burdens at the Subscriber.

10.3.2 Solution Applicability

This solution applies to Key Issue 5.

10.3.3 New Resources and Procedures

10.3.3.1 Introduction

A new resource <*crossResourceSubscription*> is proposed to implement Cross-Resource Subscription functionality, together with the existing <group> resource.

For example, assume an IN-CSE has a <*group*> resource (e.g. <*INCSEBase*>/<*group*>) which has two members <*container1*> and <*container2*>. If an IN-AE wants to receive automatic notification when the value of both contain resources exceed a threshold, it shall create a cross-resource subscription (i.e. the new resource type <*crossResourceSubscription*>) as the child resource of this <*group*> resource and set appropriate values for the attributes of <*crossResourceSubscription*> using the following CREATE command. Then IN-CSE shall issue a notification if and only if the value of both container resources change to exceed the threshold within the timeWindowSize (i.e. 60 seconds):

- CREATE <inCSEBase>/<group>/<crosssResourceSubscription>; payload: timeWindowType=1 (i.e. periodical time window), timeWindowSize=60 seconds.
- Here, timeWindowType, and timeWindowSize are the attributes for a <crossResourceSubscription> resources and they will be described in the clause 10.3.3.2.

The following clauses introduce new attributes for <*subscription*> resource supporting Cross-Resource Subscription functionality, detailed procedure to create a cross-resource subscription as a child resource of a <*group*> resource and the corresponding notification processing.

10.3.3.2 New <crossResourceSubscription> Resource to Enable Cross-Resource Subscription Functionality

- <*crossResourceSubscription*> extends existing <*subscription*> resource with the following enhancements.
- <*crossResourceSubscription*> has the same child resources as <*subscription*> has.
- <*crossResourceSubscription*> replaces "*eventNotifionationCriteria*" attribute of <*subscription*> with a new attribute "*listOfEventNotificationCriteria*". For other attributes of <*subscription*>, <*crossResourceSubscription*> will have the same.
- <*crossResourceSubscription*> has several new attributes as included in Table 10.3-1.

Table 10.3-1 lists new attributes for the <*crossResourceSubscription*> resource. Those attributes are leveraged to trigger and create a cross-resource subscription.

Table 10.3-1: New Attributes of <*crossResourceSubscription*> Resource

Attributes of <*subscription*>	Multiplicity	RW/ RO/ WO	Description
listOfEventNotification Criteria	1 (L)	RW	This attribute lists *eventNotificationCriteria* for each target resource involved in a cross-resource subscription.
numOfTargetResource sForNotification	0..1	RW	This attribute indicates the required number of target resources for generating a cross-resource notification.
timeWindowType	1	RW	This attribute indicates the type of time window mechanisms (e.g. "*timeWindowType*=1" stands for periodical time window and "*timeWindowType=2*" represents sliding time window) which will be used to determine the generation of a cross-resource notification.
timeWindowSize	1	RW	This attribute indicates the size or time duration (e.g. in seconds) of the time window, based on which cross-resource notification shall be generated. Note that the maximum window size (e.g. 60 seconds) may be enforced by the Hosting CSE for a subscriber. If the *timeWindowSize* indicated or requested by a subscriber is larger than the maximum window size, the Hosting CSE rejects the subscriber's request.

10.3.3.3 Procedure for Creating a Cross-Resource Subscription

This procedure shall be used to request the creation of a new cross-resource subscription *<crossResourceSubscription>* resource as a child resource of a *<group>* resource. As a result, cross-resource notifications shall be generated if and only if changes occur during the designated time window on each involved target resource; note that the condition for each target resource is indicated by the new attribute *listOfEventNotificationCriteria*.

