oversee

SEVENTH FRAMEWORK PROGRAMME

Project acronym:	OVERSEE
Project title:	Open Vehicular Secure Platform
Project ID:	248333
Call ID:	FP7-ICT-2009-4
Programme:	7th Framework Programme for Research and Technological Development
Objective:	ICT-2009.6.1: ICT for Safety and Energy Efficiency in Mobility
Contract type:	Collaborative project
Duration:	01-01-2010 to 30-06-2012 (30 months)

Deliverable D2.4:
Specification of Secure Communication

Authors:	Rafael Grote (TU Berlin) Florian Friederici (Fraunhofer FOKUS) Jan Holle (Uni Siegen) André Groll (Uni Siegen) Hakan Cankaya (escrypt) Thomas Enderle (escrypt)
Reviewers:	Alfons Crespo (UPVLC) Thomas Enderle (escrypt)

Dissemination level:	Public
Deliverable type:	Report
Version:	1.7
Submission date:	11 November 2013

Abstract

This document examines all internal and external types of communication within the OVERSEE platform. Each internal communication path is analysed and appropriate security measures are proposed. A security parameter management scheme is defined considering the restrictions of automotive applications and the special needs of ITS communication networks. For remote access to the OVERSEE platform, a remote diagnosis interface as well as a remote connection facility is designed.

Contents

List of Figures

List of Acronyms and Abbreviations

2G	Second generation mobile phone system
3G	Third generationmobile phone system
API	Application Programming Interface
BT	Bluetooth
CALM	Communications access for land mobiles
CAM	Cooperative Awareness Message
CAN	Controller–area Network
CEN	European Committee for Standardization
CPU	Central Processing Unit
DENM	Decentralized Environmental Notification Message
DNS	Domain Name System
DoS	Denial of Service
DSRC	Dedicated Short-Range Communications
eCall	Emergency Call
ECDSA	Elliptic Curve Digital Signature Algorithm
ECU	Electronic Control Unit
ETSI	European Committee for Standardization
EVITA	E-safety vehicle intrusion protected applications
FiFo	First in First out
HMI	Human Machine Interface
HSM	Hardware Security Module
HTTP	Hypertext Transfer Protocol
I/O	Input Output
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
ISO	International Organization for Standardization
ITS	Intelligent Transportation System
LAN	Local Area Network
LDM	Local Dynamic Map
NAT	Network Adress Translation
NIC	Network Interface Card

OEM	Original Equipment Manufacturer
OS	Operating System
OSEK	Offene Systeme und deren Schnittstellen für die Elektronik im Kraftfahrzeug
OVERSEE	Open Vehicular Secure Platform
PPP	Point-to-Point Protocol
PSAP	Public-Safety Answering Point
QoS	Quality of Service
RSA	Rivest, Shamir and Adleman algorithm for public-key cryptography
SSL	Secure Sockets Layer
SVAS	Secure Vehicle Access Service
TC	Technical Committee
TCP	Transmission Control Protocol
TLS	Transport Layer Security
USB	Universal Serial Bus
WEP	Wired Equivalent Privacy
Wi-Fi	Wireless Fidelity
WPA	Wi-Fi Protected Access

1 Introduction

The Open Vehicular Secure Platform (OVERSEE) project has produced this deliverable; therefore it contains contributions from all partners even if Fraunhofer FOKUS, Universität Siegen, and escrypt GmbH are the main contributors.

In task 2.5 any internal and external communication path of the OVERSEE platform as well as remote interfaces and required security parameters have been examined. Resulting from that work, this deliverable analyses possible security flaws and suggests appropriate countermeasures.

1.1 Scope and Objectives

This document aims to span the design of a client interface and protocol that is able to communicate with OVERSEE in different modes of security and communication (e.g., internal/external). Many different communication modes are discussed, e.g., in ETSI TC ITS as well as ISO CALM and are considered here. Using the communication modes for management purposes, this document identifies required security parameters (e.g., cryptographic keys), for OVERSEE operation. Further, a security parameter management scheme is designed, to bootstrap trust relationships necessary for a dependable and secure system. Bandwidth and performance restrictions in automotive applications are considered.

The security measures proposed in this document do not take possible hardware or software modifications before or during the boot process into account. Only a secured boot process, for instance based on trusted platform modules, can prevent such manipulations. There are existing solutions to this problem that platform integrators should consider. Nevertheless, a secured boot process is not in the scope of this document. It is assumed that the platform integrity is ensured by an appropriate technology at least until the system is up and running.

Security of peripheral components, e.g., human machine interfaces, audio devices, all kinds of sensors, external Bluetooth or USB devices, and network adapters are not in the scope of OVERSEE and will not be considered in this document.

1.2 Document Outline

Chapter 2 summarizes all internal communication paths, which D2.1 [1] described in detail. Possible weak points are revealed and analysed. Adequate security measures are proposed for each communication path.

Chapter 3 describes the key management facilities with regard to distribution, storage, and usage. The restrictions of automotive systems are considered. In particular, it examines the needs for key distribution in ITS.

Chapter 4 defines how the OVERSEE platform can be accessed remotely. This includes a remote diagnosis capability as well as remote access to the user interface.

2 Communication Paths

This chapter shortly summarizes the list of interfaces and paths of information flow, which were specified in D2.1 [1]. Subsequently, the following sections analyse each possible communication path and propose suitable security measures if necessary.

2.1 List of Paths

Within D2.1 [1] the communication capabilities of OVERSEE towards the environmental networks and the services towards the applications running within the OVERSEE runtime environments were defined. Based on the separate view per communication resource in that deliverable, Figure 2 presents all communication paths within OVERSEE.

For Figure 2 the following meanings of symbols and colours apply:

