International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.1377

(10/2022)

SERIES X: DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

Secure applications and services (2) – Intelligent transportation system (ITS) security

Guidelines for an intrusion prevention system for connected vehicles

Recommendation ITU-T X.1377

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

PUBLIC DATA NETWORKS	X.1–X.199
OPEN SYSTEMS INTERCONNECTION	X.200–X.299
INTERWORKING BETWEEN NETWORKS	X.300–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	X.600–X.699
OSI MANAGEMENT	X.700–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	X.850–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999
INFORMATION AND NETWORK SECURITY	
General security aspects	X.1000–X.1029
Network security	X.1030–X.1049
Security management	X.1050–X.1069
Telebiometrics	X.1080–X.1099
SECURE APPLICATIONS AND SERVICES (1)	
Multicast security	X.1100–X.1109
Home network security	X.1110–X.1119
Mobile security	X.1120–X.1139
Web security (1)	X.1140–X.1149
Application Security (1)	X.1150–X.1159
Peer-to-peer security	X.1160–X.1169
Networked ID security	X.1170–X.1179
IPTV security	X.1180–X.1199
CYBERSPACE SECURITY	
Cybersecurity	X.1200–X.1229
Countering spam	X.1230–X.1249
Identity management	X.1250–X.1279
SECURE APPLICATIONS AND SERVICES (2)	
Emergency communications	X.1300–X.1309
Ubiquitous sensor network security	X.1310–X.1319
Smart grid security	X.1330–X.1339
Certified mail	X.1340–X.1349
Internet of things (IoT) security	X.1350–X.1369
Intelligent transportation system (ITS) security	**X.1370–X.1399**
Distributed ledger technology (DLT) security	X.1400–X.1429
Application Security (2)	X.1450–X.1459
Web security (2)	X.1470–X.1489
CYBERSECURITY INFORMATION EXCHANGE	
Overview of cybersecurity	X.1500–X.1519
Vulnerability/state exchange	X.1520–X.1539
Event/incident/heuristics exchange	X.1540–X.1549
Exchange of policies	X.1550–X.1559
Heuristics and information request	X.1560–X.1569
Identification and discovery	X.1570–X.1579
Assured exchange	X.1580–X.1589
Cyber Defence	X.1590–X.1599
CLOUD COMPUTING SECURITY	
Overview of cloud computing security	X.1600–X.1601
Cloud computing security design	X.1602–X.1639
Cloud computing security best practices and guidelines	X.1640–X.1659
Cloud computing security implementation	X.1660–X.1679
Other cloud computing security	X.1680–X.1699
QUANTUM COMMUNICATION	
Terminologies	X.1700–X.1701
Quantum random number generator	X.1702–X.1709
Framework of QKDN security	X.1710–X.1711
Security design for QKDN	X.1712–X.1719
Security techniques for QKDN	X.1720–X.1729
DATA SECURITY	
Big Data Security	X.1750–X.1759
Data protection	X.1770–X.1789
IMT-2020 SECURITY	X.1800–X.1819

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T X.1377

Guidelines for an intrusion prevention system for connected vehicles

Summary

Recommendation ITU-T X.1377 establishes guidelines for an intrusion prevention system (IPS) for connected vehicles. This Recommendation mainly focuses on aspects of active response capability for intrusion and includes the implementation guidance and use cases of IPS for connected vehicles.

Prior in-vehicle intrusion detection systems (IDSs) have limitations, e.g., requiring too many computing resources that a vehicle cannot provide and being unable to mitigate intrusions due to characteristics of protocol and bus topology. To overcome these limitations of conventional in-vehicle IDSs, this Recommendation provides methodologies for both intrusion detection and intrusion prevention. The proposed IPS consists of the intrusion detection plane – an external component with intrusion detection algorithms – and the data plane – in-vehicle networks (IVNs) where traffic monitoring and active response happen. This Recommendation aims to protect (automotive) Ethernet-based IVNs.

History

Edition	Recommendation	Approval	Study Group	Unique ID[*]
1.0	ITU-T X.1377	2022-10-14	17	11.1002/1000/15103

Keywords

Automotive intrusion prevention system, intrusion detection system (IDS), intelligent transportation systems (ITS).

[*] To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents/software copyrights, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the appropriate ITU-T databases available via the ITU-T website at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T X.1377

Guidelines for an intrusion prevention system for connected vehicles

1 Scope

This Recommendation establishes guidelines for an intrusion prevention system (IPS) for connected vehicles. This Recommendation mainly focuses on aspects of active response capability for intrusion and includes implementation guidance and use cases of IPS for connected vehicles.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T X.1371] Recommendation ITU-T X.1371 (2020), *Security Threats to Connected Vehicles*.

[ITU-T X.1375] Recommendation ITU-T X.1375 (2020), *Guidelines for an Intrusion Detection System for in-Vehicle Networks*.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 availability [b-ITU-T X.800]: The property of being accessible and useable upon demand by an authorized entity.

3.1.2 confidentiality [b-ITU-T X.800]: The property that information is not made available or disclosed to unauthorized individuals, entities, or processes.

3.1.3 data integrity [b-ITU-T X.800]: The property that data has not been altered or destroyed in an unauthorized manner.

3.1.4 flow [b-IETF RFC 5101]: A set of IP packets passing an observation point in the network during a certain time interval. All packets belonging to a particular flow have a set of common properties. Each property is defined as the result of applying a function to the values of:

1) One or more packet header fields (e.g., destination IP address), transport header fields (e.g., destination port number), or application header fields (e.g., RTP header fields)

2) One or more characteristics of the packet itself (e.g., number of MPLS labels, etc.)

3.1.5 intrusion detection and prevention system (IDPS) [b-ISO/IEC 27039]: Intrusion detection system (IDS) and intrusion prevention system (IPS) software applications or appliances that monitor systems for malicious activities, where IDS focus is to only alert on the discovery of such activity while IPS has the potential to prevent some intrusions upon detection.

3.1.6 switch [b-ITU-T E.417]: Device that dynamically interconnects physical or virtual links to form a connection for information transfer.

3.1.7 threat [b-ISO/IEC 27000]: Potential cause of an unwanted incident, which may result in harm to a system or organization.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 external component: A software application, firmware or appliance that operates outside a vehicle for in-vehicle intrusion detection and prevention, including:

- external switch controller;
- intrusion detection algorithm and configurations;
- intrusion detection system and database storage for storing detection results;
- infrastructure to maintain vehicle-to-infrastructure communications when connected vehicles are moving.

3.2.2 flow entry: An element in a flow table used to describe a flow, corresponding instruction, and additional information that consists of the following fields: match, instruction, priority, timeout, packet counter and byte counter.

3.2.3 flow table: A set of flow entries.

3.2.4 programmable switch: A managed switch that handles incoming packets by referring to a built-in flow table.

3.2.5 switch controller: An external component that monitors and manages flow entries for programmable switches.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

ADAS	Advanced Driver Assistance System
AVTP	Audio Video Transport Protocol
CAN	Controller Area Network
DoIP	Diagnostic over Internet Protocol
ECU	Electronic Control Unit
EMS	Engine Management System
ID	Identifier
IDPS	Intrusion Detection and Prevention System
IDS	Intrusion Detection System
IP	Internet Protocol
IPS	Intrusion Prevention System
IVN	In-Vehicle Network
MAC	Media Access Control
MPLS	Multiprotocol Label Switching
OBD-II	On-Board Diagnostics-II
RADAR	Radio Detection And Range
RSU	Roadside Unit

RTP	Real-Time Transport Protocol
TCP	Transmission Control Protocol
TCU	Telematics Control Unit
UDP	User Datagram Protocol
USB	Universal Serial Bus
V2I	Vehicle to Infrastructure
V2X	Vehicle to everything

5 Conventions

None.

6 Security threats on connected vehicle environment

6.1 Attack surface

With the evolution of intelligent transportation systems, connectivity, and computerized automotive technology, connected vehicles are exposed to various security threats. Attack surfaces have grown because of the convenience features of various devices. Diverse threats to connected vehicles may give an attacker access to a target electronic control unit (ECU). General security threats related to connected vehicles are listed in clause 7 of [ITU-T X.1371]. In addition, identified threats to in-vehicle networks (IVNs) that are based on a controller area network (CAN) are discussed in clause 7 of [ITU-T X.1375].

Figure 1 shows the overall attack surface for connected vehicles. This Recommendation considers that the primary goal of an attacker is to transmit messages (e.g., CAN messages, Ethernet frames, Internet protocol (IP) packets or transmission control protocol (TCP) segments) to a target ECU to breach confidentiality, data integrity or availability. To this end, the attacker can access the target in-vehicle ECU via one of the following attack paths:

- **remote interface** e.g., a vehicle-to-everything (V2X) modem, Bluetooth, wireless fidelity, dedicated short-range communications;
- **physical interface** e.g., an on-board diagnostics-II (OBD-II) port, universal serial bus (USB) to Ethernet adapter, USB flash drive;
- **compromised ECU** e.g., a vulnerable infotainment system, sensor operated by malicious firmware.