Table 10.3-2: *<crossResourceSubscription>* CREATE over a *<group>* resource

***<crossResourceSubscription>* CREATE**	
Associated Reference Point	Mca, Mcc and Mcc'
Information in Request message	All parameters defined in table 8.1.2-2 in oneM2M TS-0001 [i.3] apply with the specific details for: ***To:*** a *<group>* resource ***Content*:** The resource content shall provide values of attributes of *<crossResourceSubscription>*
Processing at Originator before sending Request	According to clause 10.1.1.1 in oneM2M TS-0001 [i.3] with the following additions: The Request shall include *timeWindowType* and *timeWindowSize.* The Request shall include *listOfEventNotificationCriteria.* The Request shall include *numOfTargetResourcesForNotification.*
Processing at Receiver (i.e. Group Hosting CSE)	According to clause 10.1.1.1 in oneM2M TS-0001 [i.3] with the following additions: • Check if the resource as indicated in the ***To*** parameter in the Request is a *<group>* resource. • Check if the Originator has privileges for creating a child resource in the ***To*** parameter in the Request. • Upon successful validation, obtain the IDs of all member resources from the *membersIDs* attribute of the addressed *<group>* resource. Then, convert the Request to normal single-resource subscription request (i.e. removing *numOfTargetResourcesForNotification*, *timeWindowType*, *timeWindowSize*; extract *eventNotificationCriterial* for each member from *listOfEventNotificationCriteria; change notificationURI to the Group Hosting CSE*) and send the normal single-resource subscription request to the Members Hosting CSEs addressing the obtained IDs appended with the ID of the *<subscription>* resource to be created. • After receiving the responses from the Members Hosting CSEs, respond to the Originator with the aggregated results and the associated *memberIDs*. If any of the checks above fails, the Hosting CSE shall send an unsuccessful response to the Originator with corresponding error information. Otherwise, the Hosting CSE shall create the *<crossResourceSubscription>* resource under the *<group>* resource as indicated in the ***To*** parameter in the Request, and send a successful response to the Originator.
Processing at Member Hosting CSE	For the subscribe procedure, the Members Hosting CSE shall treat the request received from the group Hosting CSE as a normal single-resource SUBSCRIBE request on the addressed member resource as if it comes from the original Originator. Therefore the members Hosting CSE shall: • Check if the original Originator has the READ permission on the members resource • Upon successful validation, perform the subscribe procedures for the corresponding type of member resource as described in clause 10.2.12 in oneM2M TS-0001 [i.3]. • Send the corresponding response to the group Hosting CSE
Information in Response message	All parameters defined in table 8.1.3-1 in oneM2M TS-0001 [i.3] apply with the specific details for: • ***Content*:** address of the created *<crossResourceSubscription>* resource, according to clause 10.1.1.1 in oneM2M TS-0001 [i.3].
Processing at Originator after receiving Response	According to clause 10.1.1.1 in oneM2M TS-0001 [i.3]
Exceptions	According to clause 10.1.1.1 in oneM2M TS-0001 [i.3]

10.3.3.4 Procedure for Generating Cross Resource Notification

After the Group Hosting CSE creates the <*crossResourceSubscription*> as described in the clause 10.3.3.2, it uses the designated time window mechanism to determine if a cross-resource notification shall to be issued each time when receiving a notification from Member Hosting CSEs. Only when expected changes on all target resources occur within the required time window, the Group Hosting CSE issues a notification (i.e. cross resource notification) to the Originator and/or its designated notification receivers; otherwise, the Group Hosting CSE just simply discards the received normal single-resource notification from any Member Hosting CSE.

10.4 Solution D: Subscription Aggregation

10.4.1 Solution Description

In order to address the Key Issue 6, a middle node (e.g. the M2M Gateway in clause 6.7) can group or aggregate subscription requests received from multiple subscribers, generate one aggregated subscription request, and only forward this aggregated subscription request to the resource host (e.g. the M2M Device in clause 6.7). The basic procedure is described below:

Step 1: The resource host may publish/announce its resources and associated event notification criteria to the middle node, which could be the registrar CSE of the resource host. This message contains a list of following parameters:

- ResourceID: the identifier of the source which can be subscribed.
- EventNotifCriteria: the event notification criteria associated with the resource as denoted by resourceID.
- WhiteSubList: the list of subscribers which are allowed to make subscription to the resource as denoted by resourceID.
- BlackSubList: the list of subscribers which are not allowed to make subscription to the resource as denoted by resourceID.
- Access control criteria for allowing or disallowing subscribers. Note that the access control criteria could be based on the location of subscribers, the service or application type of subscribers, etc.

Step 2: The middle node maintains the list of resouceID and its eventNotifCriteria. It sends a response back to the resource host.

Step 3: Each subscriber sends a subscription request to the middle node. Besides resourceID, notifURI, and eventNotifCriteria, this message could optionally contain a new parameter aggrgFlag. Note that the destination of this message is the middle node:

- aggrgFlag: a flag to indicate if the subscriber likes this subscription request to be aggregated (e.g. if aggrgFlag=TRUE) or not (e.g. if aggrgFlag=FALSE).