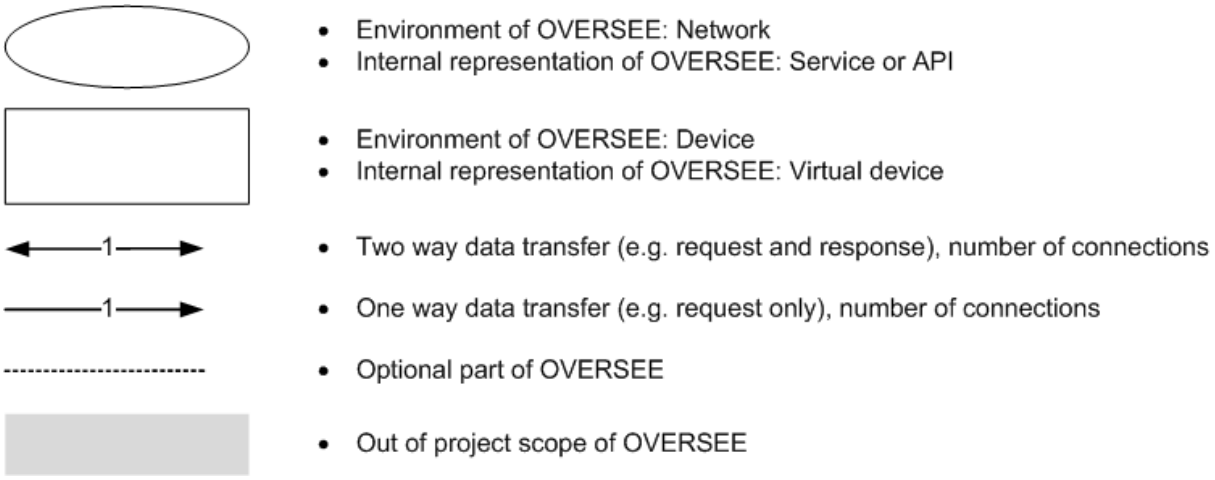

Figure 1: Key for OVERSEE communication paths

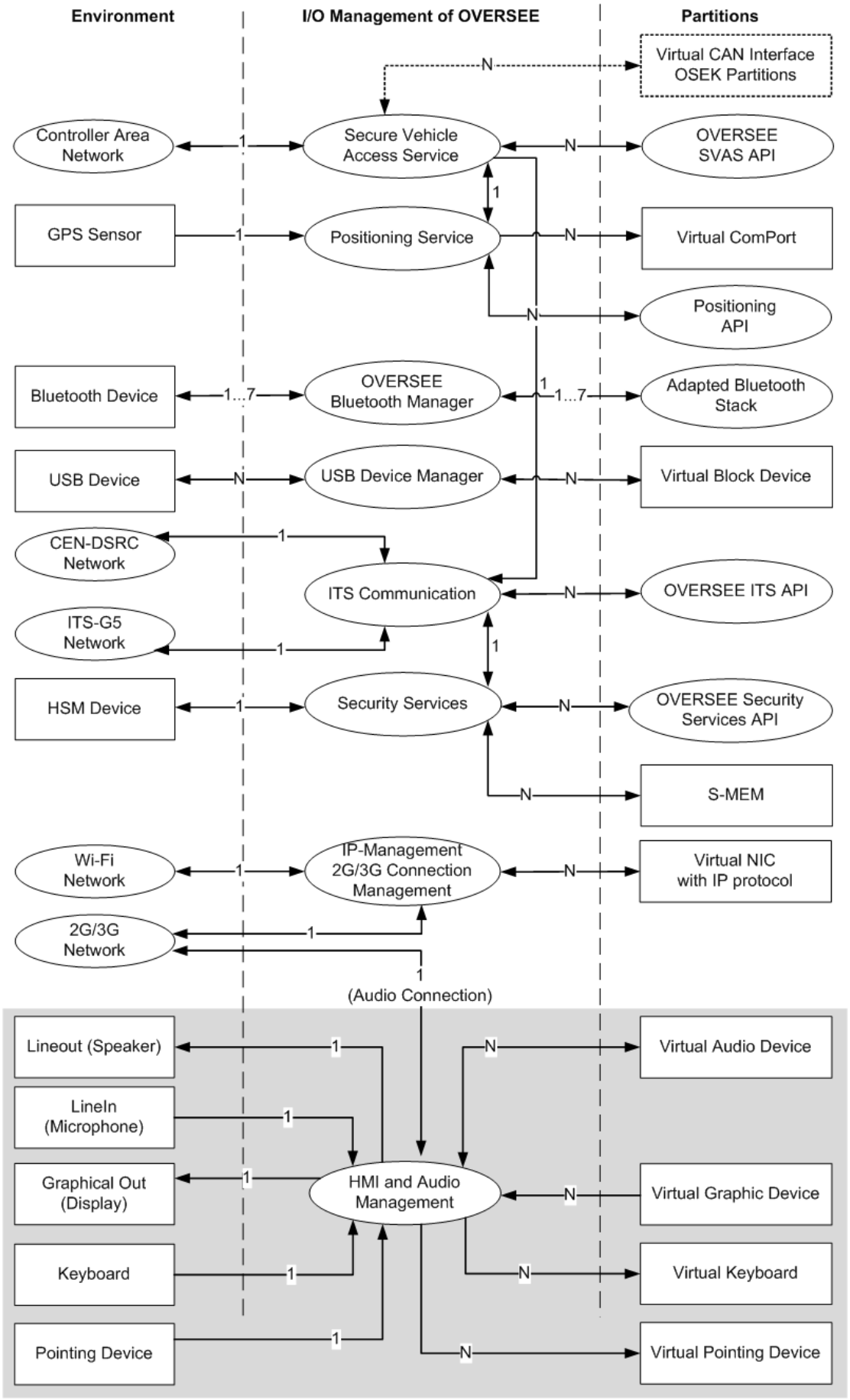

Figure 2: OVERSEE communication paths

2.2 Security Measures per Path

Each of the following sections analyses one of the internal communication paths within OVERSEE. Therefore, component by component is examined with regard to the component itself and any connection to other components. Separate sections survey the OVERSEE-internal connections between communication paths and the particular risk of these connections.

Integrators should bear in mind that a secured boot process including integrity checks is mandatory to guarantee an unmodified system. Section 2.4 in D2.2 [2] addresses that issue. Hence, we presume system integrity after boot-up here.

In the first instance, the following section takes a closer look to inter-partition communication, which is essential for each communication path but also highly security critical.

2.2.1 General Inter-Partition Communication

The OVERSEE Platform aims to provide strong runtime isolation of partitions with virtualization. In this context the hypervisor has a fundamental role in the OVERSEE platform creating virtual resources for the runtime environments. The virtual runtime environment provided by the virtualization layer in OVERSEE restricts any unauthorized access to resources and facilities not assigned to the partition.

The hypervisor in OVERSEE facilitates various methods for isolation of resources and granting access rights.

- The hypervisor provides temporal isolation by scheduling CPU resources to each partition. The developer can configure the amount of processing resource with a schedule plan thus assuring a predefined amount of processing resource for each runtime environment. Furthermore, the hypervisor also assures isolation of data transfer over the CPU among partitions and a stateless functionality of the CPU decoupling the scheduled partitions in the time domain.
- Another important feature of the OVERSEE hypervisor are dedicated memory areas for the partitions. The hypervisor assures restricted access to the addressable memory of the hardware platform. The address areas can be configured for explicit access from only one partition or can be shared among multiple partitions. Consequently, the virtualized runtime environment is only aware of assigned memory space and has no access right to unauthorized memory areas.
- The hypervisor also provides restricted access to hardware peripherals. Any hardware device has to be assigned to one partition. This partition has full access right to the peripheral whereas other partitions are not aware of this hardware device or interface. On the other hand sharing hardware devices is possible over the secure I/O partition. The secure I/O partition is granted authorization to access the peripherals as well as interfaces directly and is responsible of managing the peripherals and providing services to other partitions over the inter-partition communication facilities of the hypervisor.

- The virtual runtime environments need a secure way of communicating internally on the platform. As a solution the OVERSEE hypervisor provides secure communication channels, which can be configured by the developer. A communication channel is assigned to specific partitions with write or read rights to the communication channel, thus defining explicitly the sender and receiver using a communication channel. This assures restricted access to the communication channels by the authorized partitions, and no access to those channels by unauthorized partitions. Another aspect is the communication methods provided by the hypervisor. The first method is a broadcasting channel enabling a non-blocking communication channel. The message on the channel is available until a new message overwrites the data. The second method is a queue implementation functioning as a FiFo buffer. The messages are buffered until the receiver reads the data. The consequence is that a full buffer causes an error when attempting to write data into the communication channel.
  The communication channel also assures that the origin of data arriving at the partition over a communication channel can be securely determined.
- The communication channels provided by the hypervisor are generic channels. Any protocol or data structure for inter-partition communication will build on these channels and share the features of the OVERSEE hypervisor communication channels. The communication channels can be used to create virtual network adapters, diagnose communication, security services, etc.
- The hypervisor also provides shared memory areas among multiple partitions. The access rights to these shared memory areas are defined by the integrator as it is with the communication channels. The user partitions allowed to access the shared memory are responsible of managing the data in the shared memory in a proper way. Shared memory is usually used for creating virtual shared peripherals.
- On the other hand, a secure system and full assurance can only be provided with a secure boot process. The facilities mentioned provide secure methods for an operating system. These methods provided by the virtualization layer can be used to create a secure platform with dedicated software measures. These measures are usually not sufficient to assure a trustable platform in every case. For example, a person with direct access to the hardware platform can modify crucial data or software. Therefore, some measures to identify at least unauthorized modification have to exist on a secure platform.

2.2.2 Secure Vehicle Access Service

The secure vehicle access service (SVAS) is the communication facility between applications, executed within the OVERSEE runtime environments and the vehicle internal network. Additionally, some other services of the OVERSEE platform reuses the information gathered by the secure vehicle access service and the secure vehicle access service is also able to provide information collected via other OVERSEE communication features.

Figure 3 depicts the communication path of the secure vehicle access service (except the optional CAN binding for OSEK partitions, discussed in section 2.2.3).

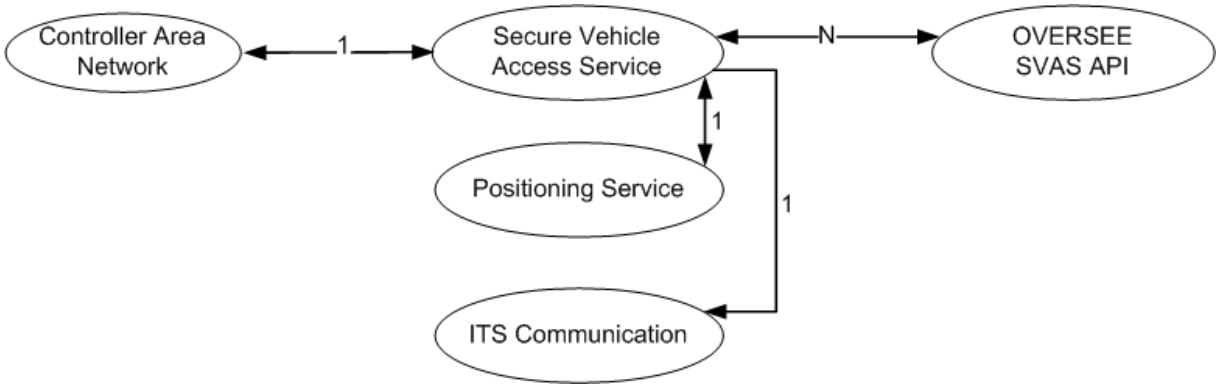

Figure 3: Secure vehicle access service communication path

The following security measures apply to the shown connections, interfaces, and components.

2.2.2.1 Controller Area Network

The controller area network (CAN) is the common vehicle internal network. Since the main focus of the network is on the reliability of the vehicle internal communication while at the same time obtaining maximum performance with cheap and simple controllers, no advanced security measures are integrated. Therefore, also the OVERSEE platform is not able to provide additional security mechanisms for CAN.

However, for example the EVITA (E-safety vehicle intrusion protected applications) project [5] aims to add advanced security mechanisms to in-vehicle communication systems. OVERSEE is able to integrate these mechanisms as soon as they are standardized due to its generic security services and the corresponding HSM.

2.2.2.2 Connection between CAN Interface and SVAS

The connection is a two way communication path, which means that messages from the controller area network are delivered to the secure vehicle access service and vice versa. The whole connection is under the responsibility of the OEM and will be implemented in the secure I/O partition, and thus is out of user space.

The following communication rules (as part of the whole security concept) apply to this connection:

- Basic rule: Don't forward any message from CAN to the secure vehicle service and vice versa.
- As an extension to the basic rule: Forward the messages from CAN to the secure vehicle access service as they are listed within the "white list for CAN read". The white list has to be defined and signed by the OEM, for the current platform configuration.
- As an extension to the basic rule: Forward the messages from the secure vehicle access service to CAN as they are listed within the "white list for CAN write". The white list has to be defined and signed by the OEM, for the current platform configuration.

For performance issues the white list for CAN read could also be compiled as a message filter into the CAN interface driver to reduce overhead and, additionally, reduce the risk of a denial of service attack from CAN against OVERSEE. Since the connection is under the responsibility of the OEM this decision belongs to him.