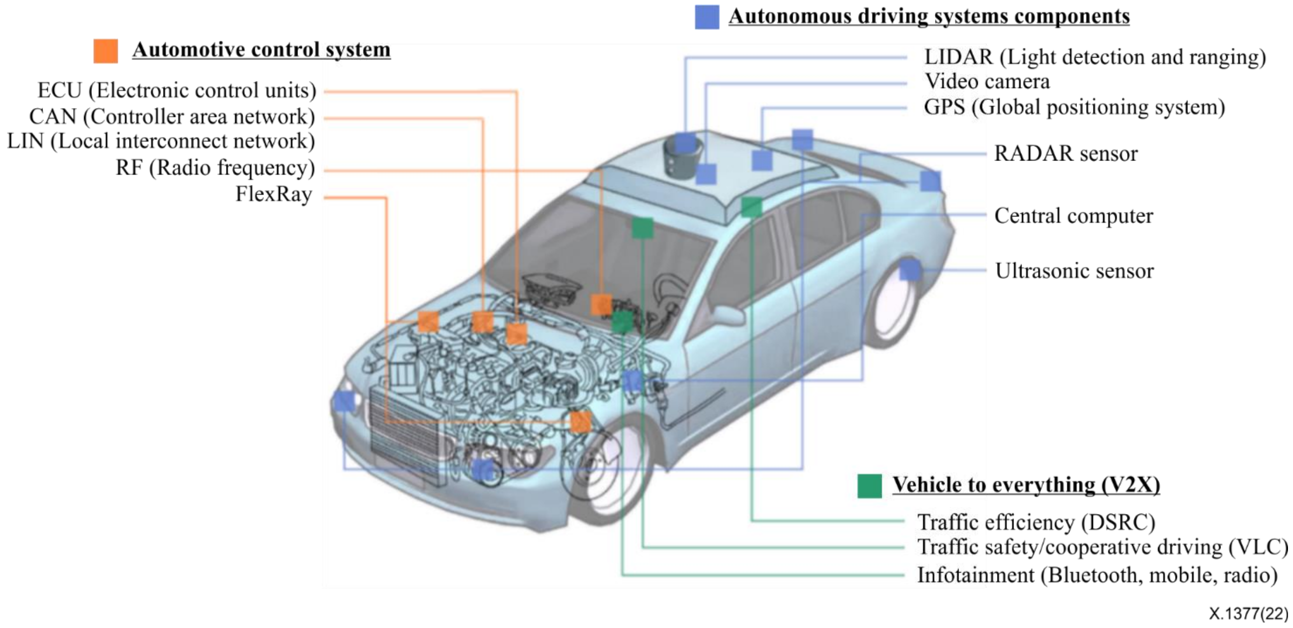

🟧 Automotive control system
ECU (Electronic control units)
CAN (Controller area network)
LIN (Local interconnect network)
RF (Radio frequency)
FlexRay
🟦 Autonomous driving systems components
LIDAR (Light detection and ranging)
Video camera
GPS (Global positioning system)
RADAR sensor
Central computer
Ultrasonic sensor
🟩 Vehicle to everything (V2X)
Traffic efficiency (DSRC)
Traffic safety/cooperative driving (VLC)
Infotainment (Bluetooth, mobile, radio)
X.1377(22)

Figure 1 – Overall attack surface for connected vehicles

6.2 Malicious behaviour in in-vehicle networks

In this clause, network-level malicious behaviour, which need to be detected by the proposed IPS for connected vehicles is discussed. Malicious behaviour can consist of intrusions, i.e., transmitting unauthorized spoofing messages. As a result of malicious behaviour, flows that did not exist before can occur. Otherwise, the statistics of the existing flow changes. Clauses 6.2.1 to 6.2.5 provide examples of malicious behaviour in Ethernet-based IVNs.

6.2.1 Identification of target electronic control units

The first malicious behaviour right after gaining access to IVNs can be the identification of target ECUs. To this end, the attacker can transmit broadcast messages, ping requests, etc. to all reachable ECUs. This behaviour may not breach confidentiality, data integrity or availability; however, the behaviour needs to be considered and detected because it is a proof that an attacker has access to the connected vehicle.

6.2.2 Port scanning or service discovery

An attacker can conduct a TCP or user datagram protocol (UDP) port scanning or a service discovery on a specific ECU. In addition, an attacker may also trigger an OBD function to acquire additional information, available services and the status of an ECU. Through malicious behaviour, unintended information can be shared or leaked.

6.2.3 Denial of service attack

To breach availability, an attacker can conduct a denial of service attack via the following methods:

– transmitting malformed application data;

– performing a TCP SYN (SYN is a message in TCP handshake) flooding attack;

– establishing multiple sessions in a short time;

– transmitting a single huge payload to cause network bottleneck.

6.2.4 Command injection and spoofing attack

An attacker can attack by command injection with well-crafted payloads. As a result, the attack can breach confidentiality and data integrity as follows:

– **Confidentiality** e.g., by acquiring recent driving logs, current geolocation, driving speed and phonebook or message entries in an infotainment system.

– **Data integrity** e.g., by controlling an advanced driver assistance system (ADAS) and convenient features such as heating, ventilation and air-conditioning.

6.2.5 Malicious behaviour related to controller area network buses

A CAN-based IVN is expected to remain in connected vehicles for time-critical legacy applications, such as the powertrain system and OBD-II. A gateway needs to be installed to support communications with ECUs over a heterogeneous network. The gateway translates a CAN message to an Ethernet frame and *vice versa*. For example, in Ethernet-based IVNs, the gateway can represent a CAN message with an audio video transport protocol (AVTP) packet.

As discussed in [ITU-T X.1375], an attacker who has access to CAN-based IVNs can inject arbitrary messages to take control of the target connected vehicle. Since the gateway continuously translates CAN messages, flows can be affected by attacks that occur on the CAN bus.

7 Architecture of intrusion prevention system

7.1 Overview

This clause specifies the IPS architecture consisting of the data, control and detection planes (see Figure 2). The data plane represents IVNs where the IPS is designed to protect against various security threats to connected vehicles. The control plane maintains a connection between those for data and detection planes. The detection plane identifies intrusions occurring on the data plane (i.e., IVNs of connected vehicles).

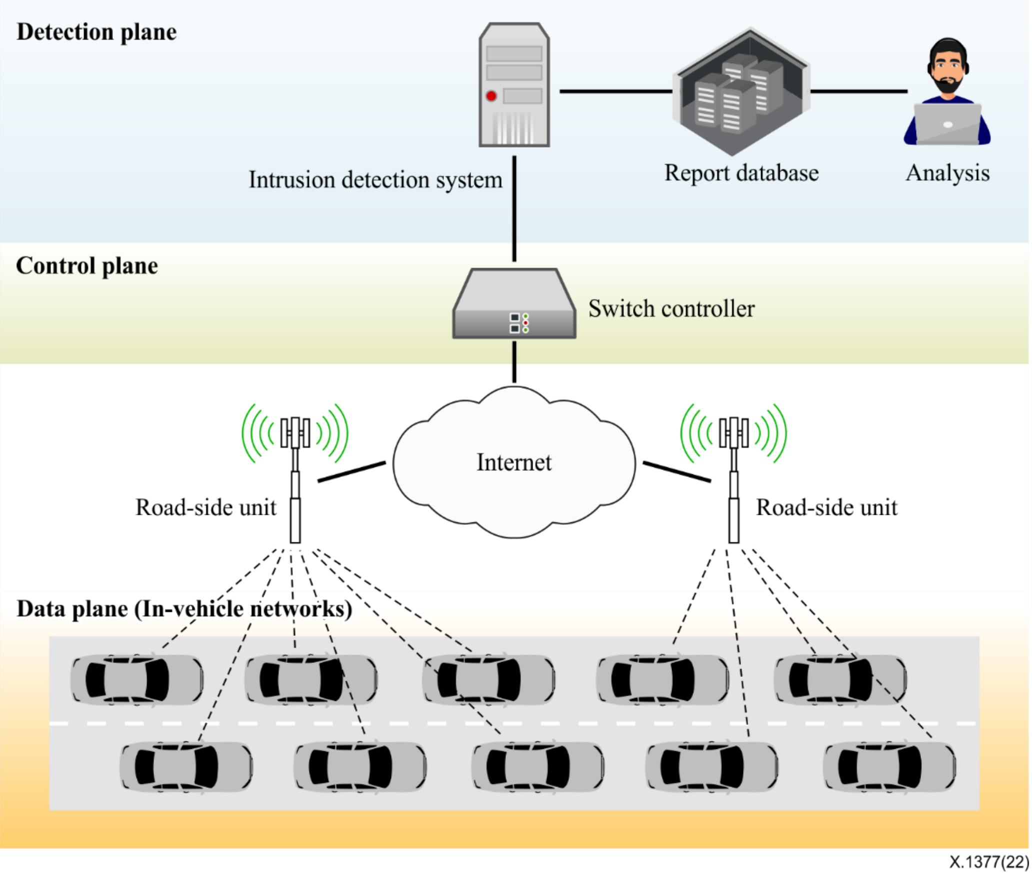


Figure 2 – Intrusion prevention system for connected vehicles

7.1.1 Data plane

In the data plane, each connected vehicle contains its own IVN. The IPS monitors, analyses, and manages all in-vehicle traffic. Connected vehicles are connected to the external control plane via V2X communications while in motion. To protect IVNs using an external IDS, the data plane transmits information about in-vehicle traffic to the control plane. Specifically, an in-vehicle programmable switch (discussed in clauses 7.2 and 7.3) monitors potential intrusions and abnormal CAN traffic. The data plane can also respond to an identified intrusion, e.g., by dropping the attack traffic.

7.1.2 Control plane

The control plane maintains connections with multiple connected vehicles and external IDSs. The switch controller translates messages between connected vehicles and external IDSs. Multiple switch controllers can be harmonized for distributed computing.

7.1.3 Detection plane

The detection plane consists of an external IDS and a database containing detection reports which can be used for post-analysis of intrusions within IVNs.

An external IDS gathers in-vehicle traffic information from the control plane and determines there is an intrusion. The external IDS stores information about any attack detected in the database so that an expert can further analyse it, identify the root cause,C and prepare a corresponding remedy.

To block an identified intrusion, the external IDS prepares a command for a particular packet header of identified traffic. The command is sent to the control plane, then the switch controller translates and delivers the command to a designated vehicle.

7.2 Topological structure of an in-vehicle network

Figure 3 depicts a configuration of an IVN composed of a vehicle-to-infrastructure (V2I) modem, various ECUs, in-vehicle programmable switches and a gateway. Each Ethernet-based ECU is connected to an in-vehicle programmable switch directly. Thus, all in-vehicle traffic is transmitted over one or more switches to reach to its destination. This characteristic of an Ethernet-based IVN allows programmable switches to monitor or manage in-vehicle traffic.

The CAN bus is utilized for limited legacy and low data rate applications, such as powertrain systems. The gateway between the Ethernet and CAN bus translates one to another type of message or packet. Consequently, an in-vehicle programmable switch can monitor potential abnormal CAN traffic.

The components are listed in Table 1.