Step 4: The middle node finds that those subscription requests in Step 3 from multiple subscribers can be aggregated (e.g. they have the same resourceID and eventNotifCriteria; and the subscribers are in the whiteSubList as received in Step 1). Then it aggregates those subscription requests, creates a subscription group SG(i), and generates an aggregated subscription request. The middle node also creates a notification group NG(i), which contains all notifURI received from all subscribers in Step 3.

Step 5: The middle node sends an aggregated subscription request to the resource host. This message may contain the following parameters, which are associated with SG(i). In addition, the middle node maintains the mapping relationship between SG(i) and NG(i):

- ResourceID: the identifier of resource which multiple subscribers are interested.
- EventNotifCriteria: the event notification criteria multiple subscribers indicate.
- NewNotifURI: indicates the address which the resource host should send the notification to (i.e. the address of the middle node or the identifier of SG(i) being created during subscription aggregation in Step 4.
- SubscriberList: the list of original subscribers included in SG(i). This parameter may be optional.

Step 6: The resource host sends a response back to the middle node. If subscriberList is included in Step 5, the resource host may disapprove some subscribers. If that happens, the middle node will update NG(i) accordingly.

Step 7: An event corresponding to eventNotifCriteria in Step 3 occurs.

Step 8: The resource host sends a notification to newNotifURI which was indicated in Step 5.

Step 9: The middle node receives the notification and distributes it to all subscribers and their notification targets as indicated in Step 3 and captured in NG(i).

Step 10: Subscribers and their notification targets send a response to the middle node.

Step 11: The middle node sends a response back to the resource host.