2.2.2.3 Secure Vehicle Access Service

The secure vehicle access service is a new service of OVERSEE. SVAS is divided in two main parts: The SVAS server (executed in the secure I/O partition) and the SVAS client (executed in the application partitions) offering the SVAS functionality to the applications via the SVAS API. The client gets or sets data objects from or to the vehicle electronics through requests to the server. Also advanced communication techniques like publish/subscribe for data objects will be available. The translation between data objects and the corresponding CAN messages or even CAN message communication procedures will be defined within the SVAS server by the OEM.

The security mechanisms are hence dedicated to restrict access to the data objects. Therefore, each data object has two access right attributes (read and write) for each connection to the SVAS server. The following rules apply within the secure vehicle access component:

- Basic rule: Don't provide any access to data objects for any incoming connection from the SVAS clients. Additionally, even hide the existence of all data objects to the incoming connections.
- As an extension to the basic rule: Show data objects and provide read access to these data objects for a specific incoming connection if, and only if, there is an item within the access control list of the SVAS granting read access to the specific data object for the specific incoming connection. The access control list for the SVAS has to be defined and signed by the OEM for the specific platform configuration.
- As an extension to the basic rule: Show data objects and provide write access to these data objects for a specific incoming connection if, and only if, there is an item within the access control list of the SVAS granting write access to the specific data object for the specific incoming connection. The access control list for the SVAS has to be defined and signed by the OEM for the specific platform configuration.

It has to be stressed that the SVAS server is only able to grant or deny access to data objects based on the incoming connections, which are assigned to specific partitions. Therefore, the security of the access control within the SVAS relies on the security and authenticity of the connections.

2.2.2.4 Connection between SVAS Server and SVAS Client

The connections between the SVAS Server and the SVAS clients will be according to [1] based on the virtual NICs and IP. The connections between the virtual NICs will be implemented by use of XtratuM internal communication techniques (shared memory). Since the communication will be conducted out of user space, no additional security mechanisms are necessary for this connection. The only security concern is about authenticity of the

whole connection, from the partition to the SVAS server. This authenticity has to be ensured by the OVERSEE platform. Therefore, the following rule applies:

- Provide authenticity for the whole connection between the partition and the SVAS server through the whole communication stack, including the chance to identify which connection to the SVAS server belongs to which partition for the access control component within the SVAS server.

This rule still applies in the case that the communication between SVAS server and SVAS client will be changed to the native mode, as proposed in the annex of [1]. Since this type of implementation will avoid the additional IP layer and the use of virtual NICs, it should be indeed much easier to ensure the authenticity of the whole connection.

2.2.2.5 SVAS Client

The SVAS client will be executed within the partitions and therefore in user space. Hence, it is in an untrustworthy environment and OVERSEE will not assume any additional security attributes of the SVAS client. Nevertheless, it is up to the responsible organization of the partition to decide which application or user is allowed to use the SVAS client. However, since this is out of the scope and surveillance of OVERSEE, this aspect is not considered any further. **Please keep in mind: OVERSEE ensures security on partition level!**

2.2.2.6 Connection between SVAS and Positioning Service

The security issues of this additional connection, which is provided for ease of use for the application developers, will be treated in section 2.2.5.

2.2.2.7 Connection between SVAS and ITS Communication

The security issues of this additional connection, which is provided for ease of use for the application developers, will be treated in section 2.2.9.

2.2.2.8 Residual Risk for the Secure Vehicle Access Service Communication Path

Concerning the whole communication path from SVAS API clients, executed in the partitions, to the hardware CAN module, the following residual risks have to be assessed and as far as possible reduced within the implementation tasks:

- The security of the CAN access directly relies on the implementation between the SVAS server and the CAN module. If read and write access to the CAN bus is conducted with the same CAN module and possibly also the same driver there is the risk of a successful attack from another domain.
- The access control mechanisms of SVAS rely on the authenticity of the underlying connections, provided by OVERSEE and in particular XtratuM. If an attacker succeeds in breaking the security features of the internal communication stack, all security functions of the SVAS are lost.

2.2.3 Optional CAN Interface for OSEK Partitions

The OSEK operating system is a well-established OS in the automotive domain, especially due to its strong relation to AUTOSAR. Since OSEK is dedicated to real time use cases, it has strong requirements concerning the timely communication. Additionally, OSEK and all resources will be compiled and linked together with the final application. Therefore, the flexible and secure connection to the vehicle electronics, as discussed within section 2.2.2, is probably too costly in terms of communication time and implementation overhead. Hence, we suggest an alternative based on a virtual CAN interface in the OSEK partition. This advanced approach is optional and no part of the generic OVERSEE implementation in the current project. It is up to the integrator of a real word OVERSEE ECU to integrate this option, it is strongly recommended to consider the following security design decisions.

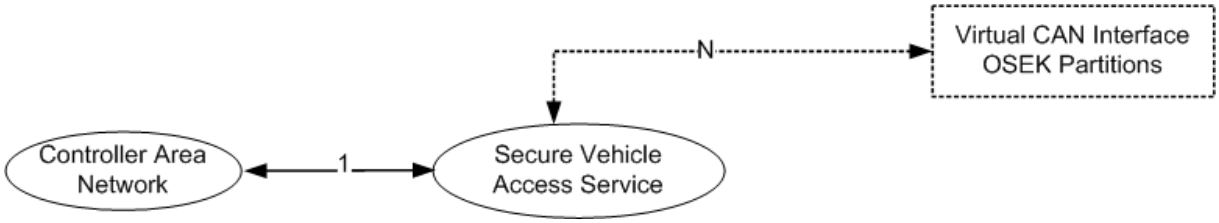

Figure 4: CAN access for OSEK partitions communication path

2.2.3.1 Controller Area Network

Please look up section 2.2.2.1.

2.2.3.2 Connection between CAN Interface and SVAS

Please look up section 2.2.2.2.

2.2.3.3 Secure Vehicle Access Service

Basically, there are two options how to integrate the additional interface for the access from OSEK partitions to the SVAS:

The straightforward idea would be to build a virtual collection of data objects on top of the SVAS server, reusing the access control mechanism described in the section 2.2.2.3, and providing the resulting data objects as CAN messages towards the specific OSEK partition.

- If the interface for the virtual CAN interface in OSEK partitions will be implemented on top of the SVAS server, all security rules as defined within section 2.2.2.3 apply.

Unfortunately, this approach would be probably too costly in terms of communication time and computing overhead in order to fulfil the real time requirements of applications executed in OSEK partitions.

The second approach is an additional entity within the SVAS based on the same connection to the CAN bus; as the SVAS sever, but transmitting the CAN messages in original form between the CAN bus and the virtual CAN interface in the OSEK partition. While this implementation would offer probably much more performance, it also requires an additional

set of policies to restrict the CAN bus access in a partition specific manner. Additionally, the abstraction layer between CAN and the applications is lost, which could lead to non-vehicle independent applications. Hence, this approach should be used only if necessary. If the described implementation is used the following rules apply:

- Basic rule: Don't provide any access to CAN messages for any incoming connection from virtual CAN interfaces in OSEK partitions.
- As an extension to the basic rule: Forward CAN messages to a specific incoming connection from a virtual CAN interface in an OSEK partition if, and only if, there is an item within the access control list for CAN of the SVAS granting read access to the specific CAN message for the specific partition. The access control list has to be defined and signed by the OEM for the specific platform configuration.
- As an extension to the basic rule: Forward CAN messages from a specific incoming connection (linked to an virtual CAN interface in an OSEK partition) to the CAN bus if, and only if, there is an item within the access control list for CAN of the SVAS granting write access to the specific CAN message for the specific partition. The access control list has to be defined and signed by the OEM for the specific platform configuration.

2.2.3.4 Connection between SVAS and Virtual CAN Interface in OSEK Partitions

The connection will be conducted by the use of XtratuM queuing ports [1]. Hence, the common security issues for this connection type (cf. section 2.2.1) apply, especially in terms of authenticity of the connection.

2.2.3.5 Virtual CAN Interface in OSEK Partitions

Since the virtual CAN interface driver will be executed in user space, there are no special security assumptions concerning this component. Since the resources, the OS, and the application are compiled together, the responsible organization would probably examine the specific partition very thoroughly.

2.2.3.6 Residual Risk for the Optional CAN Interface for OSEK Partitions

Since the virtual CAN interface for OSEK partitions is only an optional concept for OVERSEE, there is no statement regarding the residual risks in detail. Nevertheless, in general the results of the risk analysis for the SVAS in section 2.2.2.8 apply. However, there could be additional security risks, which are strongly depending on the selected implementation type as described in section 2.2.3.3.

2.2.4 Positioning Service

The positioning service receives information on the geographical position (longitude, latitude, altitude) from a GPS, Galileo, or similar sensor. Alternatively – if no positioning sensor is available, positioning data may be obtained from the secure vehicle access service. Positioning data is delivered to privileged user partitions through virtual COM ports ensuring

compatibility to native GPS reception or the positioning API. Figure 5 shows an overview to the communication path of the positioning service.