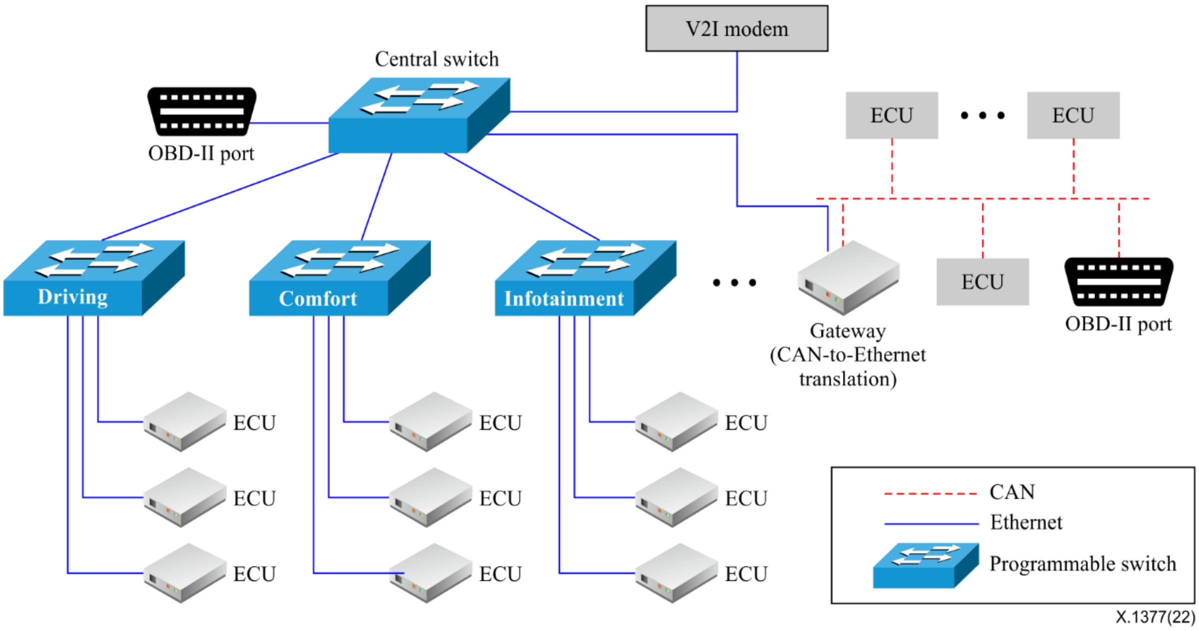


Figure 3 – Configuration of an in-vehicle network and programmable switches

Table 1 – In-vehicle network components in connected vehicles

Component	Features
Programmable switch	A programmable packet-forwarding device. Each switch has at least one flow table that can be managed by an external switch controller. The flow table is a set of flow entries. The switch forwards traffic based on its flow table
V2I modem	A wireless interface device used for V2I communication for navigation systems with the intelligent transport systems, user content transfer and over-the-air updates. Furthermore, the modem transmits and receives control messages between the external switch controller and in-vehicle programmable switches
Gateway	Translates a CAN message to an Ethernet frame and *vice versa*
ECU	Any of the sensors, infotainment devices and comfort devices, like a power-operated window. This device can be targeted for attack. Additionally, a compromised ECU can be an attack node
CAN bus	Provides message transmission over traditional applications that require deterministic latency and message prioritization
OBD-II port	An access point to the IVN for vehicle diagnosis

7.3 In-vehicle programmable switch

7.3.1 Flow table and flow entry

A flow table in an in-vehicle programmable switch contains flow entries. The external switch controller manages the flow table by adding, modifying, or removing flow entries.

A flow entry consists of the following fields:

– **Match**: A set of combinations of packet header and value used to specify inbound packets. For example, the match field "ip.src==10.0.0.5 and udp.src==3000" is used to identify all packets from the IP address 10.0.0.5 and the UDP port 3000.

– **Priority**: Determines which flow entry is examined first.

– **Counter**: Consists of a packet counter and a byte counter. A packet counter contains the number of matched packets. A byte counter contains the sum of the packet size of the matched packets.

– **Instruction**: Determines how an in-vehicle programmable switch handles matched packets. The instruction can be one of the following: forward to a physical port; forward to the external switch controller; forward to all ports (i.e., broadcast); forward to a group (i.e., multicast for audio/video bridging); drop; and a list of those instructions.

– **Timeout**: Is optional and applied to flow entries to prevent them occupying switch memory indefinitely. A soft timeout is calculated after the last matched packet and a hard timeout is calculated after the first installation of the corresponding flow entry.

– **Cookie**: Is optional and can contain any values determined by the external switch controller.

7.3.2 Flow table lookup procedure

The in-vehicle programmable switch looks up a flow table for every inbound packet. The flow table lookup procedure examines all flow entries one by one, according to their priority. For each flow entry, the flow table lookup procedure refers to the match field and examines a given inbound packet.

If the table lookup procedure finds an existing flow entry that matches the inbound packet: the packet counter and byte counter increase; an instruction is executed for the packet; and the procedure stops the flow table lookup. A general instruction is to forward a benign packet to a designated port (or drop a suspicious packet).

7.3.3 Table miss

An in-vehicle programmable switch cannot process inbound packets when a flow table is empty, or a corresponding flow entry is not installed. The term "table miss" represents the situation when the flow table lookup procedure cannot find any flow entry (except a table miss entry) for a given packet.

The table miss is handled by a flow entry, called a "table miss entry". The table miss entry is designed to notify inbound packets to the switch controller. The values of the table miss entry are detailed in Table 2.

Table 2 – Values of a table miss entry

Field	Value	Description
Match	* (ANY)	The table miss entry needs to handle any packets
Priority	Lowest	The table miss entry needs to be examined last
Instruction	Forward to the switch controller	An in-vehicle programmable switch sends an unknown packet to the switch controller
Timeout	idle_timeout == None and hard_timeout == None	The table miss entry has not expired

7.3.4 Communication with switch controller

An in-vehicle programmable switch needs to communicate with a switch controller to notify an event and receive a new flow entry. The communication is especially important, since in-vehicle programmable switches do not make any decision about packet processing. In other words, an in-vehicle programmable switch does not modify its flow table itself.

To implement the proposed IPS, in-vehicle programmable switches and external switch controllers need to support the message types listed in Table 3. A switch controller can distinguish an in-vehicle programmable switch using a switch identifier (ID).

The packet-in message occurs when an in-vehicle programmable switch executes the "forward to the switch controller" instruction. In many cases, a packet-in event is a table miss event. The switch controller can identify the table miss event by checking whether the flow entry in the parameter is a table miss entry or not. To address the table miss event, the switch controller needs to build an appropriate flow entry with the instruction to "forward to a physical port". The switch controller then sends a packet-out message to manipulate a flow table of an in-vehicle programmable switch.

The messages are used to detect intrusion occurring in IVNs. The event-driven detection (clause 9.2.1) and deep packet inspection (clause 9.2.3) mainly depend on packet-in messages. The data-driven detection (clause 9.2.2) mainly depends on query and response flow table messages.

Table 3 – List of messages between an in-vehicle programmable switch and a switch controller

Message type	Direction	Parameter(s)	Description
Hello	Switch→ Controller	Switch ID	The switch is online
Packet-in	Switch→ Controller	Switch ID, a flow entry, a packet	The controller receives a packet from the switch
Packet-out	Controller→ Switch	Switch ID, a flow entry, *a packet (optional)*	The controller sends a flow entry to the switch to install or modify it in or remove it from the flow table of the switch. If a packet is given, the packet is processed at the switch after setting up the flow entry
Flow removed	Switch→ Controller	Switch ID, a flow entry, a reason	An existing flow entry is removed. The reason is either an idle timeout or a hard timeout
Link status changed	Switch→ Controller	Switch ID, Ethernet port No., is_active	A physical link status is changed. The parameter "is_active" indicates whether the corresponding port has become active or inactive
Query flow table	Controller→ Switch	Switch ID	The controller requests a flow table (i.e., flow entries)
Response flow table	Switch→ Controller	Switch ID, list of flow entries	The switch sends a flow table to the controller in response to the request
Bye	Switch→ Controller	Switch ID	The switch is shutting down

7.4 External intrusion detection system in the detection plane

The in-vehicle programmable switch communicates with the IDS in the detection plane through the external switch controller. After collecting traffic from many vehicles, the IDS decides whether a flow captured in a specific connected vehicle is harmful.

In this Recommendation, it is emphasized that the IDS is placed on the cloud platform. This characteristic contrasts to the in-vehicle IDS, which is mainly discussed in [ITU-T X.1375], in which an IDS operates in a vehicle. The external IDS allows dynamic addition, reconfiguration, or revocation of detection algorithms. Such updates are applied immediately and do not need to be deployed in vehicles. It allows algorithms to be updated anytime regardless of the status of connected vehicles; neither a physical access nor an over-the-air security update is needed. It is useful especially when vehicles have new functions or even when new patterns of attack arise.

Detection algorithms are executed outside connected vehicles. Hence, it is not required to consider the computational performance of connected vehicles.

The detection algorithm can aggregate traffic information from numbers of managed connected vehicles. For example, when an ambiguous flow is captured from a vehicle, traditional approaches face problems in deciding to alert, which could give rise to a false alarm. If a decision maker observes similar flows from many other vehicles, the detection result can be more precise.

8 External intrusion detection system

This clause covers two intrusion detection functions, namely of the remote and collaborative types.

The proposed IPS provides the following functions for detecting in-vehicle intrusions:

1) **remote detection** – from an external backend server;

2) **collaborative detection** – by collaboration of more than one external backend server.

8.1 Remote detection

Remote detection is a basic function for detecting in-vehicle intrusion, assisted by an IDS installed outside connected vehicles. The remote detection function allows any complex detection model (such as deep-learning-assisted artificial intelligence methods) to be considered to build an intrusion detection model.

Figure 4 depicts an external IDS and connected vehicles that communicate with each other over a control plane. Note that the control plane abstracts all connections between an external IDS and each connected vehicle. Specifically, a switch controller (located in the control plane, see Figure 2) maintains connections with the suitable vehicles and receiving events from in-vehicle programmable switches. The switch controller is the only component that the external IDS communicates with. Thus, the external IDS can focus on detecting intrusions.