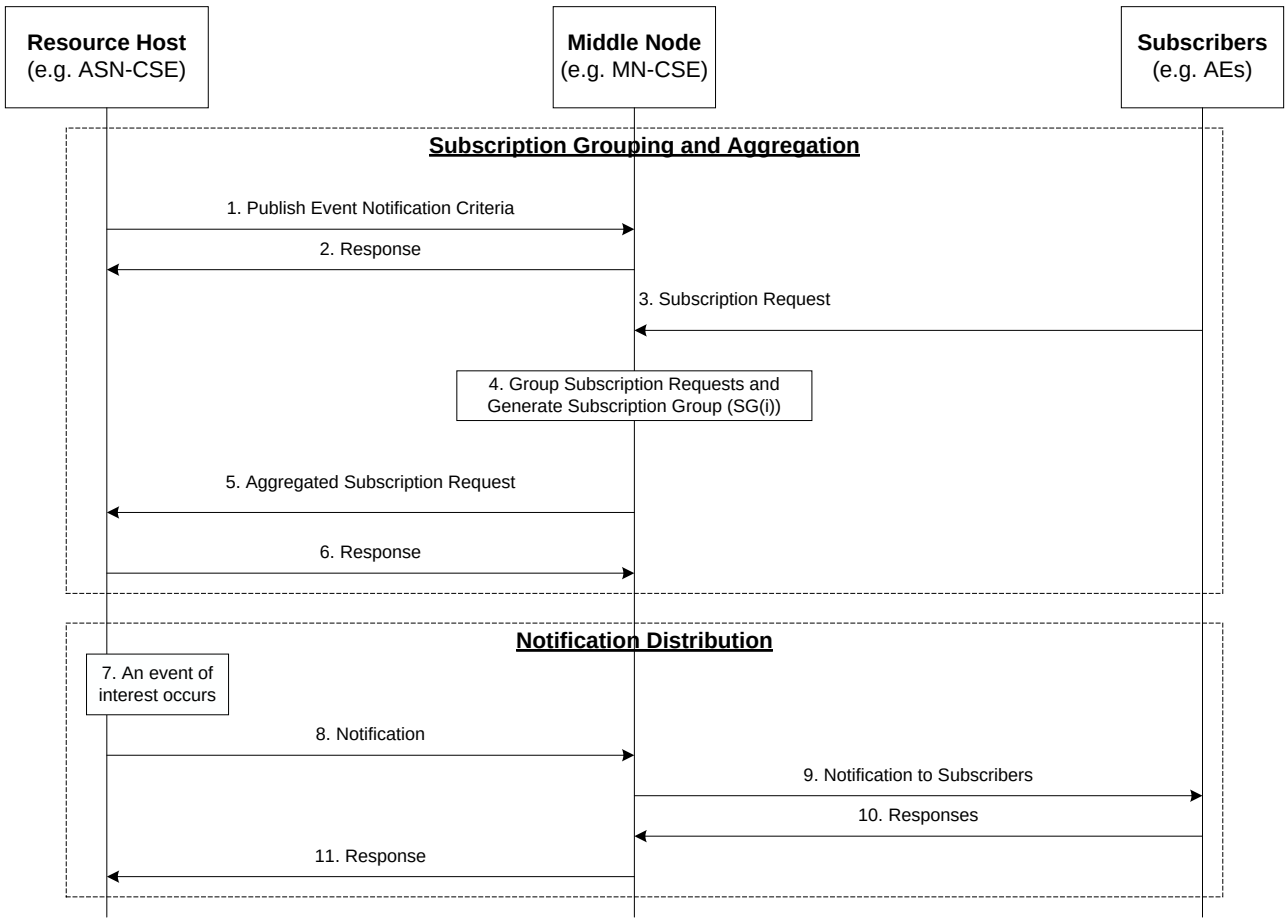

Figure 10.4.1-1: Solution Procedures for Subscription Aggregation

10.4.2 Solution Applicability

This solution applies to Key Issue 6.

10.5 Solution E: Secure Channel Establishment

10.5.1 External communication and inter-vehicle communication

10.5.1.1 Solution Description

In order to address key issue 4: security, vehicular security requires secure communication and lightweight cryptography to protect data confidentiality, data integrity and availability. In general, assuming support of an Internet

Protocol layer, Transport Layer Security (TLS) [i.6] is used to establish secure channel between entities and also supports data integrity in transport layer.

According to clauses 8 and 9, external communication (EC) is defined as communication between vehicle gateway (MN or ASN) and server at infrastructure domain (IN-CSE). For example, the UU interface in 3GPP in V2X [i.9] and [i.10]. Inter-vehicle communication (InterVC) is defined as communication between vehicle gateways (MN or ASN). For example, PC5 in 3GPP in V2X [i.9] and [i.10]. These communications could use TLS or DTLS, provided that latency constraints can be satisfied. The protection of communications is also discussed in PREparing SecuRe Vehicle-to-X (PRESERVE) Project and 3GPP. PRESERVE proposed secure communication for EC and InterVC [i.8] and 3GPP defined specification for EC and InterVC using LTE [i.9] and [i.10].

10.5.2 Intra-vehicle communication

10.5.2.1 Solution Description

Intra-vehicle communication (IntraVC) is defined as communication between vehicle gateway (MN or ASN) and ECU (ADN-AE or non-oneM2M) or external sensors (ADN-AE or non-oneM2M) in clause 8.

The former is a communication in the control system, and Controller Area Network (CAN) is used as the communication protocol. Although ECU is a low resource device, a strict response latency is defined as service performance requirement because the control system is impacting human life.

The latter is a communication in information system and Wi-Fi, Bluetooth or Ethernet are used as communication protocol. The devices in this communication range from low resource device such as external sensor to relatively powerful resource such as smartphone. Service performance requirements level in the information system is not high as compared with the control system. We propose two categories from the point of view of systems in IntraVC. The classifications are described in [i.17].

Table 10.5.2-1: IntraVC classifications

System	Resource of devices	Protocol	Example
Information system	High or low resource	Wi-Fi, Bluetooth, Ethernet	Vehicle gateway and smartphone Vehicle gateway and external sensor
Control system	Low resource	CAN	Vehicle gateway and ECUs ECU and the other ECU

In the information system, it is possible to establish a secure channel using security protocols which are described in clause 10.5.1.1, because there is no strict requirement such as response latency within a few tens of milliseconds and it can use TCP/IP network. If external sensors have low resource, security protocols which are designed based on lightweight cryptography (i.e. lightweight encryption algorithm or lightweight MAC algorithm) may be suitable to establish secure channel. Lightweight cryptography is discussed in ISO/IEC29192 [i.7] and lightweight encryption algorithm is discussed in Recommendation ITU X.1362 [i.5].