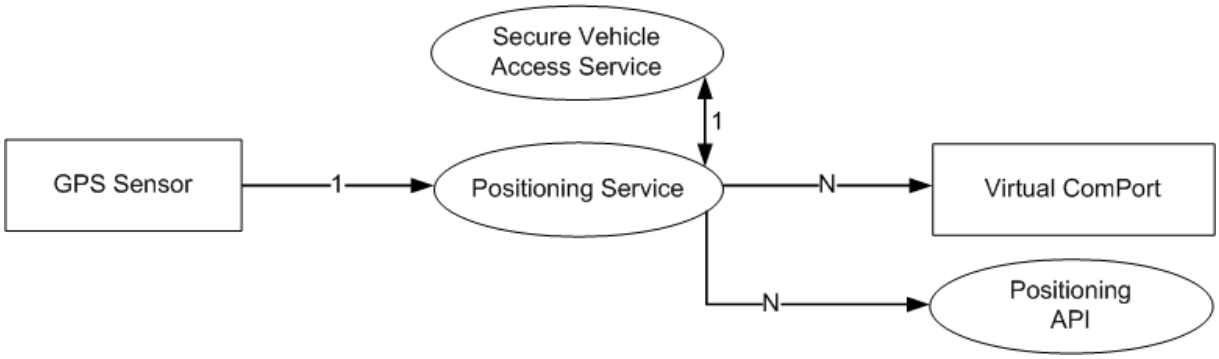

Figure 5: Communication path of the positioning service

The data transmission in positioning systems is purely unidirectional. Data can only be read and is never written. Therefore, primarily security measures for the process of receiving data are necessary. Nevertheless, the trustworthiness of positioning data has to be ensured to avoid manipulation through attackers or malicious partitions. Possible targets of an attacker might be the integrity, availability, or reliability of the inter-partition communication channels, the positioning sensor, the positioning service, or the user partition itself.

2.2.4.1 Positioning Sensor

The positioning sensor is a trusted component without need of further internal security measures. The positioning system may define its own security measures that the sensor should fulfil.

2.2.4.2 Connection between Positioning Sensor and Positioning Service

This connection is unidirectional. Sensor data is delivered exclusively to the positioning service. As long as integrity and dependability of the connection is ensured, no further security measures are required.

2.2.4.3 Positioning Service

Manipulation of this component from another partition is prevented through isolation, a trusted boot process and integrity checks.

Authorization of user partitions in order to receive positioning data enforces confidentiality and privacy. This security measure ensures privacy by preventing position tracking through unprivileged partitions, e.g., infotainment partitions that possibly may run malicious applications. Access to positioning data can only be granted to entire partitions – not to individual applications. Consequently, an authorized partition should contain only applications with the same level of trust regarding the positioning capability.

There are only two access modes to positioning data:

- Basic rule: No access. Do not forward any data to any partition without specific authorization.

- Full read-access. Forward positioning information to virtual COM ports and positioning API of authorized user partitions.

Graduated access (e.g., by narrowing the precision of locations) is not in the scope and will not be provided by OVERSEE.

2.2.4.4 Connection between Positioning Service and Virtual COM Ports / Positioning API

Connections to virtual COM ports and the positioning API are unidirectional. Data is provided by the positioning service and consumed by applications through virtual COM ports or the positioning API. It must be ensured that only authorized partitions may receive positioning data, as defined before.

The integrity and confidentiality of communication channels between positioning service and user partitions is examined in section 2.2.1 where appropriate security measures are proposed. Hence, it is presumed that these channels are secure and can neither be manipulated nor intercepted.

2.2.4.5 Virtual COM Port / Positioning API in the User Partitions

Security inside user partitions is not in the scope of OVERSEE and depends on the trustfulness of installed applications. It is the duty of system integrators to ensure security inside user partitions including integrity of virtual COM ports, positioning API and delivered data.

2.2.4.6 Residual Risk for the Positioning Service Communication Path

Application developers should be aware that any malicious application inside the same partition could possibly manipulate data provided by virtual COM ports or the positioning API. Integrators may minimize this risk by using adequate privileges for applications and services. Still, security inside user partitions depends on the installed system and applications.

Since a large number of applications use positioning data, the risk of indirect access to the data is omnipresent. Integrators should strictly separate applications with varying access privileges. However, applications could take conclusions to a vehicle's position based on data from its environment. For example, ITS applications could use locations of neighbouring vehicles to guess their own location.

2.2.5 Connection between SVAS and Positioning Service

The direct connection between secure vehicle access service (SVAS) and positioning service (as shown in Figure 3 and Figure 5) provides access to positioning data regardless of the available sources (CAN or direct sensor data). Additionally, application developers must only access a single service (namely SVAS) in order to read positioning and other sensor values from the vehicle bus.

Of course, these enhancements of availability and usability should decrease under no circumstances the system's security. Hence, the interface between both services must be as minimal and simple as possible. Exchange of any data different from position information must be excluded by design. Neither the secure vehicle access service nor the positioning service can write any data to its counterpart or rather read anything else than positioning data.

Data is only taken from the connection if the actual service does not have access to a positioning sensor by itself. The integrator should disable the connection partially or completely if one or both of the services have access to positioning sensors. Consequently, data of existing positioning sensors is never overwritten. Data exchange on this connection follows the set of rules given below:

- Basic rule: Do not exchange any data.
- As an extension to the basic rule: If neither the SVAS nor the positioning service has access to any positioning sensors, do not exchange any data and disable this connection.
- As an extension to the basic rule: If both, the SVAS and the positioning service, have access to their own positioning sensors, do not exchange any data and disable this connection.
- If the SVAS has no access to a positioning sensor, the positioning service should forward positioning data to this interface. The SVAS should read the provided information and take it as a replacement for its missing sensor.
- If the positioning service has no access to a positioning sensor, the SVAS should forward positioning data to this interface. The positioning service should read the provided information and take it as a replacement for its missing sensor.

Further, it has to be ensured that the link interface cannot be accessed by any other partition, in particular not by a user partition.

2.2.5.1 Residual Risk

As every new access point to a system opens potentially new security gaps, the risk of implementation errors is increased by this connection. The option of disabling the connection for the case of existing positioning sensors minimizes the residual risk.

2.2.6 Bluetooth

Bluetooth is the standard for wireless peripherals.

Figure 6: Bluetooth communication path

2.2.6.1 Connection between Bluetooth Interface and OVERSEE Bluetooth Manager

Access to the Bluetooth hardware interface is directly routed to the secure I/O partition by the hypervisor over a secure channel as described in 2.2.1.

2.2.6.2 OVERSEE Bluetooth Manager

The Bluetooth stack is devided: In the Secure I/O partition, the OVERSEE Bluetooth Manager, basically the lower part of a Bluetooth stack, will handle all the lower level functions, e.g. discovery, pairing etc. The manager will then offer a special interface to the user partitions, while controlling access to them based on the partition and the Bluetooth profile, or on a per-device-level, e.g. one partition is only allowed to use OBEX, but with any device, whereas one other may communicate using arbitrary profiles, but only with a predefined device.

2.2.6.3 Connection between OVERSEE Bluetooth Manager and User Partition

In every user partition that should be able to use Bluetooth devices there is an adapted Bluetooth stack, more exactly the upper part of a Bluetooth stack that will communicate with the Bluetooth manager over a secure channel provided by the hypervisor.

The drivers for Bluetooth profiles are located in these user partitions.

The goal of this separation is the possibility to attach Bluetooth peripherals to different user partitions in a flexible way, where each device is exclusively attached one partition, but multiple devices can be used at the same time.

Not having to implement the various profile drivers themselves in the secure I/O partition on the one hand reduces the attack surface considerably, while not limiting available profiles.

2.2.6.4 Residual Risk for the Bluetooth Communication Path

As any Bluetooth device is forwarded to a driver on a defined user partition, any security risk related with the device itself is also forwarded to the user partition. It is the responsibility of the runtime environment in the user partition to assure a secure handling of any device.

The interface between the secure I/O partition and a user partition introduces a split in the Bluetooth stack, which could be a source of implementation errors, so this interface must be developed with caution, especially on the side of the secure I/O partition.

2.2.7 Universal Serial Bus

A USB Interface enables easy and flexible hardware extension and is the de facto plug and play standard today. OVERSEE provides the capability to share USB devices compatible to the mass storage device class among partitions in a secure way.

Figure 7: USB communication path

2.2.7.1 Connection between USB Interface and USB Device Manager

The direct access to the USB interface will be handled within the secure I/O partition. The virtualization layer provides an explicit and secure communication path between the USB interface and the secure I/O partition creating a single point of access to the hardware interface. The USB device manager handles the USB interface in the secure I/O partition and forwards the interface as a virtualized block device to the user partitions in a secure way. Thus the secure I/O partition only acts as a forward path for the memory area to the user partition.

2.2.7.2 Connection between USB Device Manager and User Partition

The virtualization layer provides isolated end-to-end communication paths between the authorized user partitions and the secure I/O partition for the USB interface forwarding mechanism. The isolated communication paths are configured in the configuration phase of the platform assuring explicit communication rights between user partition and secure I/O partition.

A configuration method exists to restrict or allow user partitions to access the USB device. This configuration is performed in an early phase and is not meant to be changed dynamically assuring a restricted access to the interface. As many partitions can have the right to access an interface, a dynamic method assures the assignment of the interface to a user partition in a secure way. The method assures the avoidance of any conflicts accessing the USB interface.

2.2.7.3 Residual Risk for the USB Communication Path

As only USB devices compatible with the mass storage device class can be forwarded to the user partitions the main risk for the OVERSEE platform is a flaw in the corresponding driver in the secure I/O partition. Furthermore, for partitions, which are allowed to read from USB storage devices, there is a risk concerning malicious files. However the possible faults will be isolated in the affected partition.

2.2.8 ITS Communication

The communication interface of OVERSEE is based on existing standards, and those currently under development, e.g. in ETSI TC ITS. The ITS communication module is the single point of access to ITS communications. Hence, all ITS messages are received, sent, assembled, signed, verified, encrypted, respectively decrypted by the ITS communication module (possibly using functions provided by the security services partition, see section 2.2.9). ITS applications running in user partitions may receive or send messages through the ITS communication module using the OVERSEE ITS API. Figure 8 depicts an overview of the information flow of ITS communications.

