5GAA Automotive Association

Cross Working Group Work Item
gMEC4AUTO

Cybersecurity for Edge Computing

5GAA Automotive Association
Technical Report on Global MEC Technology to support automotive services

CONTACT INFORMATION:
Lead Coordinator – Thomas Linget
Email: liaison@5gaa.org

MAILING ADDRESS:
5GAA c/o MCI Munich
Neumarkter Str. 21
81673 München, Germany
www.5gaa.org

VERSION:	v1.0
DATE OF PUBLICATION:	
DOCUMENT TYPE:	Technical Report
EXTERNAL PUBLICATION:	
DATE OF APPROVAL BY 5GAA BOARD:	

Contents

Foreword

This Technical Report has been produced by 5GAA.
The contents of the present document are subject to continuing work within the Working Groups (WG) and may change following formal WG approval. Should the WG modify the contents of the present document, it will be re-released by the WG with an identifying change of the consistent numbering that all WG meeting documents and files should follow (according to 5GAA Rules of Procedure):

x-nnzzzz

(1) This numbering system has six logical elements:
(a) x: a single letter corresponding to the working group:
where x =
T (Use cases and Technical Requirements)
A (System Architecture and Solution Development)
P (Evaluation, Testbed and Pilots)
S (Standards and Spectrum)
B (Business Models and Go-To-Market Strategies)

(b) nn: two digits to indicate the year. i.e. ,17,18 19, etc
(c) zzz: unique number of the document

(2) No provision is made for the use of revision numbers. Documents which are a revision of a previous version should indicate the document number of that previous version

(3) The file name of documents shall be the document number. For example, document S-160357 will be contained in file S-160357.doc

Introduction

When it comes to communications and computing technologies, cybersecurity is nowadays a topic of outmost importance. This is true in all technology fields around cloud computing, and especially in MEC (Multi-access Edge Computing), which is an emerging trend in the industry together with the introduction of 5G systems. The 5GAA approach to MEC security, privacy and trust, from automotive perspective, is following the work started in MEC4AUTO [28][29] (and continued in the present gMEC4AUTO work item), where the reference architecture is targeting MEC systems deployed in Multi-MNO, Multi-OEM and multi-vendor environments. As a consequence, this report is targeting a very specific and tailored scenario, thus covering a smaller part of the entire **"galaxy"** of cybersecurity, and an even smaller space of MEC Security (i.e. Multi-MNO, Multi-OEM and multi-vendor). In this context, it is important to understand what are in these MEC scenarios the specific security threats, and which are the available countermeasures (e.g. available tools which can be potentially reusable from traditional cloud and IT domains), in order to derive possible actions that may have to be taken, e.g. to find new or more specific solutions to address these security issues.

This Technical Report (TR) thus provides first an overview of related work and gaps from standard bodies and industry groups. It then analyses the main threats from security, privacy and trust perspectives, as tailored to the gMEC4AUTO architecture (MEC in Multi-MNO, Multi-OEM and multi-vendor environments); then, the TR provides an overview of the most relevant mitigation strategies available in the industry, by finally evaluating them in terms of suitability for the 5GAA gMEC4AUTO architecture and targeted use cases. Finally, the report highlights possible gaps and suggested future work in that perspective.

1. Scope

This gMEC4AUTO Technical Report (TR) studies cybersecurity aspects related to MEC (Multi-access Edge Computing) deployments in Multi-MNO, Multi-OEM and multi-vendor environments. Starting from an analysis of the threats in this gMEC4AUTO architecture (from security, privacy and trust perspectives), it provides an overview of Mitigation Strategies, as a toolbox of solutions available from standards or from industry-led consolidated implementations, by finally assessing them in terms of suitability from a gMEC4AUTO perspective (MEC in Multi-MNO, Multi-OEM and multi-vendor environments).

2. References

[1] I. De Oliveira Nunes, S. Jakkamsetti, N. Rattanavipanon and G. Tsudik, "On the TOCTOU Problem in Remote Attestation," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security (CCS '21)*, 2021.

[2] A. Biondo, M. Conti, L. Davi, T. Frassetto and A.-R. Sadeghi, "The guard's dilemma: efficient code-reuse attacks against intel SGX," in *Proceedings of the 27th USENIX Conference on Security Symposium (SEC'18)*, 2018.