In the control system, there are severe service performance requirements such as response latency within a few tens of milliseconds because information sent from the ECU is involved in vehicle control. Almost all communications in control system send messages by plain text. However, Hardware Security Module (HSM) was discussed in E-safety Vehicle Intrusion proTected Application (EVITA), and EVITA defined three categories of security requirements: EVITA-HSM-Full-Version, EVITA-HSM-Medium-Version and EVITA-HSM-Light-Version [i.11]. According to the definition of EVITA, ECUs correspond to EVITA-HSM-Medium. EVITA also proposed transport protocol with secure features based on EVITA HSM [i.12].

10.5.3 Solution Applicability

This solution applies to Key Issue 4.

10.6 Solution F: Hardware Secure Element

10.6.1 Solution Description

In some use case, credentials are critical to the authentication between entities and the verification of integrity. If the credential is compromised or tampered with, the result of authentication or verification of integrity is not reliable. Therefore, the credential should be stored in tamper resistance storage and protect cryptographic operation from side channel attack. The following hardware secure elements provide tamper resistance and the highest protection level 3 which is defined at clause 6.3.1 in oneM2M TS-0003 [i.14], and could be suitable for protection of the credentials.

- Trusted Platform Module (TPM):
  - Trusted Computing Group (TCG) defined the specification of TPM for vehicle domain [i.9].
- EVITA-Hardware Security Module (HSM) implemented in tamper-resistant hardware:
  - EVITA defined the specifications of EVITA-Full-HSM, EVITA-Medium-HSM, EVITA-Light-HSM [i.11].
- Universal Integrated Circuit Card (UICC), embedded Universal Integrated Circuit Card (eUICC) [i.9]:
  - Usage of UICC is described in oneM2M TS-0003, annex D [i.14]. GSMA defined the specification of eUICC [i.9].
- Other secure elements supporting basic cryptographic services as specified in oneM2M TS-0003, annex L [i.14] specifies a set of cryptographic services.to be supported on tamper-resistant IoT secure elements.

10.6.2 Solution Applicability

This solution applies to Key Issue 4.

10.7 Solution G: Cross-Resource Subscription #2

10.7.1 Solution Description

In order to address the Key Issue 5, a subscriber AE/CSE creates resource subscriptions where automatic notifications depend on two or more resources, not a single resource as described in the previous solution. In this clause, a new candidate solution for the same use case but with different resource types and procedures is illustrated.

In short, a new resource type *subscriptionAssociation* is suggested which has two-ways links from/to existing *subscription* resource type. When a <subscriptionAssociation> resource is created which associates existing <subscription> resources, notifications generated per <subscription> resource is sent to the <subscriptionAssociation> resource Hosting CSE to generate a single but not just aggregated notification to the cross-resource subscriber.

The main difference between the previous and this one is whether the solution relies on group functionality (i.e. *group* resource type) or not while making cross-resource subscription relationship.

10.7.2 Solution Applicability

This solution applies to Key Issue 5.

10.7.3 New Resources and Procedures

10.7.3.1 Introduction

A new resource <*subscriptionAssociation*> is proposed to implement Cross-Resource Subscription functionality, associating with existing <*subscription*> resources.

Basically two-way links are maintained between a <*subscriptionAssociation*> resource and <*subscription*> resource(s). So that individual notification per <*subscription*> resource is sent to the <*subscriptionAssociation*> resource Hosting CSE referring the link in the <*subscription*> resource.

This link is also used to manage the list of individual subscriptions in the <*subscriptionAssociation*> resource. E.g. when a <*subscription*> resource is removed, then the link to the resource in the <*subscriptionAssociation*> resource is removed also. If there is a newly created resource subscription and needs to be associated with the existing <*subscriptionAssociation*> resource, then the subscriber can just add the link to that <*subscription*> resource. There's no need to do something for the <*group*> resource update.

To have a single event notification from cross or multiple resource subscriptions, the *timeWindow* attribute is suggested to check all the individual events are occurred during the time window, so the single notification needs to be sent to the subscriber of the <*subscriptionAssociation*> resource.

10.7.3.2 New *subscriptionAssociation* resource type

The new resource type *subscriptionAssociation* is suggested to have link/association to existing <*subscription*> resources. The Hosting CSE of this resource receives notifications from the <*subscription*> resource Hosting CSEs, and generate a single notification to the subscriber when all the events from individual subscription occurred in a time window that is set by the subscriber. The subscriber considers those events occurred within the window consist of a meaningful information so the single notification for that is enough.