Figure 4 also depicts the following three advantages of remote detection. First, parameters can be updated for intrusion detection algorithms anytime regardless of vehicle location and status. Second, the complex algorithm indicates that the external IDS can rely on a deep-learning model or intrusion detection algorithm that requires a lot of computing power. Finally, accurate detection means the external IDS has the advantage of identifying a stealthy attack by analysing traffic from a macro perspective. It is possible because the external IDS observes multiple vehicles simultaneously. Rich records and statistics captured from neighbouring vehicles can help reduce detection errors for obscure footprints caused by an intrusion and allow for more accurate intrusion detection.

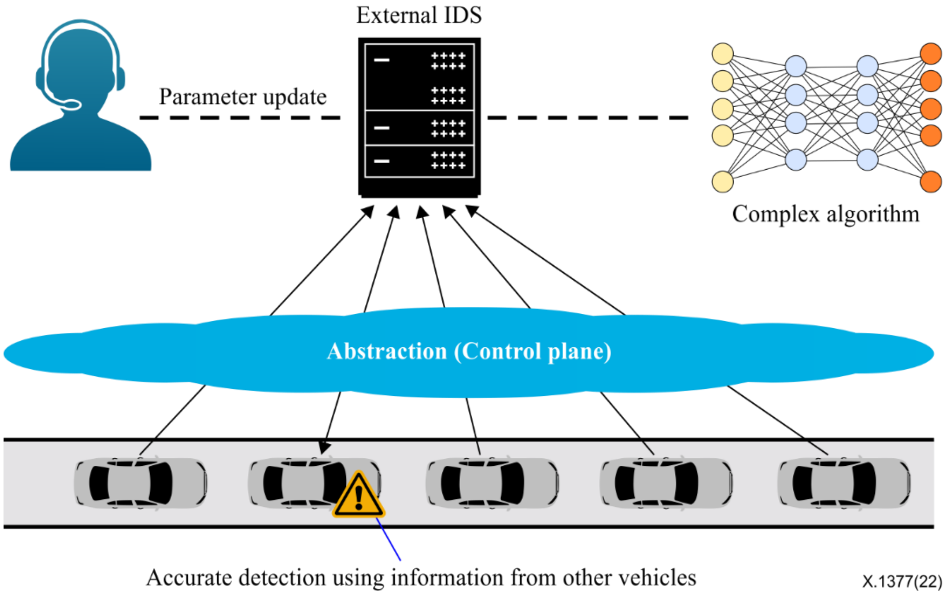

External IDS
Parameter update
Complex algorithm
Abstraction (Control plane)
Accurate detection using information from other vehicles
X.1377(22)

Figure 4 – External intrusion detection system and its characteristics

8.2 Collaborative detection

On the detection plane, multiple IDSs can be deployed and cooperate simultaneously. Depending on the situation, the following detection strategies can be prepared:

– Each IDS takes charge of a specific protocol. For example, IDS 1, IDS 2, and IDS 3 can be deployed to examine secure shell packets, AVTP packets and diagnostic over Internet protocol (DoIP) packets, respectively; or

– Multiple IDSs receive the same traffic information and calculate their own detection algorithm.

Given the same input, the multiple IDSs could return different detection results. For example, Figure 5 assumes a situation that two of the three IDSs detect malicious activities. The final decision is made using an ensemble detection, which is a method that decision is made by a majority vote.

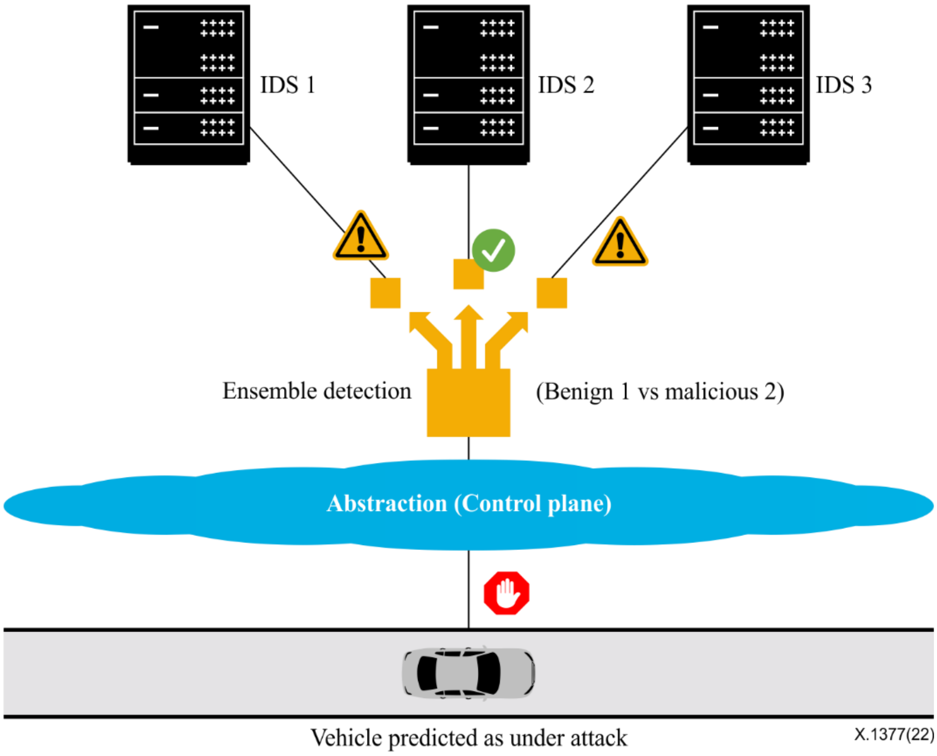

IDS 1
IDS 2
IDS 3
Ensemble detection
(Benign 1 vs malicious 2)
Abstraction (Control plane)
Vehicle predicted as under attack
X.1377(22)

Figure 5 – Multiple external intrusion detection systems for hybrid detection

9 Implementation guidelines for intrusion prevention systems

9.1 Overall procedure for the intrusion prevention system

As shown in Figure 6, the overall intrusion prevention procedure works as follows:

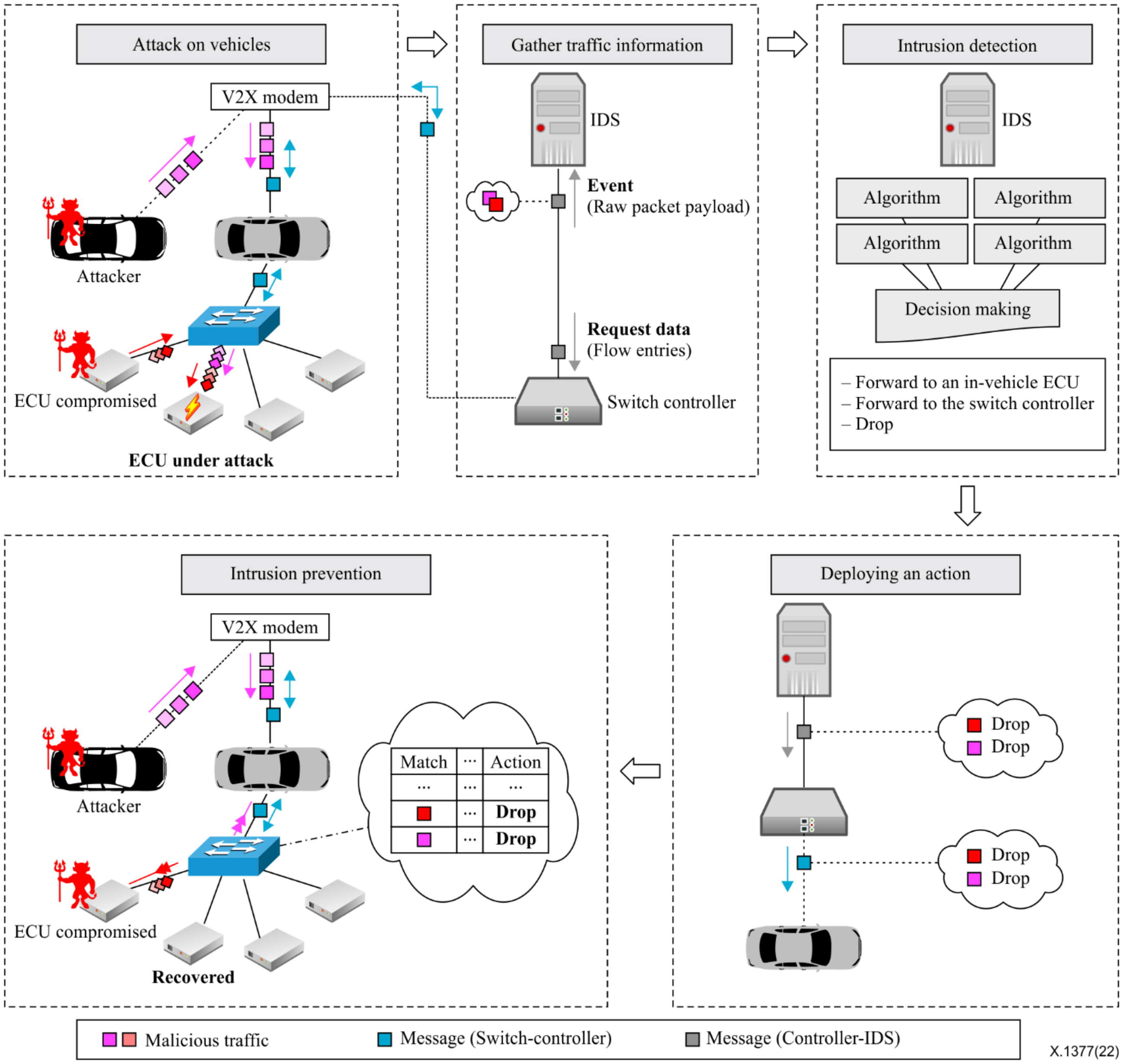
(1) **Attack on vehicles**: An intrusion happens at a connected vehicle. Attacks can occur from inside or outside. For example, a compromised ECU can inject malicious packets into IVNs directly. Also, nearby threats can access a target connected vehicle via the Internet or V2X communications.