[3] TCG, "TCG PC Client Specific Implementation Specification for Conventional BIOS," 2012.

[4] ETSI GR NFV-SEC 007 V1.1.1 , "Report on Attestation Technologies and Practices for Secure Deployments," 2017.

[5] IEEE 1609.2-2016, "IEEE Standard for Wireless Access in Vehicular Environments--Security Services for Applications and Management Messages," 2016.

[6] B. Brecht, D. Therriault, A. Weimerskirch, W. Whyte, V. Kumar, T. Hahn and R. Goudy, "A Security Credential Management System for V2X Communications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 12, pp. 3850 - 3871, 2018.

[7] European Commission, "Certificate Policy for Deployment and Operation of European Cooperative Intelligent Transport Systems (C-ITS)," 2018.

[8] 5G Automotive Association; WG7 "Security and Privacy", "5GAA Efficient Security Provisioning System," 2020.

[9] 5G Automotive Association, "Privacy by Design Aspects of C-V2X," 2020.

[10] ETSI, "TR 103 460 V2.1.1. Intelligent Transport Systems (ITS); Security; Pre-standardization study on Misbehaviour Detection; Release 2.," 2020.

[11] ETSI, "DTS/ITS-00561. Intelligent Transport Systems (ITS); Security; Misbehaviour Reporting service; Release 2.".

[12] W3C, "Verifiable Credentials Data Model v1.1," 2022.

[13] World Wide Web Consortium (W3C), "Decentralized Identifiers (DIDs) v1.0 - Core architecture, data model, and representations," 2022.

[14] B. Ali, M. Gregory and S. Li, "Multi-Access Edge Computing Architecture, Data Security and Privacy: A Review," IEEE Access, vol. 9, pp. 18706-18721, 2021.

[15] J. Gu, Z. Hua, Y. Xia, H. Chen, B. Zang, H. Guan and J. Li, "Secure Live Migration of SGX Enclaves on Untrusted Cloud," in *47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2017.

[16] National Institute of Standards and Technology (NIST), "Zero Trust Architecture," NIST Special Publication 800-207, 2020.

[17] GAIA-X, "Gaia-X Trust Framework - 22.10 Release," 2022.

[18] Trust over IP Foundation, “Introduction to Trust Over IP,” 2021.

[19] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Z. Ma, F. Kargl, A. Kung and J.-P. Hubaux, “Secure vehicular communication systems: design and architecture,” *IEEE Communications Magazine*, vol. 46, no. 11, pp. 100 - 109, 2008.

[20] ETSI GS MEC 009 v3.1.1 (2021-06), “ Multi-access Edge Computing (MEC); General principles, patterns and common aspects of MEC Service APIs”, https://www.etsi.org/deliver/etsi_gs/MEC/001_099/009/03.01.01_60/gs_MEC009v030101p.pdf

[21] S. D. Tambe, Y. Mandge and A. Antony Franklin, “Performance Study of Multi-access Edge Computing Deployment in a Virtualized Environment,” 2020 IEEE 3rd 5G World Forum (5GWF), Bangalore, India, 2020, pp. 424-429, doi: 10.1109/5GWF49715.2020.9221113.

[22] ETSI GS MEC 003 v3.1.1 (2022-04), “ Multi-access Edge Computing (MEC); Framework and Reference Architcture”, https://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/03.01.01_60/gs_MEC003v030101p.pdf

[23] 3GPP TS 23.558: “3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Architecture for enabling Edge Applications; (Release 17)”

[24] ETSI GS MEC 016 V2.2.1 (2020-04), “Multi-access Edge Computing (MEC); Device application interface”, https://www.etsi.org/deliver/etsi_gs/MEC/001_099/016/02.02.01_60/gs_MEC016v020201p.pdf

[25] ETSI GS MEC 014 V3.1.1 (2023-03), “Multi-access Edge Computing (MEC); UE Identity API”, https://www.etsi.org/deliver/etsi_gs/MEC/001_099/016/02.02.01_60/gs_MEC016v020201p.pdf

[26] GSMA Operator Platform Group (OPG), https://www.gsma.com/futurenetworks/5g-operator-platform/