Table 10.7-1: Child resources of <*subscriptionAssociation*> resource

Child Resources of <*container*>	Child Resource Type	Multiplicity	Description
subDel	<*subscriptionLinkDeletion*>	1	This is the virtual resource only permits DELETE operation. The <subscription> Hosting CSE in the *subsciptionIDs* list is allowed to delete the <subscription> resource from the list.

Table 10.7-2: Resource specific attributes of <*subscriptionAssociation*> resource

Attributes of <*subscriptionAssociation*>	Multiplicity	RW/ RO/ WO	Description
subscriptionIDs	1 (L)	RW	This attribute indicates the resource address(es) of associated <*subscription*> resources.
timeWindow	1	RW	This attribute indicates the time duration (e.g. in seconds) that the cross-resource subscriber considers all associated subscription events are occurred in the time window so that a single notification is need for those event.
notificationType	1	RW	This indicates the type of information for notifications of this <*subscriptionAssociation*> resource. Possible values are: 1) simple notice that all associated subscriptions have events in the time window; and 2) aggregation of the notifications from associated subscriptions.

10.7.3.3 Extension to *subscription* resource type

A new attribute is added to have a link to <*subscriptionAssociation*> resource. This is used to remove the link from the associated subscription to the individual subscription as well as to send a notification to the <*subscriptionAssociation*> resource Hosting CSE.

Table 10.7-3:New attributes of <*subscription*> resource

Attributes of <*subscription*>	Multiplicity	RW/ RO/ WO	Description
associatedSub	0..n	RW	This attribute lists *eventNotificationCriteria* for each target resource involved in a cross-resource subscription.

10.7.3.4 Procedure to create subscription association

When an Originator requests to create a <*subscriptionAssociation*> resource, after the basic checking procedures for resource creation as defined in TS-0001, the Hosting CSE creates the <*subscriptionAssociation*> resource. After this, the Hosting CSE sends UPDATE request(s) to <*subscription*> resources indicated in the *subscriptionIDs* attribute to include the *associatedSub* attribute targeting the created <*subscriptionAssociation*> resource.

10.7.3.5 Procedure to manage subscription association

After the creation of a <*subscriptionAssociation*> resource, when the subscriber wants to associate a new <*subscription*> resource to the existing subscription association, the <subscription> Hosting CSE sends UPDATE Request to the <*subscriptionAssociation*> resource to update the *subscriptionIDs* attribute.

After the creation of a <*subscriptionAssociation*> resource, when an associated <*subscription*> resource having the associatedSub attribute gets deleted, the <*subscription*> Hosting CSE sends DELETE Request to the virtual child resource, which has the fixed name "subDel", of the <*subscriptionAssociation*> resource. Then the <*subscriptionAssociation*> resource does not wait for notifications coming from the <*subscription*> Hosting CSE anymore to generated a single event to the subscriber.

10.7.3.6 Procedure to generate notifications of cross-resource subscription

After the creation of a <*subscriptionAssociation*> resource, when the subscriber wants to associate a new <*subscription*> resource to the existing subscription association, the <subscription> Hosting CSE sends UPDATE Request to the <*subscriptionAssociation*> resource to update the *subscriptionIDs* attribute.

Annex A(informative) : oneM2M data model for vehicular domain

A.1 AUTOPILOT

European Large Scale Pilot (LSP) Autopilot [i.18] is project which is focused on use of IoT for improving autonomous driving. “Automated driving Progressed by Internet Of Things” (AUTOPILOT) will bring IoT into the automotive world to transform connected vehicles — moving ”things” in the IoT ecosystem — into highly automated vehicles (towards levels 4 and 5 – see 5.4 Levels of Driving Automation on description of levels of automation). While using the IoT data for automated driving, AUTOPILOT will also make data from autonomous cars available to the Internet-of-Things platforms, thus being also source of data.

The extent and volume of information sources that can be addressed through internet of things is seamlessly unlimited, offering potential improvements of automated driving functions (including improvements in security, efficiency, accuracy, etc.) and the information will enable services involving automated driving. Various use cases are executed implemented at the 6 pilot sites of AUTOPILOT in large scale demonstrations in order to evaluate the potential and calculate the related impacts of using Internet of Things for Automated Driving.

The AUTOPILOT consortium represents all relevant areas of the IoT eco-system. Thanks to AUTOPILOT, the IoT eco-system will involve vehicles, road infrastructure and surrounding objects in the IoT, with particular attention to safety critical aspects of automated driving. AUTOPILOT IoT enabled autonomous driving cars are tested, in real conditions, at six permanent large-scale pilot sites in Finland, France, Italy, the Netherlands, Spain and South Korea.