(2) **Gather traffic information**: Information about incoming traffic (from inside or outside) is collected by an in-vehicle programmable switch and then sent to the control plane (i.e., the external switch controller discussed in clause 7.1.2). The switch controller forwards the information to the external IDS.

(3) **Intrusion detection**: The external IDS examines in-vehicle traffic by using various detection methodologies. If the IDS detects the intrusion, the IDS reports the attack information to the control plane.

(4) **Deploying an action**: The switch controller creates a new flow entry to block a flow identified as intrusion based on the report from the external IDS. The switch controller then sends the flow entry to the connected vehicle under attack.

(5) **Intrusion prevention**: After receiving a new flow entry from the control plane, the in-vehicle programmable switch applies the action immediately. Thus, any traffic is blocked according to the flow entry.



Match	···	Action
···	···	···
🟥	···	**Drop**
🟪	···	**Drop**

Figure 6 – Illustration of the intrusion prevention procedure

9.2 Methodology of intrusion detection in an intrusion prevention system

The IPS can handle event, data (flow statistics), and raw packet payload as input of external IDSs. Consequently, three types of intrusion detection method are available on the proposed IPS: 1) event-driven; 2) data driven; and 3) payload based. An IDS may utilize either a single or multiple methodologies, as necessary.

9.2.1 Event-driven detection

The event-driven detection method detects in-vehicle attacks based on triggered events, e.g., numbers of packet-in events from IVNs. In particular, the event-driven detection methodology is useful to detect attacks that trigger various table-miss events, e.g., port-scanning attacks and multiple simultaneous TCP connections. The methodology requires an IDS to receive events occurring in IVNs, including packet-in, flow removed, and link status changed messages.

Figure 7 shows how the event-driven detection methodology can detect in-vehicle intrusion with packet-in messages. The solid arrow means the control flow within a node, e.g., an in-vehicle programmable switch, a switch controller, or an external IDS, whereas the dotted arrow means the data flow between two planes. Especially, the two dotted arrows between the data plane and the control pane represent a packet-in event and a packet-out event.

The procedures tinted blue are the key part of the event-driven detection methodology. When a packet-in message is sent to the control plane, the control plane forwards it to the detection plane. As a result, the external IDS can recognize the event immediately. In the event-driven detection methodology, one of the two following policies can be considered.

– **"Allow all" policy**: The switch controller in the control plane does not wait for a detection result. Instead, the switch controller sends a packet-out message containing the "forward" instruction to the data plane immediately.

  The external IDS reports a new intrusion only when the external IDS identifies an attack. Then, the switch controller once again sends a packet-out message containing the "drop" instruction to the switch controller. This policy is helpful to reduce an initial latency of in-vehicle communication.

– **"Deny all" policy**: The switch controller does not immediately reply to all packet-in events from in-vehicle programmable switches. Instead, the switch controller waits for the detection result from an external IDS.

  When the external IDS does not find an in-vehicle intrusion, the switch controller sends a packet-out message including a "forward" instruction to the vehicle. On the contrary, when the external IDS detects an intrusion, a packet-out message will contain the "drop" instruction.

  "Deny all" is stricter and more secure policy compared to that of "allow all", in which every new packet needs to be examined by IDSs before being forwarded to an ECU.

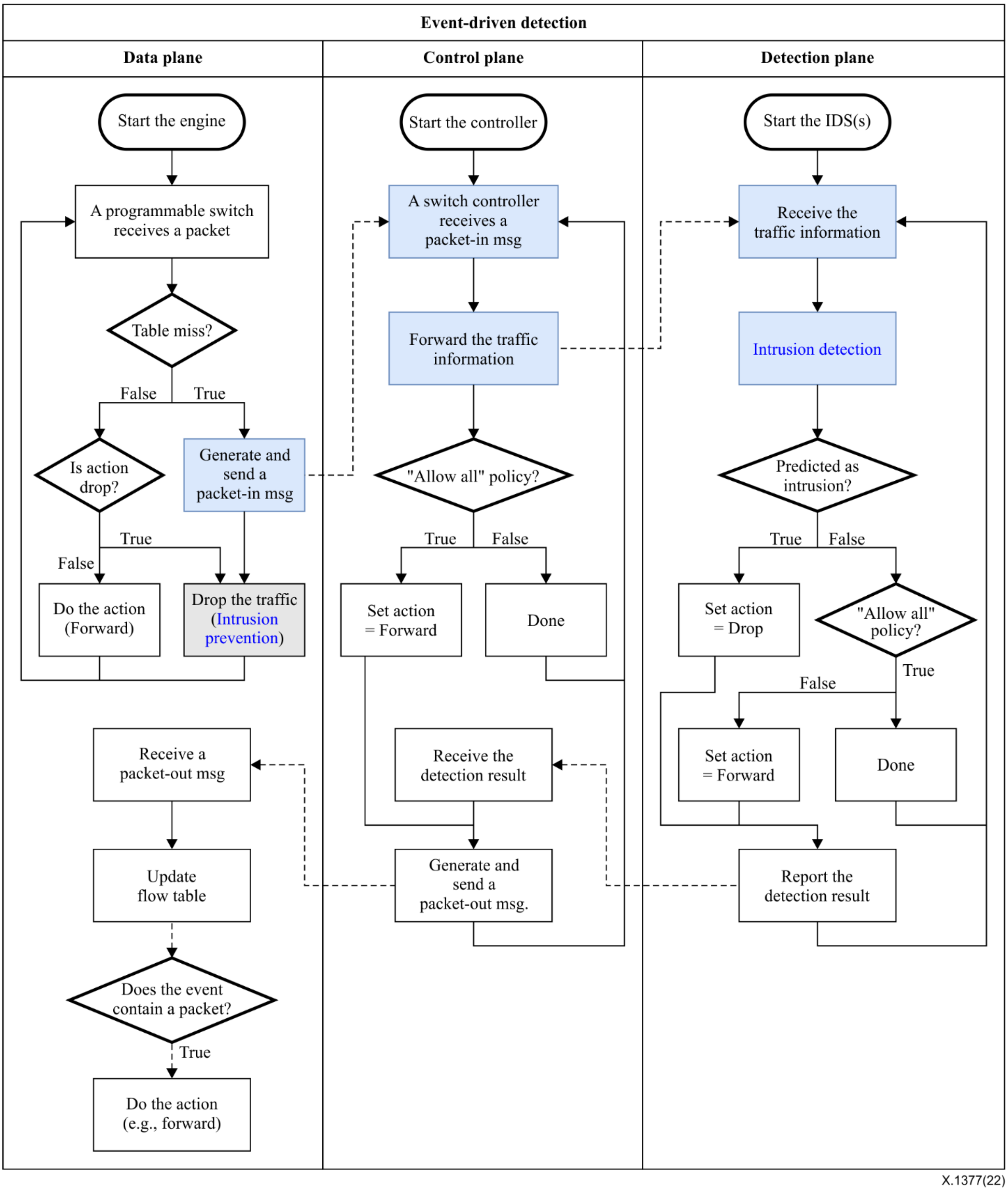


Figure 7 – Event-driven detection method

9.2.2 Data-driven detection

The data-driven method allows flow-based intrusion detection based on statistics such as packet per second and bytes per second. It is useful for detecting an adversary who transmits a bulk of packets.

Figure 8 shows how to implement the data-driven detection methodology. The solid arrow means the control flow whereas the dotted arrow means the data flow between two planes. The procedures tinted blue are the key parts of the data-driven detection methodology.

To examine statistics of IVNs, the detection plane requests traffic statistics on every interval, t. The shorter query period allows the IDS to achieve high resolution of traffic information. Also, the IDS can identify attacks more responsively. However, too short a period may cause excessive V2I communication traffic. Thus, a query interval needs to be considered carefully when the IPS is implemented. Note that a switch controller in the control plane assigns the "forward" instruction for a packet-out message regardless of the content of packet-in messages. This is because statistics can only be obtained by established communications. The detection plane returns detection results only when the external IDS detects an intrusion.