[27] GSMA PRD (Permanent Reference Document), Operator Platform Telco Edge Requirements, v 3.0, Oct 2022

[28] 5GAA MEC4AUTO technical report, “MEC for Automotive in Multi-Operator Scenarios”, March 2021, https://5gaa.org/content/uploads/2021/03/5GAA_A-200150_MEC4AUTO_Task2_TR_MEC-for-Automotive-in-Multi-Operator-Scenarios.pdf

[29] 5GAA MEC4AUTO technical report, “Use Cases and initial test specifications review”, July 2021, https://5gaa.org/content/uploads/2021/07/5GAA_MEC4AUTO.pdf

[30] 5GAA gMEC4AUTO technical report, “ Moving toward federated MEC demos/trials (global MEC “, March 2023, 5gaa.org/moving-toward-federated-mec-demos-trials

[31] 5GAA gMEC4AUTO technical report, “ MEC System Interoperability and Test Framework “, March 2023, 5gaa.org/global-mec-technology-to-support-automotive-services

[32] TEC (Telco Edge Cloud) Forum, https://www.gsma.com/futurenetworks/telco-edge-cloud-forum/

[33] GSMA OPG white paper, “Operator Platform Concept; phase 1: Edge Cloud Computing”, February 2020. https://www.gsma.com/futurenetworks/resources/operator-platform-concept-whitepaper/

[34] GSMA OPG white paper, “Telco Edge Cloud: Edge Service Description & Commercial Principles Whitepaper”, October 2020, https://www.gsma.com/futurenetworks/resources/telco-edge-cloud-october-2020-download/

[35] ETSI White Paper No. 49, MEC federation: deployment considerations”, first edition, June 2022, https://www.etsi.org/images/files/ETSIWhitePapers/ETSI_WP_49_MEC-Federation-Deployment-considerations.pdf

[36] ETSI White Paper No. 46, “MEC security; Status of standards support and future evolutions “, second edition, September 2022, https://www.etsi.org/images/files/ETSIWhitePapers/ETSI-WP-46-2nd-Ed-MEC-security.pdf

[37] 3GPP TS 33.558, “ Security aspects of enhancement of support for enabling edge applications”

[38] 3GPP TR 33.739, “Study on security enhancement of support for edge computing phase 2”

[39] 3GPP TR 33.839, “ Study on security aspects of enhancement of support for edge computing in the 5G Core (5GC)”

[40] 3GPP TS 23.548, “ 5G System Enhancements for Edge Computing; Stage 2”

[41] 3GPP TR 23.748, “Study on enhancement of support for Edge Computing in 5G Core network (5GC)”

[42] 3GPP TR 33.818, “Security Assurance Methodology (SECAM) and Security Assurance Specification (SCAS) for 3GPP virtualized network products”

[43] 3GPP TS 33.122, “Security aspects of Common API Framework (CAPIF) for 3GPP northbound APIs”

[44] 3GPP TR 33.848, “Study on security impacts of virtualisation”

[45] ETSI GS NFV 002, “ Network Functions Virtualisation (NFV); Architectural Framework “

3. Abbreviations

For the purposes of the present document, the following abbreviations apply:

G3GPP	3rd Generation Partnership Project
5GAA	5G Automotive Association
5GC	5G Core
AC	Application Client
AF	Application Function
API	Application Programming Interface
CAPIF	Common API Framework
DN	Data Network
DNN	Data Network Name
EAS	Edge Application Servers
ECS	Edge Configuration Server
ECSP	Edge Computing Service Provider
EDN	Edge Data Network
EEC	Edge Enabler Client
EES	Edge Enabler Servers
eNB	evolved Node B
E2E	End-to-End
ETSI	European Telecommunications Standards Institute
ETSI ISG	ETSI Industry Specification Group
GSMA OPG	GSM Association Operator Platform Group
KPI	Key Performance Indicator
LCM	Life-Cycle Management
MEC	Multi-access Edge Computing
MEO	Multi-access Edge Orchestrator
MEP	MEC Platform
MEAO	Mobile Edge Application Orchestrator
MEPM	MEC Platform Manager
MNO	Mobile Network Operator
MSP	Mobility Service Provider
NAT GW	Network Address Translation GW
NEF	Network Exposure Function
NFV	Network Function Virtualisation
NMS	Network Management System
OEM	Original Equipment Manufacturer
PDU	Protocol Data Unit
PGW	PDN Gateway
PoP	Point-of-Presence
PSA	PDN Session Anchor
QoS	Quality of Service
RAN	Radio Access Network
RSU	Road Side Units
RTA	Road Traffic Authority
ToD	Tele-operated Driving
UE	User Equipment
UALCMP	User Application LCM proxy
WI	Work Item