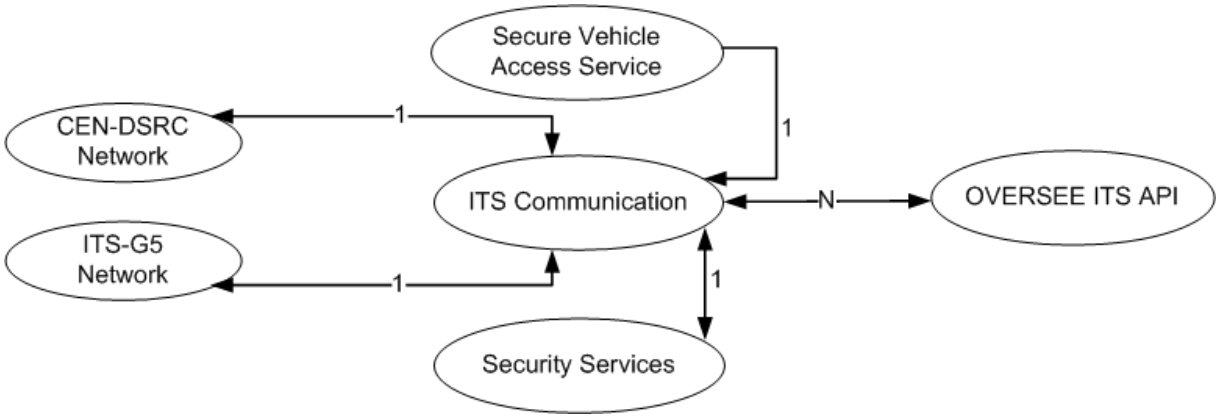

Figure 8: ITS communication path

2.2.8.1 CEN-DSRC Network

The European Committee for Standardization (CEN) published multiple standards on Dedicated Short Range Communications (DSRC). In contrast to ITS-G5, which allows vehicle-to-vehicle communications, CEN-DSRC is mostly limited to vehicle-to-infrastructure communications. Its main applications are automatic toll collection systems. The DSRC technology brings its own security mechanisms, which are not in the scope of OVERSEE.

2.2.8.2 ETSI ITS-G5 Network

Within the European Telecommunications Standards Institute (ETSI) a technical committee (TC) was created that shall produce standards for ITS. The different working groups within TC ITS deal with specifying functional requirements, architectures, geo-networking services, communication media, as well as security. The work within TC ITS is strongly linked to activities in other standardization bodies, in particular ISO CALM and IEEE (802.11 and 1609).

OVERSEE has the objective to be compatible with the ITS communication standards defined by ETSI TC ITS. Since these standards are still under development, OVERSEE may possibly not fully comply with particular standards. Section 3.2 describes how the ETSI ITS standards are integrated into the OVERSEE platform.

The ETSI TC ITS has defined separate security mechanisms concerning ITS communications, which are not in the scope of this document.

2.2.8.3 Connections between ITS Networks and ITS Communication Service

The ITS communication service accesses exclusively the hardware for the connection into ITS networks, e.g., CEN-DSRC and ITS-G5 networks. No other components of OVERSEE, especially no user partition, should access the ITS hardware.

2.2.8.4 ITS Communication Service

The ITS communication service provides all necessary functions for ITS communications and manages basic communication functions independently. Its field of duty covers the following tasks:

- Receive incoming Cooperative Awareness Messages (CAMs) from neighbouring cars. CAMs may be delivered to privileged user partitions or optionally evaluated and incorporated into a Local Dynamic Map (LDM).
- Optionally, the relevance of incoming messages may be checked. Based on the LDM and the vehicle's location a received message can be determined as relevant or not in the current context.
- Broadcast the current vehicle status periodically into the ITS-G5 network. Therefore, information like the vehicle's location and speed is gathered and condensed into CAMs.
- Deliver incoming Decentralized Environmental Notification Messages (DENM) to registered (and privileged) user partitions.
- Send different types of message (e.g., DENMs) on demand of ITS applications, which are running in privileged user partitions.
- Perform cryptographic operations and plausibility checks on all kinds of ITS messages:
    - Verify and/or decrypt incoming messages.
    - Sign and/or encrypt outgoing messages.
- Support for CEN-DSRC

The respective standards already define security measures for ITS communications. Some of them (e.g., signing and verifying of messages) will be implemented in OVERSEE's ITS communication service.

2.2.8.5 Connection between ITS Communication Module and OVERSEE ITS API

Usage of ITS communication is restricted individually for each partition. Authorization may be granted based on a fine-grained set of rules:

- Basic rule: Do not provide any access.
- Provide read-access to partitions that are explicitly authorized for read-access. Further restrictions determine which kind of messages are forwarded or which information may be read. By default no information is readably and no notifications on incoming messages or events are forwarded.
- As an extension to basic read-access rule: Forward incoming CAMs to partitions, which are authorized explicitly for that feature.
- As an extension to basic read-access rule: Grant read-access to an optional LDM to partitions, which are authorized explicitly or that feature.
- As an extension to basic read-access rule: Forward incoming DENMs to partitions, which are authorized explicitly for that feature.

- As an extension to basic read-access rule: Forward incoming CEN-DSRC messages to partitions, which are authorized explicitly for that feature.
- Provide write-access to partitions that are explicitly authorized for write-access. Further restrictions determine which kind of messages may be sent or which information may be set. By default no parameters may be written and no kind of message may be sent.
- As an extension to basic write-access rule: Allow authorized partitions to set parameters that are optionally broadcasted with CAMs (e.g., emergency lights on/off).
- As an extension to basic write-access rule: Allow partitions to send CAMs, if they are authorized explicitly for that feature.
- As an extension to basic write-access rule: Allow partitions to send DENMs, if they are authorized explicitly for that feature.
- As an extension to basic write-access rule: Allow partitions to send CEN-DSRC messages, if they are authorized explicitly for that feature.

2.2.8.6 OVERSEE ITS API

Security inside user partitions is not in the scope of this document.

2.2.8.7 Residual Risk for the ITS Communication Path

The ITS communication service is an additional interface to external entities, which brings additional risks. Allowing specific environmental conditions or behaviour of communication participants to affect the internal state of the vehicle involves the risk of manipulations. Since processing of ITS on application level is done inside user partitions and therefore outside the responsibility of OVERSEE, it is the task of integrators and application developers to consider these risks. By doing so, they should be aware that ITS communication technologies are novel developments that might contain undiscovered security flaws.

2.2.9 Connection between ITS Communication Service and SVAS

The facility of the ITS communication service to send independently periodic status messages (i.e., CAMs) requires access to sensor data of the vehicle. Therefore, the hosting partition should have the permission to read specific values from the secure vehicle access service. Writing data to the CAN bus is not required and should not be allowed.

Privileges for the ITS communication Service to access the secure vehicle access service should be granted according to the following set of rules, which is derived from the specification of CAM in ETSI TS 102 637-2 [3]:

- No write access
- Read access to the values below is mandatory for periodic CAMs:
    - position (and position confidence)

- vehicle type
- speed (and speed confidence)
- heading (and heading confidence)
- vehicle length and width
- curvature (and curvature confidence)
- longitudinal acceleration
- exterior lights
- acceleration control

- Optionally, read access to further values is required depending on the vehicle type, e.g., emergency vehicles should read and broadcast the state of their siren and emergency lights.

For the purpose of reading sensor data, it is sufficient to use the SVAS API as ordinary user partitions in order to integrate the secure vehicle access service. Hence, the same security measures as described in section 2.2.2 apply here.

The ITS communication module must not provide any sensor data to user partitions. As a consequence, received echoes from own ITS messages must be filtered out.

Including status information in the header of lower layers is disputed currently within ETSI. Hence, it may be possible that some status information, e.g. position, speed, heading, etc. will be part of any ITS message in future. See ETSI TS 102 636-4-1 [4] for more details.

2.2.9.1 Residual Risk for the Connection between ITS Communication Service and SVAS

Since this connection relies on the interface described in section 2.2.2, the same risks remain here. Furthermore, there is an additional risk (depending on the correctness of implementation) that sensor values may be read by accident from unprivileged user partitions.

2.2.10 Security Services

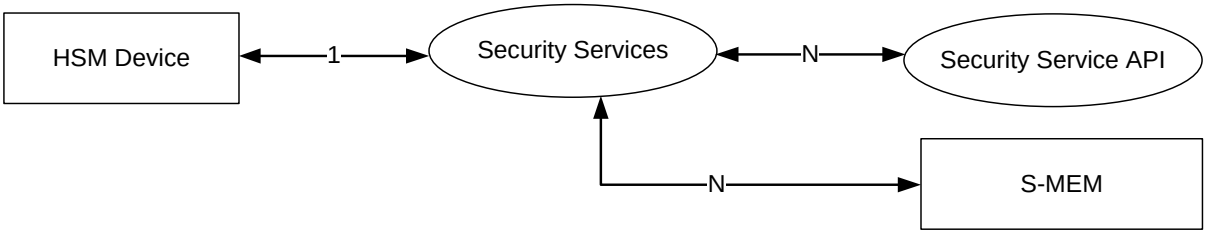

Figure 9: Security services communication path

The security services architecture of OVERSEE consists of hardware security module (HSM), the security services running in the dedicated security partition, and the security service API providing the necessary interface for the user partitions to the security services.

From this point, a restricted authorization is granted to the different user partitions. Usage rights of security functions and especially key data are only granted indirectly through the

security services. The user partitions can access the security services partition over a dedicated API using a secure communication channel provided by the hypervisor. As mentioned in 2.2.1 the security service application can securely determine the origin of any request and check for authorization of the requesting partition for the requested service.

The security application in the security service application provides a configuration table to assign usage rights for key data, certificates, and security functions to specific partitions. More information concerning key management can be found in chapter 3.