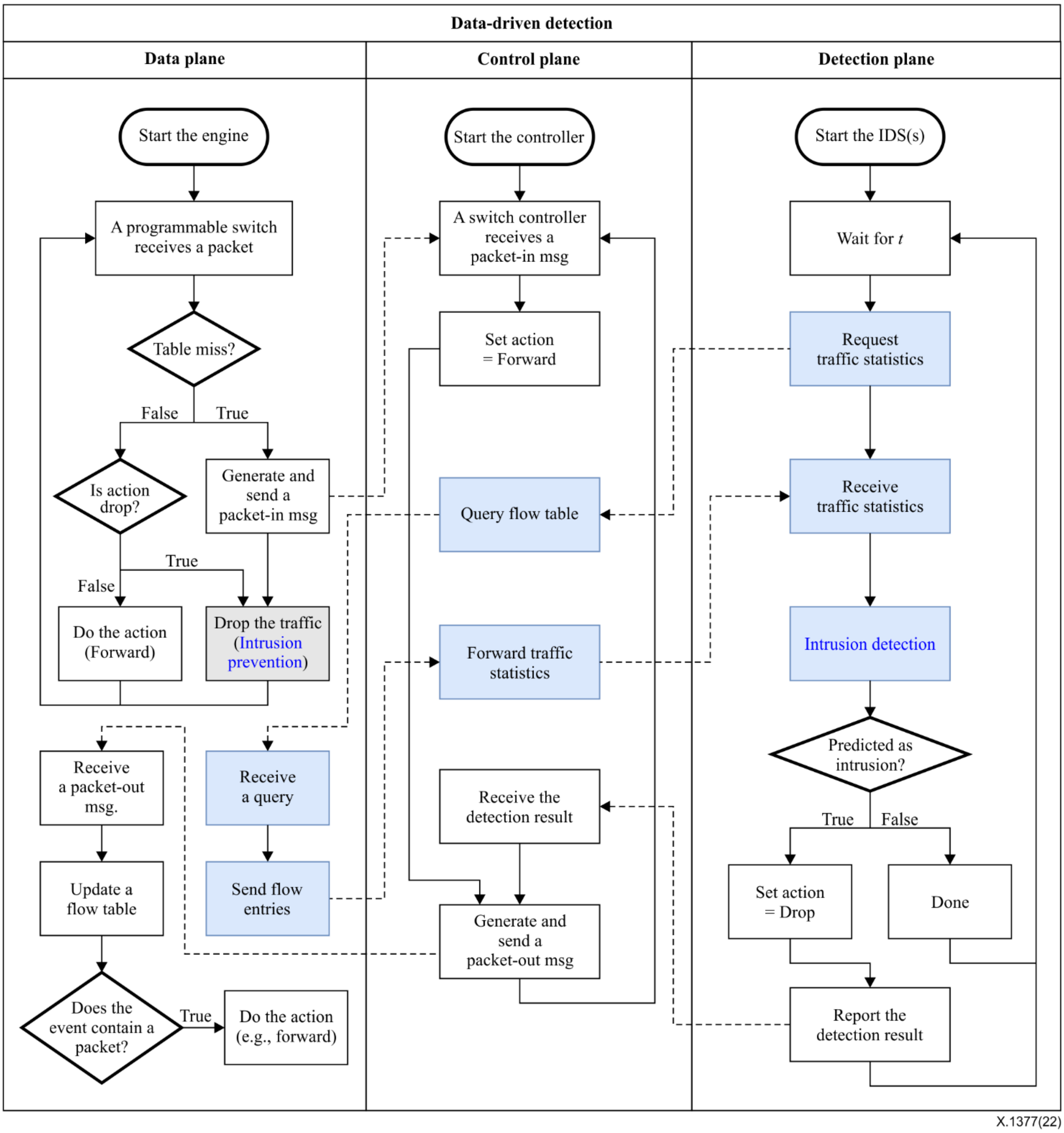

Data-driven detection
Data plane
Control plane
Detection plane
Start the engine
A programmable switch receives a packet
Table miss?
False
True
Is action drop?
Generate and send a packet-in msg
True
False
Do the action (Forward)
Drop the traffic (Intrusion prevention)
Receive a packet-out msg.
Update a flow table
Does the event contain a packet?
True
Do the action (e.g., forward)
Receive a query
Send flow entries
Start the controller
A switch controller receives a packet-in msg
Set action = Forward
Query flow table
Forward traffic statistics
Receive the detection result
Generate and send a packet-out msg
Start the IDS(s)
Wait for t
Request traffic statistics
Receive traffic statistics
Intrusion detection
Predicted as intrusion?
True
False
Set action = Drop
Done
Report the detection result
X.1377(22)

Figure 8 – Data-driven detection method

In the detection plane, the value t is the statistics refresh interval

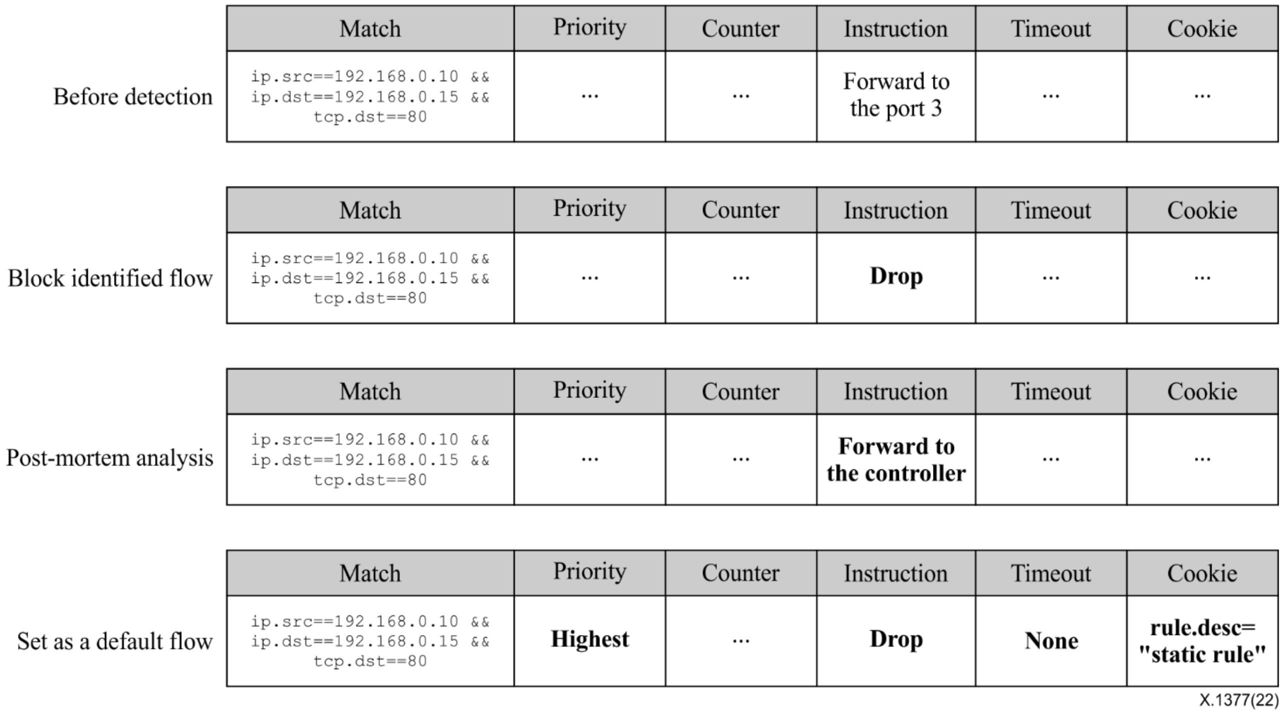
9.2.3 Payload-based detection

If necessary, the IPS can provide raw packet payloads to the detection plane for deep packet inspection. Payload-based detection is helpful in the diagnosis of the root cause of an unknown vulnerability.

In the detection plane, remote experts can watch all payloads in a suspicious flow by installing a flow entry with the instruction – a list of "forward to a designated port" and "forward to the controller". As a result, packets in the suspicious flow are delivered to its destination and the detection plane.

9.3 Methodologies of intrusion prevention

Intrusion can be prevented by changing the value of the field "instruction" of the flow entry. Figure 9 shows three specific methods of intrusion prevention.



	Match	Priority	Counter	Instruction	Timeout	Cookie
Before detection	ip.src==192.168.0.10 && ip.dst==192.168.0.15 && tcp.dst==80	...	...	Forward to the port 3	...	...

	Match	Priority	Counter	Instruction	Timeout	Cookie
Block identified flow	ip.src==192.168.0.10 && ip.dst==192.168.0.15 && tcp.dst==80	...	...	**Drop**	...	...

	Match	Priority	Counter	Instruction	Timeout	Cookie
Post-mortem analysis	ip.src==192.168.0.10 && ip.dst==192.168.0.15 && tcp.dst==80	...	...	**Forward to the controller**	...	...

	Match	Priority	Counter	Instruction	Timeout	Cookie
Set as a default flow	ip.src==192.168.0.10 && ip.dst==192.168.0.15 && tcp.dst==80	**Highest**	...	**Drop**	**None**	**rule.desc= "static rule"**

Figure 9 – Example of applying three intrusion prevention methods

9.3.1 Block identified flow

Regardless of the type of attack, a simple response method for identified traffic is to block packets of a specific flow after intrusion detection, which modifies an instruction for identified flow entry as "drop" in the in-vehicle programmable switch. After modification of the instruction, the intrusive traffic cannot be transferred by IVNs of connected vehicles.

9.3.2 Post-mortem analysis

After blocking identified flows, in-vehicle programmable switches can keep track of trends of ongoing attacks because the traffic still matches flow entries, which increases corresponding counters. Through the data-driven detection method, external IDSs can monitor the flow table to see whether the attack is still happening or has stopped.

Furthermore, an in-vehicle programmable switch can forward the identified traffic to the external switch controller and further external IDSs for a post-mortem analysis of the intrusion. The incident response team easily collects data regarding the identified attacks without any effect on the target connected vehicles. This allows them to perform deep packet inspections and network forensics and determine the root causes of attacks. They can then build and deploy some countermeasures within the IDS.

9.3.3 Set a default flow

If a security operation centre identifies that the same types of attack are happening consistently on many vehicles, a default flow for blocking them can be considered so to be installed as soon as a vehicle engine is started. Therefore, intrusive traffic will not affect IVNs in this method because in-vehicle programmable switches drop the traffic immediately. Installing a default flow can be useful until a vulnerable component in the vehicle is updated with the latest version of the security software.

9.4 Secure implementation of an intrusion prevention system

This clause establishes guidelines for the secure implementation of an IDPS for connected vehicles.

9.4.1 Access control to the external components

When operating the IPS, the external components have control of the configurations of in-vehicle programmable switches. Thus, attackers who want to compromise a connected vehicle can achieve their objectives by compromising external components instead of the connected vehicle. Therefore, external component must prevent arbitrary physical and remote access by unauthorized users.

9.4.2 Secure V2X communication

The data plane and the control plane are connected via the Internet. Hence, messages between in-vehicle programmable switches and the switch controllers should be transmitted using a secure communication protocol, like transport layer security. Also, both planes should support authentication of the communications (discussed in clause 7.3.4) so that it is not modified accidently or intentionally during transmission.

9.4.3 Personally identifiable information protection on external components

The in-vehicle traffic information, other driving-related information or any incident reports should not be revealed to unauthorized parties in order to protect privacy of drivers. A database of the detection plane should record which data was viewed by whom and when, and to check the records for future audits.

9.4.4 Requirements for an in-vehicle programmable switch

An in-vehicle programmable switch cannot work as expected in some scenarios. To keep availability of connected vehicles and IVNs, an in-vehicle programmable switch should work as an ordinary switch in the following situations:

- initializing in-vehicle components: starting the engine, rebooting after firmware updates;
- unexpected disconnections from V2X communications: out of service in rural areas, jamming attack;
- switch failure: out-of-order of an in-vehicle programmable switch, physical damage due to a traffic accident;
- cyberattacks on the programmable switch: an in-vehicle programmable switch could be out of order caused by a flow table overflow attack, e.g., an attacker could spoof various flows by modifying source IP address, destination IP address, and port number and floods such traffic to the switch.