4. MEC Security Requirements in Multi-MNO environments

4.1 Related Work and Gaps

In previous MEC4AUTO report [28], 5GAA has described the reference scenarios to be considered as relevant for global MEC deployments of automotive services. These scenarios are natively characterised by Multi-MNO, Multi-OEM and multi-vendor environments, and the report has performed a first analysis of the MEC4AUTO architecture from a security perspective, providing a general guidance for the secure implementation of MEC on a global scale.

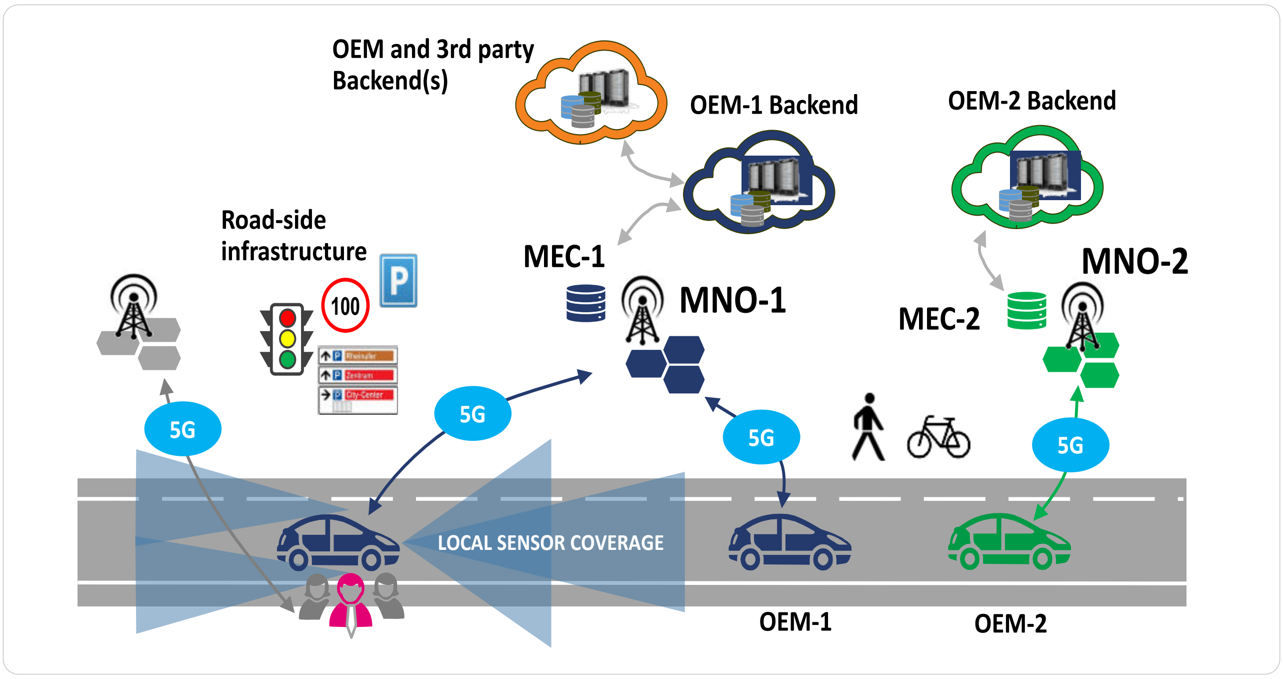


Figure 4.1-1: Reference MEC scenarios in 5GAA MEC4AUTO: Multi-MNO, Multi-OEM and multi-vendor environments [28]

That initial 5GAA guidance on security aspects (clause 9 of the MEC4UTO TR [28]) started from the awareness that in these scenarios MEC global deployments need collaboration between many parties, and organizations have to establish a Shared Responsibility Security Model to deal with this, whereas security and compliance is a shared responsibility between the MNO, the MEC tenant application provider and the application user. In particular, MEC deployments are characterized by the presence of multiple MNOs, and edge computing infrastructures, where systems are virtualized (where also potentially multiple parties can provide portions of an overall compute solution).