A.2 AUTOPILOT and use of IoT

Automated driving is expected to increase safety, provide more comfort and create many new business opportunities for mobility services. The Internet of Things (IoT) is about enabling participating devices (and applications) to be able to access more data than typical connected car would, and in this way to be able to make better decisions, or create more precise world model. In both cases, idea is that having more data available, it will be possible to improve existing services as well as provide new types of services.

As example of new type of services, one can think of application which can predict movements of VRU (Vulnerable Road Users) and provide them as input to vehicle’s world model. In this example, cameras from smart city can monitor streets, and using object recognition software, detect cyclist going in particular direction. An application, let’s call it ‘motion trajectory prediction’ can use data on observed movement of cyclist and predict its trajectory. Another sapplication can take trajectory and combine it with known or prediucted trajectory of vehicle. If two would meet in the same spot and same time, that means that crash might occur between two – vehicle and cycle. This is something that will be out of scope of traditional connected car and cooperative ITS (intelligent Transport Systems) – since cyclist is not directly observable by vehicle.

A.3 AUTOPILOT, oneM2M and other IoT platforms

In real-life deployments there will be another IoT platforms deployed by other entities.

Example of multiple platforms are different IoT platforms that will be deployed by road operator and smart city operator. Typically, highways are under jusrisdiction and control of so-called road operator, or road authority, which is able to set and enforce rules for use of roads. For example, road operator can open emergency lane for traffic when road becomes too busy. Or they can reserve particular lane for platooning vehicles.

But when vehicle leaves highway and enters the city, roads / streets within city are under jurisdiction and control of other entity – smart city operator, one that has information on state of streets, parking places, roadworks, …

Depending on its location, vehicle will exchange data with different platforms. And those paltforms are not necessarily oneM2M based.

But some information about vehicle must be exchanged between these two platforms in our example, and in effect will lead to situation that we have federation of IoT platforms. And that puts emphasis on interworking, but also on data that will be exchanged via interworking interfaces.

A.4 oneM2M data model for vehicular domain

In AUTOPILOT, central IoT platform is oneM2M platform.

As currently there is not data model which is defined by oneM2M for vehicular domain, partners within AUTOPILOT are facing choice whether to go bottom-up, and to define which data is needed for given use cases, and then determine which data model to use. Or, to start from one of existing data models – for example SAREF [i.19] or Sensoris [i.20].

SAREF [i.19] stands for Smart Appliances Reference ontology, and it is OWL language (Web Ontology Language). It is conceived as a shared model of consensus that facilitates the matching of existing assets in the smart appliances domain, reducing the effort of translating from one asset to another, since SAREF [i.19] requires one set of mappings to each asset, instead of a dedicated set of mappings for each pair of assets.

But, SAREF [i.19] was designed initially for small applicances, and is currently being expanded (ETSI taskforce is working right now on it) to energy domain, agriculture and healthcare. But vehicular domain is still not covered.

Another development which is from automotive domain is Sensoris [i.20]. It is result of cooperation onf number of parties from automotive domain, and it proposed data model for exchanging data between vehicles and cloud. You can find more details in reference [i.20].

Currently, vehicle sensor data exists in different formats across automakers and it is typically carmaker specific. Even some protocols like CAN are defining transpoprt protocol for transporting data within vehicle, but do not go into standardizing messages that are transported. When connecting car to IoT platform - standardization is needed, as pooling analogous vehicle data from millions of vehicles will be a key enabler for bringing vehicle-to-vehicle and vehicle-to-infrastructure communication to the next level.

SENSORIS [i.20] was initiated by HERE in June 2015 when the company published the first open specification for how vehicle sensor data gathered by connected cars will be sent to the cloud (as well as between clouds) for processing and analysis.

A.5 Proposal for oneM2M data model for vehicular domain

Seeing that SAREF [i.19] is ETSI standard, but lacks vehicular data model, while Sensoris [i.20] is industrial standard which covers vehicular doimain, and of course there are others that for example cover real-time systems data models, we would like to propose to use SAREF [i.19] as the base to which we will add vehicle specific elements from Sensoris [i.20], and others namely regarding real-time systrem's data models.

History

Publication history		
V3.0.1	May 2019	Release 3 - Publication