2.2.10.1 Connection between HSM and Security Services Partition

The security services partition provides a secure and dependable partition for security related services and functions. The hardware security module is assigned to the security services partition granting authorization to its communication interface. The access to the HSM is provided by the virtualization layer, restricting any access to the HSM by unauthorized partitions. The security and trustworthiness of the hardware interface between OVERSEE and the HSM is an important issue assuring the security of the platform, nevertheless this issue will not explored any further in the OVERSEE design and is left to later implementations of the platform.

Another important issue is the secure key storage located in the HSM. This storage is an internal memory area and is not accessible directly over the HSM interface or any other debug interface, thus assuring a secure storage for key material.

The services provided by the HSM and any other service provided by the security partition are isolated from the rest of the OVERSEE platform by the virtualization layer.

2.2.10.2 Connection between Security Services Partition and User partition

The user partitions can access the security services provided by the security services partition only through dedicated communication channels provided by the virtualization layer. The security partition provides restricted access to security services and key data depending on the access rights of the user partitions. The virtualization layer assures the origin of the request to the security partition.

2.2.10.3 S-Mem

The secure memory functionality is principally a file encryption technique and a mimicked file system embedded in the user partition for the encrypted files. The files are transferred between the user partition and the security services partition with the same secure communication channel mentioned in 2.2.10.2. The file encryption is done with a random session key created for the file. The key creation is done in the security partition, thus the key cannot be accessed by other partitions. Afterwards the session key is encrypted with a dedicated key. The user partition has to be authorized for the usage of this key. The encrypted file and encrypted key are sent back afterwards to the user partition. The pseudo file system mimics an extra storage for the user in which the encrypted files are stored. The pseudo file system is embedded in the user partition, which is isolated and restricted for access by other partitions.

2.2.11 IP Communication

IP based communication is the prevalent communication for IT applications and expected for many infotainment applications (e.g., map update for navigation systems). Hence, OVERSEE provides access to Wi-Fi and 2G/3G data connections via IP to the partitions, as described within [1]. The OVERSEE implementation of the IP-Management component is placed in the secure I/O partition and the corresponding communication path is presented in Figure 10.

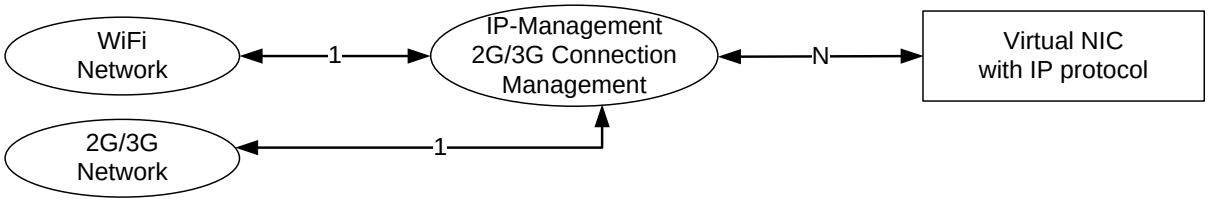

Figure 10: IP communication path

2.2.11.1 Wi-Fi Network

The security of Wi-Fi networks is a topic of well-known standards in the IT domain (e.g., IEEE 802.11i-2004 or 802.11i [6] better known as WPA (Wi-Fi Protected Access)). Nevertheless, especially for spontaneous connectivity there is still the need for unsecured connections for some use cases (e.g., location-based advertisement without need for an additional, maybe costly, IP connection through a 2G/3G network). Obviously, the security issues raised by insecure IP connections, like the confidentiality and authenticity of communication partners, have to be solved in higher layers of the communication stack.

OVERSEE is able to connect to Wi-Fi networks supporting the following security standards:

- Wi-Fi networks without security mechanisms according to 802.11 [7]
- WEP (Wired Equivalent Privacy) secured networks according to 802.11 [7]
- WPA secured networks according to 802.11i [6]

Please keep in mind:

- Do not assume any security assurance by native IP connections provided by OVERSEE. If applications need security assurances consider the use of SSL/TLS on top of IP or the use of application specific security services, please consult also section 2.2.11.7 on end-to-end connections over IP.

2.2.11.2 2G/3G Network

Current mobile phone networks are protected against many security weaknesses, according to the related standards. Unfortunately, some of the applied security mechanisms are already, at least partially, broken (see, e.g., [9]). Therefore, also IP connections over 2G/3G networks should be treated as insecure connections and the same indications as in the section 2.2.11.1 apply.

2.2.11.3 Connection between Wi-Fi Network and IP Management

The connection between the Wi-Fi network and the IP management will be the driver for the network module. Since filtering will be done within the IP management, no additional security topics are addressed here.

2.2.11.4 Connection between 2G/3G Network and IP Management

The connection between the 2G/3G network and the IP management will be the driver for the 2G/3G module and the PPP (Point-to-Point Protocol) to establish the connection. Since filtering will be done within the IP management, the only security topics addressed here are related to the Point-to-Point Protocol:

- Support the authentication mechanisms as required by the Point-to-Point Protocol.

2.2.11.5 IP Management and 2G/3G Connection Management

Within the IP management the IP packets will be routed between the interfaces bounded to the IP protocol. Additionally, by use of NAT (Network Address Translation) the internal communication details towards the partitions will be hidden to the outside world, see Figure 11 for an exemplary configuration.

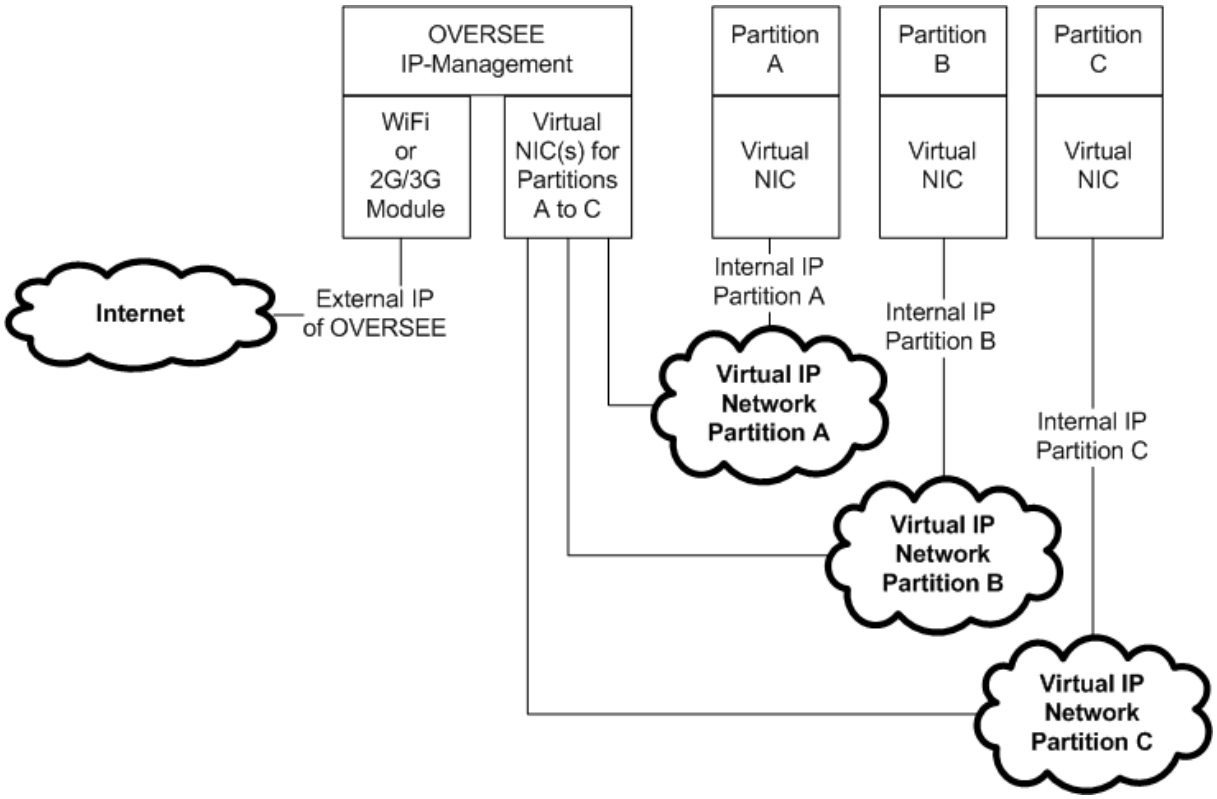

Figure 11: NAT in OVERSEE

The routing of IP packets and NAT will be implemented by reuse of well-established libraries, e.g., the netfilter project [8], which also provides a flexible and secure way of packet filtering at different positions in the communication path and packet processing. Please consult the section about the OVERSEE firewall in [2] for more details about this topic.

Application Layer Firewall also called Proxy Firewall (optional)

It is well known that packet filtering on source and destination addresses as well as port and protocol information is only an insufficient protection mechanism for IP networks, as long as internal communication partners are able to change communication settings (e.g., changing the port of a protocol). Often application layer firewalls also called proxy layer firewalls are used to improve the protection. These firewalls are able to filter the transferred packets on application protocol level (e.g., HTTP or DNS). Hence, the firewalls are able to detect protocols working on non-default and maybe maliciously selected ports. However, also these advanced firewalls could be bypassed by the use of advanced tunneling techniques (e.g. HTTP through DNS tunnels). It is up to the system integrator of a real word OVERSEE implementation, to decide whether or not he would like to implement such an application layer firewall, also adding new complexity and overhead to the system. The integration of such an application layer firewall in the OVERSEE communication stack would be rather easy, while this firewall is not part of the generic OVERSEE implementation.