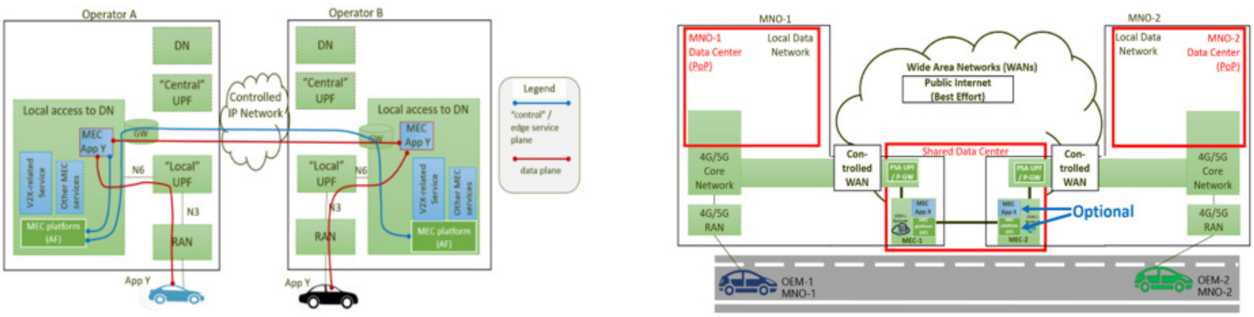


Figure 4.1-2: Reference MEC architecture in 5GAA MEC4AUTO [28]:

(left) MNO setup; (right) Neutral Host setup

The report analyzed the MEC4AUTO reference architecture (depicted in the above Figure 4.1-2) and described at high level the elements that are to be secured by the system, by providing some examples of security boundaries (identified by the MEC system itself and the associated security services that the MEC hosts for connected vehicles), in some key cases of interest:

- Security boundary in single OEM use case
- Security boundary in single OEM multi-MNO MEC use case
- Security boundary in multi-MNO MEC roaming use case

These examples of security boundaries provisionally identified in MEC4UTO technical report are very useful to better focus on the issues to consider and the specific measures to put in place to secure the whole architecture. However, the preliminary analysis done in that report requires a more accurate and complete study, which needs also to be referred to the updated gMEC4AUTO architecture [28], in order to determine how the whole architectural arrangement is securable with appropriate services and controls, and guarantee the needed level of security in global MEC deployments, to provide automotive services to customers.

In this perspective, the updated architecture (below), can be the reference for identifying more accurately the various security boundaries, when it comes to multi-MNO, multi-OEM and multi-vendor scenarios targeted by gMEC4AUTO.