Fail-safe operations must be considered and implemented properly. One of the following two options can be considered.

1) An in-vehicle programmable switch may operate as a normal switch that learns the media access control (MAC) addresses of Ethernet-based ECUs.

2) An in-vehicle programmable switch may refer to a pre-defined static MAC address table that does not allow a MAC address-learning feature. In this fail-safe operation, the connected vehicle provides a limited service until the fail-safe operations disengage.

Appendix I

Use-case scenario

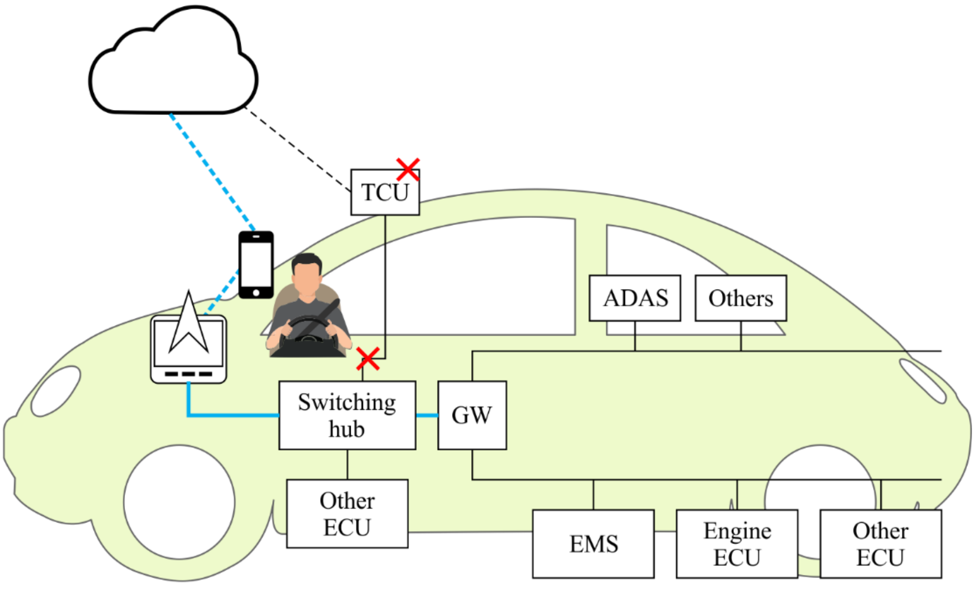
(This appendix does not form an integral part of this Recommendation.)

Use-case 1: Telematics control unit is compromised

See Figure I.1. It is assumed that the driver's smartphone and infotainment unit have been paired in advance. Under this assumption, the IDS detects that the telematics control unit (TCU) tried to compromise the message between the cloud and the IVN. After detection, the following procedure can be considered.

Steps:

1) The switching hub cuts off the connection between the TCU and cloud, and the TCU and the switching hub.
2) The driver's smartphone and the infotainment unit request that communication be mediated between the cloud and the IVN instead of the TCU.
3) The driver is requested to stop at the nearest car shop for repair.



X.1377(22)

Figure I.1 – Procedure when a telematics control unit is compromised

Use-case 2: Advanced driver assistance system unit is compromised

See Figure I.2. The IDS detects that the ADAS unit has tried to send illegal messages to the IVN. After detection, the following procedure can be considered.

Steps:

1) The connection between the ADAS unit and the IVN is cut off or the ECU mode is changed to safety depending on the automotive safety integrity level. Such a disconnection can be performed by modification of a flow table in the switching hub.
2) The driver is immediately requested to switch to manual driving.
3) The driver is requested to stop at the nearest car shop for repair.

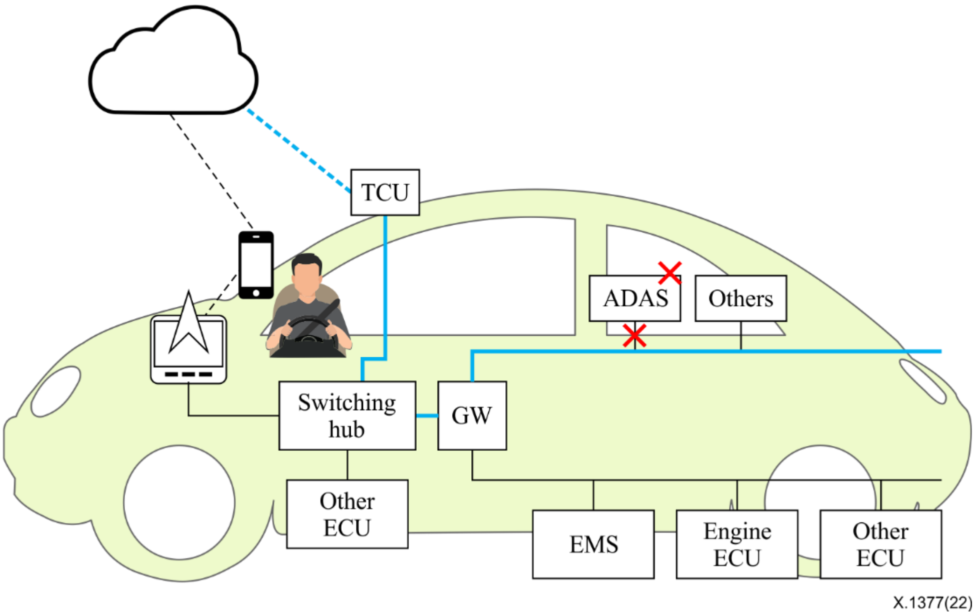


Figure I.2 – Procedure when an advanced driver assistance system unit is compromised

Use-case 3: Accelerator engine management system is compromised

See Figure I.3. The IDS detects that the engine management system (EMS) has tried to send illegal messages. After detection, the following procedure can be considered.

Steps:

1) The control plane requests the in-vehicle programmable switch to overwrite the illegal messages from the EMS with the error frame.

2) The switching hub and the GW are requested to mediate communication between the cloud and the IVN.

3) A message to the engine ECU to slowly reduce the speed is sent.

4) The car is requested to stop on the shoulder of the road and road service is called to help the vehicle.

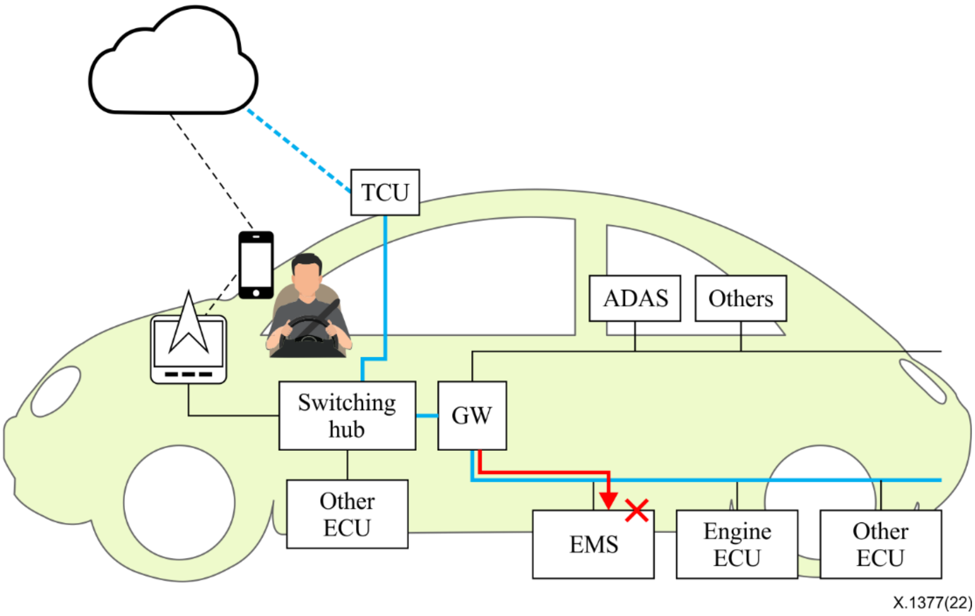


Figure I.3 – Procedure when an engine management system is compromised

Use-case 4: Secure in-vehicle firmware upgrade procedure

See Figure I.4. The central upgrade server, responsible for authorization, verifies that firmware for installation in a connected vehicle has the correct signature, is run by a legitimate user, and is the appropriate version of the software. The detection plane can monitor and control any flow required for the firmware upgrade. Consequently, the IPS can support legitimate in-vehicle firmware upgrades. The possible secure firmware upgrade steps follow.

Steps:

1) An authorized mechanic declares to the central upgrade server that a new firmware upgrade is about to start for a connected vehicle.
2) The authorized mechanic starts the engine and initializes a firmware upgrade procedure. DoIP packets are injected through the OBD-II port.
3) The data plane (i.e., in-vehicle programmable switch) informs the detection plane of the occurrence of new DoIP packets and asks how to handle the traffic. Meanwhile, the connected vehicle does not respond to wire or wireless diagnostic devices.
4) The detection plane communicates with the central upgrade server and computes intrusion detection algorithms.
5) The central upgrade server returns a message, "A firmware upgrade is scheduled at this time."
6) The detection plane generates a new flow entry (i.e., forward DoIP packets) and sends it to the data plane.
7) The DoIP packets are transmitted to the target ECU after the installation of the flow in an in-vehicle programmable switch. A firmware upgrade will soon be in progress.

If the central upgrade server does not recognize the upgrade plan or intrusion detection algorithms identify an intrusion, the detection plane generates another flow that contains the "drop DoIP packets" action. In this case, an in-vehicle ECU will not be affected by an attacker who spoofs the firmware upgrade procedure.