Connection management, prioritization of network traffic

Basically, the OVERSEE connection management is responsible to build an IP connection via the best available network link (Wi-Fi or 2G/3G network). However, since 2G/3G data transmission is mostly causing additional costs, while Wi-Fi network connections are rarely reliable during driving, additional control measurements are necessary. Thus, there is a routing rule for each partition, granting or denying access to the Wi-Fi or 2G/3G network link.

Since at least 2G/3G connections could cause additional costs, the decision whether or not a partition is granted to use a network link will be configured by the platform owner. Except for obligatory rules by the platform integrator (e.g., mandatory access to 2G/3G network for the partition serving the eCall application).

As OVERSEE supports a wide range of applications, from games to serious applications relying on timely data transfer (e.g., real time traffic information), network traffic has to be prioritized. Therefore, each partition which is granted to use a specific network link (Wi-Fi or 2G/3G) has corresponding priority information, which could be analyzed by the IP and connection management of OVERSEE to prioritize the network traffic, see also the section about secure assurance of prioritized resource access in [2] on this topic.

The following rules apply to the IP and connection management:

- Basic rule: Do not build any network connection over Wi-Fi or 2G/3G network links.
- Basic rule: Do not route any packets between the network interfaces.
- As an extension to the basic rules: Build an IP connection over a Wi-Fi network if, and only if, a partition requesting an IP connection has granted access to use IP connections over Wi-Fi network links.
- As an extension to the basic rules: Build an IP connection over a 2G/3G network if, and only if, a partition requesting an IP connection has granted access to use IP connections over 2G/3G network links.
- As an extension to the basic rule: Forward packets over an already available 2G/3G connection only if the related partition has granted access to 2G/3G networks.

- As an extension to the basic rule: Forward packets over an already available Wi-Fi connection only if the related partition has granted access to Wi-Fi networks.
- Shutdown idle 2G/3G connections after an appropriate time, which could be configured by the platform owner.
- If there are more than one partition using an external IP connection and the data transfer rate of the connection is lower than the requirements of all partitions using this link, use the priority setting for the specific partitions to manage network traffic.
- Hide the internal communication structure within OVERSEE by the use of NAT.
- Do not route network packets between the internal virtual NICs connected to the partitions (no inter-partition communication through IP).
- Only route packets between the partitions and the external network interfaces (Wi-Fi or 2G/3G) if, and only if, the packets are fulfilling the firewall policies. Please consult the section OVERSEE firewall in [2] for more information on this topic.
- If an application layer firewall will be used, only route packets which are checked by this firewall and are fulfilling the firewall policies.

2.2.11.6 Connection between IP Management and Virtual NICs in the Partitions

The connection between the virtual NICs will, according to [1], be conducted by use of queuing ports and probably in the future also shared memory techniques. Since this communication is out of the user space, no additional security measures are necessary. However, it has to be stressed that the network connections to the partitions have to be authentic, to ensure the restricted access to network resources as described in section 2.2.11.5.

2.2.11.7 End-to-End Connections over IP

Based on TCP/IP connections, TLS (Transport Layer Security), also known as SSL (Secure Sockets Layer), offers secure network connections by the use of a hybrid cryptographic protocol. Since TLS is integrated in the transport layer (TCP/IP model) of the communication stack in most modern operating systems, it offers secured end-to-end communication for a lot of IP based applications (e.g., https also called HTTP over SSL). Since the implementation of TLS is already available in the guest operating systems, which will be the end points of the secured connections, the implementation is out of the project scope of OVERSEE. Nevertheless, since the security of the TLS secured connection depends on the trustworthy storage of the security credentials (public keys or root certificates and private keys for client certificates) the OVERSEE key storage supported by HSM capabilities offers a valuable service to enhance the security level of end-to-end connections over IP. Additionally OVERSEE provides a service for secured communication on top of IP connections, please consult the section about the secure communication API in [2] for more information on this topic.

- Store security credentials for secured end-to-end connections within the HSM supported key storage of OVERSEE.

- Use guest operating TLS libraries or the secure communication API of OVERSEE to establish secure end-to-end connections over IP.

2.2.11.8 Residual Risk for the IP Communication Path

Opening closed systems for the IP protocol and binding these systems to the internet introduces a lot of new security weaknesses. Since all external IP networks have to be considered as insecure and standard firewall techniques offer only an insufficient protection level, it should be seriously examined which partitions really require external IP connectivity. This is especially important if partitions additionally are allowed to access the vehicle internal electronics through SVAS, or other safety relevant communication links (e.g., ITS networks). In cases where connectivity to IP and other resources are not avoidable, particular attention should be paid to the installable applications, since most of the attacks over IP networks are enabled through malicious software (e.g., trojan horses) on client side. The following residual risks for the IP communication path have to be considered:

- Use of unsecured IP connections for applications with security issues.
- Loss or modification of security credentials for secure TLS connections.
- Acceptance of faked security credentials for TLS connections, in case of buggy implementations of certificate validation procedures or PKI. Unknown security breaches in the used firewall software or security libraries.
- Trojan horses in the partitions may open the system for malicious attackers.

2.2.12 Voice Connection over 2G/3G Networks

Some use cases for OVERSEE are also using a voice connection over the 2G/3G network link (e.g., voice connection to the PSAP (Public Safety Answering Point) in eCall). Due to the safety relevance of this functionality, there is also a security issue concerning the guaranteed prioritized access on shared communication interfaces for some applications. Since this security issue does not only apply to the voice connection over 2G/3G networks, this issue will be discussed in the section about secure assurance of prioritized resource access in [2].

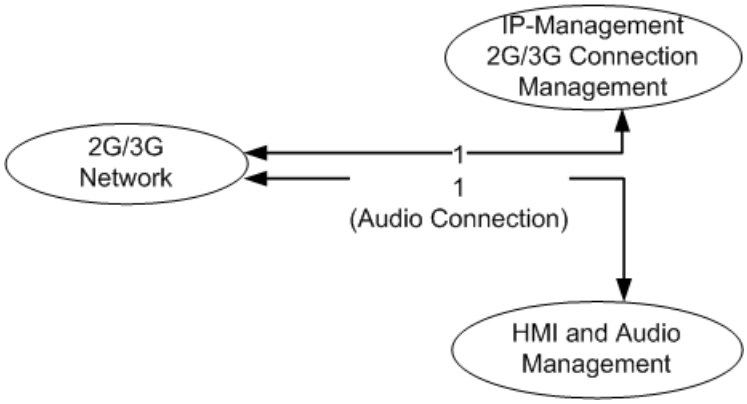

Figure 12: 2G/3G networks voice connection path

3 Key Management

This chapter describes the security parameter management of OVERSEE. First, general key handling features are described. Furthermore, requirements of ITS are explained and harmonized with the facilities of OVERSEE.

3.1 Organization of Key Distribution, Storage, and Usage

To create a dependable and secure platform the security, integrity and authenticity of key data has to be assured. OVERSEE architecture takes a number of measures to assure a trustworthy key organisation.

First of all, OVERSEE includes a hardware security module with secure key storage. Key data stored in the HSM cannot be accessed directly over the HSM interface or any other debug interface. Any key stored in the HSM is dedicated to specific use cases, which are specified via the usage flags during creation or importation of the key. Furthermore each usage option of the key can be further restricted with an authorization password. Thus, the key can only be used for the specified usage options and only by users with the proper authorization. When the key is used for any cryptographic function in the HSM, the user has still no direct access to the key material even though having the right to use the key data. To get direct access to the key data, the user has to export the key from the HSM which is again coupled to the usage flags and authorization. It is also possible to allow exportation of a key only in an encrypted state, again restricting any access to key data. To sum up the HSM ensures a secure storage for key material enforcing access restrictions even against users directly accessing the HSM.

A key consists of the following information fields.

- Key ID.
- Validity information.
- Use flags, indicating operations that can be performed with this key (e.g. verification, encryption etc.).
- Usage authorization data, indicating the authorization requirements for each use flag. This includes the definition of authorized partitions and authorization passwords.
- Key data.
- Key handle for internal use.
- Key signatures and/or certificates.

From this point, we can build a further restriction to key material through the security services partition. As the HSM is accessible only through the security services partition, any user partition requesting the usage of a key has to contact the security service in the security services partition. In this stage, the security service can further restrict the usage of the keys or any other service depending on the authorization of the requesting partition.

To create a trust anchor the OVERSEE Platform is shipped with two integrated keys, the MVK (public module manufacturer verification key) and the OVK (OEM verification key), to e.g. verify imported key data, sign outgoing data, proof authenticity etc. These keys can be used as trust anchors as they are provided before shipment of the platform. They can be used to verify authenticity of imported data signed by the manufacturer or OEM. For instance, key data to be imported can be signed with private keys of the platform manufacturer or the OEM and afterwards verified in the platform with the dedicated verification keys, providing a secure mechanism to import authentic keys into the platform. Furthermore, classic public key infrastructures can be used to import key data. Beyond that, the platform implies an DIK (device identity key), which is also integrated into the platform prior to shipment and provides a unique key pair for each platform.

The user partitions have access to the security services over a dedicated API. This API enables access to services and indirectly to key data. Cryptographic functions are executed in the hardware security module or security services partition and access key data internally. The user partition thus has no direct access to key data and refers to key information with handles. The security service and/or HSM subsequently check the needed authorization for the key data and approve or deny usage of key data.

3.2 ITS Key Management

The ITS subsystems require key management as well. Since OVERSEE shall offer a single point of access to ITS communications, the ITS communications module needs to manage all keying material for this communications path as well.

3.2.1 Functionality and Requirements of Key Management in ETSI ITS-G5

The ITS subsystems require key management as well. Since OVERSEE shall offer a single point of access to ITS communications, the ITS communications module needs to manage all keying material for this communications path as well.