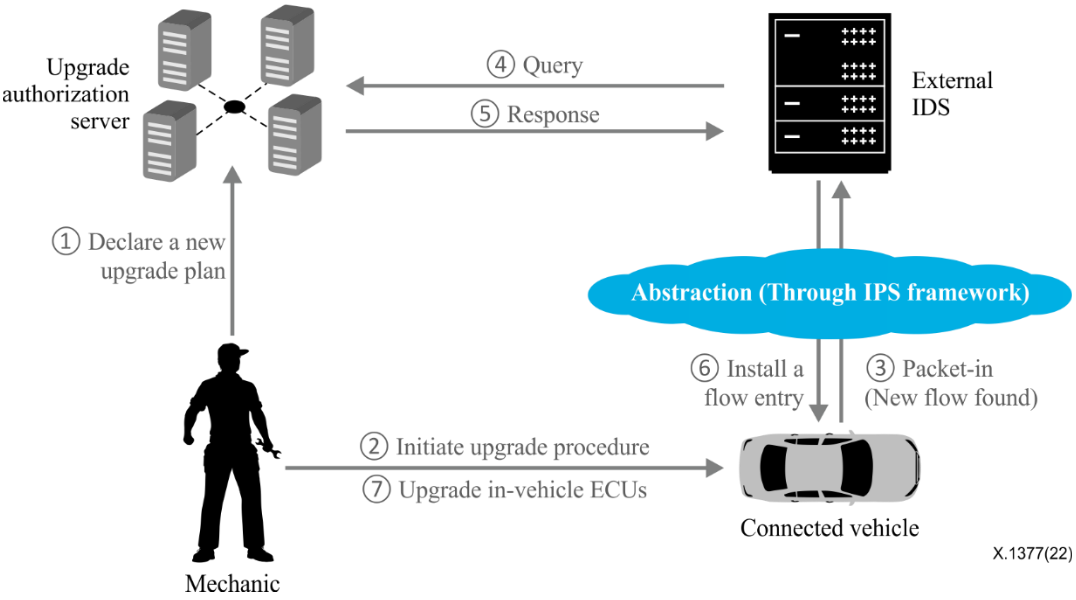

Upgrade authorization server
④ Query
⑤ Response
External IDS
① Declare a new upgrade plan
Abstraction (Through IPS framework)
⑥ Install a flow entry
③ Packet-in (New flow found)
② Initiate upgrade procedure
⑦ Upgrade in-vehicle ECUs
Connected vehicle
Mechanic
X.1377(22)

Figure I.4 – Procedure for in-vehicle electronic control unit firmware upgrade

Use-case 5: Incident response for connected vehicles

See Figure I.5. Assume that a new malicious bot that triggers a new vulnerability on an infotainment device of connected vehicles becomes widespread. The vulnerability remains in connected vehicles until drivers respond to the manufacturer's recall. The IPS can secure vulnerable connected vehicles against such situations by the following procedure.

Steps:

1) An attacker tries to compromise many connected vehicles with a zero-day exploit.

2) Each in-vehicle programmable switch sends a packet-in message. The external IDS stores all packet-in events in the database.

3) At this moment, an external IDS cannot detect the intrusion. The detection plane generates a new flow entry (i.e., forward the packet) and sends it to the data plane.

4) At the external detection plane, an incident response team identifies there is a recurrence of harmful traffic patterns from numbers of connected vehicles.

5) The incident response team devises a new rule or signature to filter traffic patterns.

6) In the data plane, the attacker tries to compromise another connected vehicle by sending malformed in-vehicle traffic.

7) An in-vehicle programmable switch in another vehicle sends a packet-in message to the external IDS.

8) The external IDS can detect the zero-day exploit. The detection plane generates a new flow entry (i.e., drop the packet) and sends it to the data plane. Now each vehicle is secure.

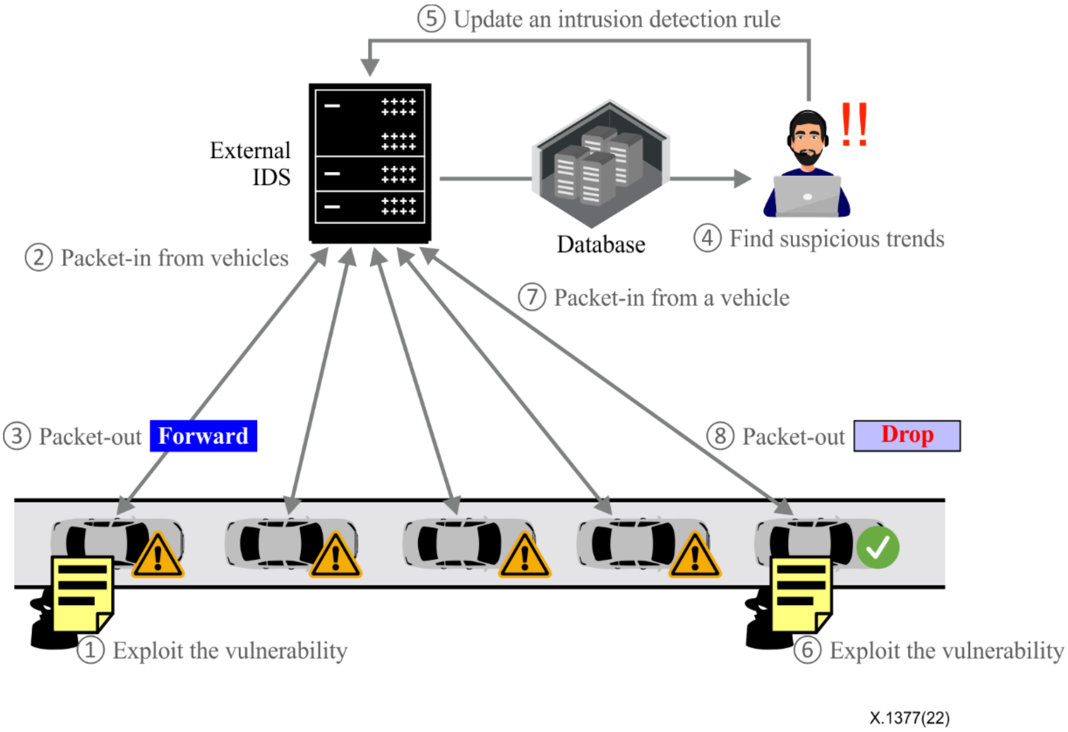

⑤ Update an intrusion detection rule
External IDS
Database
!!
④ Find suspicious trends
② Packet-in from vehicles
⑦ Packet-in from a vehicle
③ Packet-out Forward
⑧ Packet-out Drop
① Exploit the vulnerability
⑥ Exploit the vulnerability
X.1377(22)

Figure I.5 – Procedure for incident response for connected vehicles

Appendix II

Two intrusion detection systems for in-vehicle network security

(This appendix does not form an integral part of this Recommendation.)

This appendix gives brief information about two IDSs that can be used to identify network-level threats in IVNs. [ITU-T X.1375] focuses on detecting intrusion and malicious activities in various types of IVN, such as CAN and FlexRay, that cannot be covered by general IDSs designed for the Internet. Thus, the guidelines can be applied to almost any vehicle. [ITU-T X.1375] provides guidelines for various detection methodologies and implementation of IDSs. The in-vehicle IDS has a simple structure and is thus relatively easy to implement. The in-vehicle IDS performs traffic collection, analysis, intrusion detection and saving the detection result on its own inside the vehicle. However, it does not consider intrusion prevention due to the characteristics of communication protocol specifications.

Connected vehicles are highly encouraged to adapt the (automotive) Ethernet for IVN because of the network bandwidth required by high-definition applications. However, general IDSs may not be useful, even connected vehicles use the Ethernet and IP as primary protocols, due to the lack of consideration of vehicular-specific communications and environments. This Recommendation provides methodologies for IPSs on connected vehicles to detect intrusions in Ethernet-based IVNs and block and post-analysis intrusions. To prevent intrusions in connected vehicles, original equipment manufacturers need to follow appropriate IDS design and detection strategy. Thus, two types of IDS for vehicles are introduced in [ITU-T X.1375] and this Recommendation.

Table II.1 compares two IDSs discussed in [ITU-T X.1375] and this Recommendation. A description of each field in Table II.1 follows.

- **Location of IDS**: Location of the system evaluating intrusion detection within the IVN.
- **Data capture point**: Point at which to collect data to be used as input to intrusion detection algorithms.
- **Input data type**: The type of data that will be used as input to the intrusion detection algorithm.
- **Destination of intrusion detection results**: Location at which the output of the intrusion detection algorithm is stored or utilized.
- **Supporting traffic aggregation over vehicles**: Whether traffic from multiple vehicles can be utilized.
- **V2X communication required**: Whether V2X communication is required for intrusion detection (and prevention).

Table II.1 – Comparison of two types of IDSs discussed in [ITU-T X.1375] and this Recommendation

	[ITU-T X.1375]	This Recommendation
Location of IDS	Inside a vehicle	Outside a vehicle
Data capture point	In-vehicle gateway, ECU	In-vehicle programmable switches
Input data type	CAN messages	Address resolution protocol, TCP, UDP, Internet control message protocol, AVTP, DoIP, etc.
Destination of intrusion detection results	Security operations centre	Security operations centre and programmable switches
Supporting traffic aggregation over vehicles	No (single vehicle)	Yes (multiple vehicles if necessary)
V2X communication required	No	Yes

Bibliography

[b-ITU-T E.417]	Recommendation ITU-T E.417 (2005), *Framework for the network management of IP-based networks*.
[b-ITU-T X.800]	Recommendation ITU-T X.800 (1991), *Security architecture for Open Systems Interconnection for CCITT applications*.
[b-ITU-T Y.2770]	Recommendation ITU-T Y.2770 (2012), *Requirements for deep packet inspection in next generation networks*.
[b-IETF RFC 5101]	IETF RFC 5101 (2008), *Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information*.
[b-ISO/IEC 27000]	ISO/IEC 27000:2018, *Information technology – Security techniques – Information security management systems – Overview and vocabulary*.
[b-ISO/IEC 27039]	ISO/IEC 27039:2015, *Information technology – Security techniques – Selection, deployment and operations of intrusion detection and prevention systems (IDPS)*.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	**Data networks, open system communications and security**
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2022