The final specification of ITS security standards is not finished at the writing time of this deliverable, but the principles are already set. ITS communications will require different types of keys for identification and authorization. These are namely the enrolment credentials and the authorization credentials. See ETSI TS 102 731 for more details.

The main functionality for ITS key management in the OVERSEE platform will therefore be:

- Secure storage for both types of credentials
- Offer interfaces for credential update from an external authority for
    - Add credentials
    - Remove credentials
- Allow the use of credentials via the OVERSEE API
    - Offer high level access functions
- Interface with OVERSEE's low level cryptographic functions for
    - Cryptographically secure generation of random numbers

- Calculation of hash values (sha-1 and sha-2 algorithms)
- Hardware accelerated calculations for RSA and ECDSA signatures (sign and verify operation)

3.2.2 Integration into the OVERSEE Platform

The ITS communication service will take advantage of OVERSEE's security services by using its cryptographic and secure storage functions. With support for RSA and ECDSA, it provides all of the cryptographic functions, which are necessary for ITS-G5. The common OVERSEE security services API, which is used also by user partitions is sufficient for this task.

Due to the periodic exchange of pseudonyms, respective certificates are renewed and replaced accordingly. Like any ITS specific task, pseudonym exchange is managed by the ITS communication module. In addition, certificates of neighbouring vehicles must be obtained and stored in a secure way in order to verify received messages. Consequently, the ITS subsystem needs write access to its assigned security parameters. The security services API offers appropriate functions for importing, exporting, and generating keys for verification, signing, encrypting, as well as decrypting messages. An adequate configuration of the security services is necessary therefore and ensures that no other partition or user application may manipulate any ITS related certificates.

4 Remote Access

Accessing the OVERSEE platform via a remote connection is an advanced feature, not only for diagnosis reasons, but also for comfort of the platform user. The capability to upload and download data from the platform and configure the platform from home or office would be really appreciated by many customers. Concerning security, the following issues have to be considered:

- Security of the remote connection ensuring confidentiality and reliable authentication of the communication partners will be tackled in section 4.1.
- Authorization of remote users, according to their appropriate user rights (e.g., platform owner, platform user, OEM service) will be treated in section 4.2.

A further question, besides security, is how to activate and reach an offline OVERSEE platform. Since the vehicle ignition is off, normally the platform is also not available. While providing a proper solution is up to the integrator of a real world OVERSEE implementation, it is suggested that OVERSEE supports a suspend mode for this purpose. Within the suspend mode only the 2G/3G communication module should be active to be able to receive incoming connections and wakeup the whole platform (like wake on LAN known from PC environments).

4.1 Secure Remote Access to the OVERSEE Platform

As already indicated in the previous section, remote access to OVERSEE will be achieved through an IP connection. Obviously, this connection has to be secured against eavesdropping and malicious communication parties. Hence, a TLS secured end-to-end connection between the remote access component in OVERSEE and the remote access application on client side will be used. See Figure 13 for a schematic view based on the TCP/IP model.

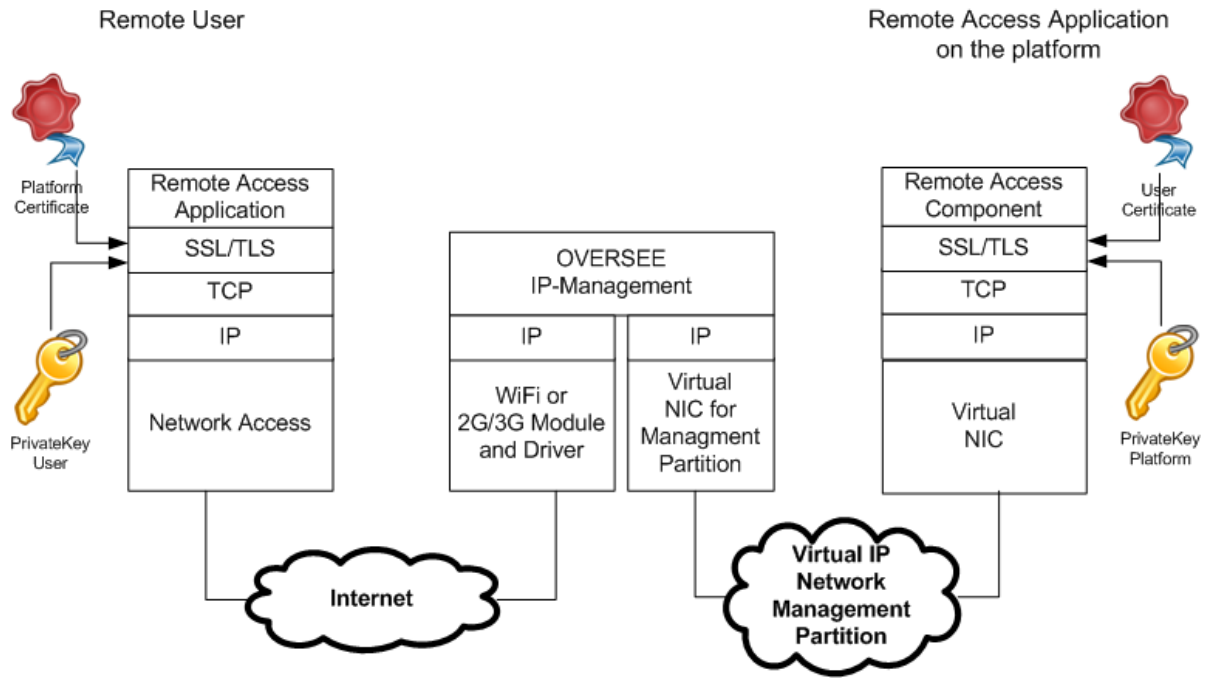

Figure 13: TLS secured remote access for OVERSEE

The mentioned certificates for remote users and the platform have to be issued by an appropriate certification authority, see section about user authentication in [2]. The private key of the platform has to be stored in the platforms secure key storage during an appropriate secure process at the setup of the platform. The user certificate would be transferred at the beginning of the communication session, see section about user authentication in [2].

In addition, there is the need for a secure replacement process of both security credentials (root certificate or public key of the certification authority and private key of the platform). The secure setup process and the secure replacement process are in the responsibility of the real world platform integrator, hence out of project scope.

4.2 Remote Diagnosis to Partitions

This part of the chapter assumes that a secure connection for diagnosis to the platform exists and deals with the distribution of the diagnosis functionalities to the user partitions.

OVERSEE provides basically two methods for diagnosis services. The first one is a central log to which partitions can report problems via commands provided by the virtualization layer. In addition, the hypervisor reports any peculiar activity of the partitions in this log. The second method is a communication channel provided between the diagnosis service and the user partitions. With this communication channel, the user partition developer can implement his own diagnosis services and use the secure diagnosis connection as a wrapper for his own diagnosis protocol.

The log and the diagnosis channels of the user partitions can only be accessed from the authorized partition running the diagnosis service. This restriction is provided by the virtualization layer. Any external user first has to establish remote access to the diagnosis service of OVERSEE. To gain further access rights to a specific diagnosis channel or the logbook the external user has to provide authorization credentials for these services. This

has to be assured with a proper authorization process. In the following, we will discuss shortly how such an authorization and authentication process could be realized.

The first step of a secure access is the authentication of the user. As it is not feasible to define all possible users statically in the OVERSEE platform there has to be a flexible method to enable authentication of users. This could be realized by importing the public key of the user with a certificate, which is signed by e.g. the OEM. As OVERSEE owns the OEM verification key the certificate can be validated and the public key of the user can be imported. The next step would imply a challenge response process to validate the authenticity of the user.

To improve the process further another step should be added to define fine granularity for authorization for services. This could be provided by a certificate defining authorization for specific services coupled with specific user IDs. The certificate again has to be signed by an authority, e.g. OEM.

5 Summary

In this document each communication path of OVERSEE has been analysed. Various security measures and many rules for restricting access were suggested. Residual risks were documented. Nevertheless, the following advice to integrators an application developers should always kept in mind; the security of user partitions and peripheral (external) components is not in the duty of OVERSEE and was therefore not considered in this document.

Handling of security parameter was described with regard to secure distribution, storage, and usage of keys. Special requirements of ITS were considered and mapped to the OVERSEE platform. Furthermore, secure remote access facilities as well as remote diagnosis interfaces were designed.

References

[1] OVERSEE Project: D2.1 List of interfaces and specifications of information flow. Dec 2010

[2] OVERSEE Project: D2.2 Specification of security services incl. virtualization and firewall mechanisms. Dec 2010

[3] ETSI: TS 102 637-2 V1.1.1. Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service. Apr 2010

[4] ETSI: TS 102 636-4-1. Intelligent Transport System (ITS); Vehicular Communications; Part 4: Geographical Addressing and Forwarding for Point-to-Point and Point-to-Multipoint Communications; Sub-part 1: Media-Independent Functionality. Apr 2008

[5] E-safety vehicle intrusion protected applications (EVITA) project, www.evita-project.org

[6] IEEE: 802.11i-2004, Medium Access Control (MAC) Security Enhancements, standards.ieee.org/getieee802/download/802.11i-2004.pdf

[7] IEEE: IEEE Standard for Information Technology- Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks-Specific Requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications

[8] Netfilter project: www.netfilter.org

[9] Karsten Nohl, Sylvain Munaut: Wideband GSM Sniffing, Presentation at the 27th Chaos Communication Congress, events.ccc.de/congress/2010/Fahrplan/attachments/1783_101228.27C3.GSM-Sniffing.Nohl_Munaut.pdf