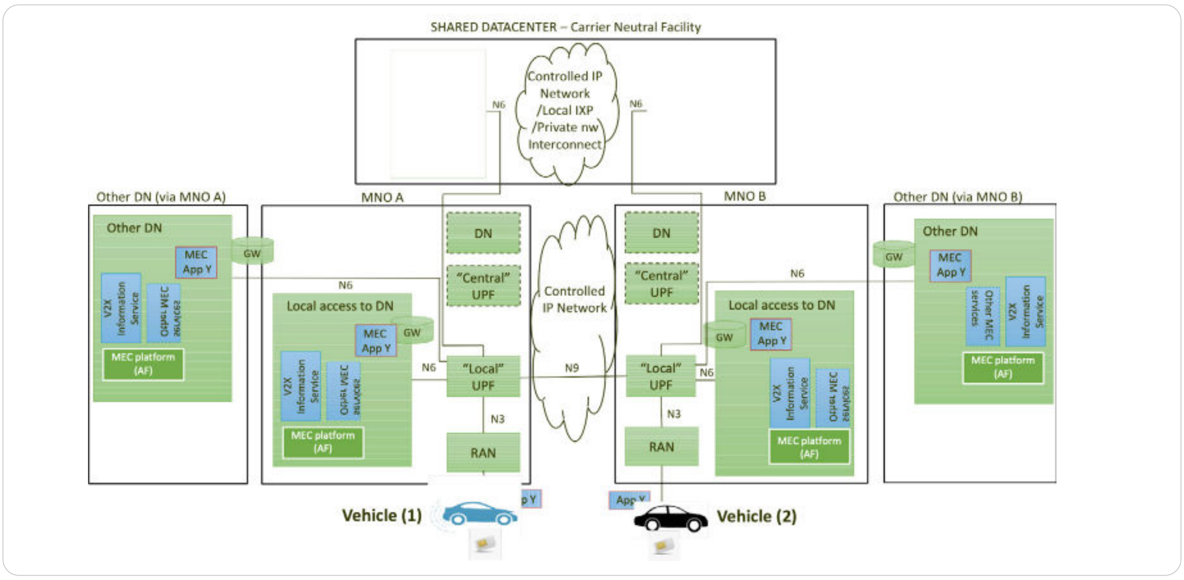


Figure 4.1-3: Reference MEC architecture in 5GAA gMEC4AUTO [31]

This architecture expands the set of security boundaries, as introduces the possibility to host MEC Platforms and MEC applications also in other DNs via the MNO networks, other than the presence of shared datacenters (as carrier neutral facilities). The above figure (taken from [31]) represents a generic architecture, that can be instantiated in multiple ways, depending on the following 5 attributes (dimensions):

1. Presence of MEC Application instance(s);
2. Presence of MEC Platform (s) to expose edge services;
3. Network Subscription of the end-user (vehicle (sub)system);
4. Available interconnection between MNOs;
5. Roaming options.

So, a complete enumeration of all combinations derived from these 5 dimensions is unpractical, also for the purpose of identifying the various security boundaries in all cases. Then, in this perspective, only some examples can serve as clarification for identifying security boundaries in some practical cases. For example, the figure below offers an overview of the security boundaries identifies in the case of Trial #3 described in [30], which involves vehicles from different OEMs, and multiple MEC systems managed by respective MNOs connected via N9 reference points.

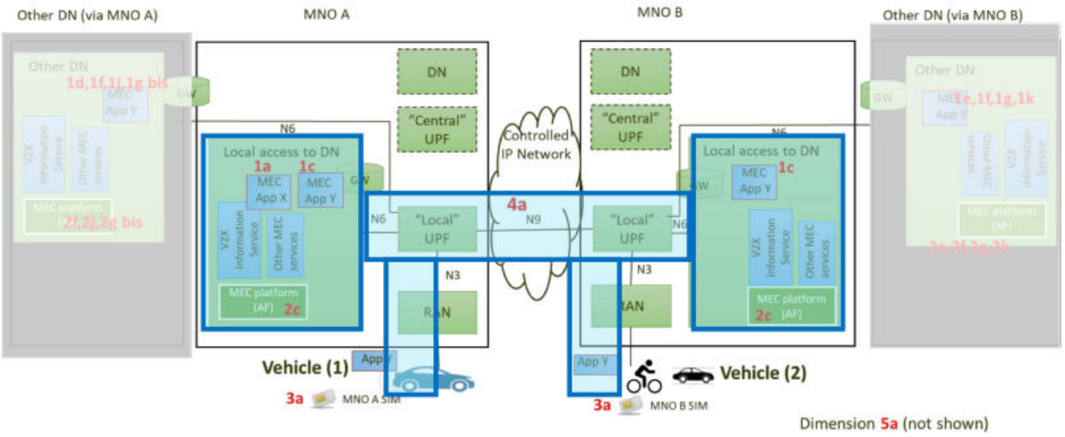


Figure 4.1-4: Example of security boundaries for Trial #3 (described in [30])

The remainder of this report is organized as follow: Section 4.1 is first providing an overview of the related work and gaps from standard bodies and industry groups, as an important preliminary step to elaborate suitable MEC Security Requirements by reusing existing work and avoiding work duplication; then, Security Threats in gMEC4AUTO architecture (Section 4.2) and Privacy Threats (Section 4.3) together with Trust Concerns (Section 4.4) are analyzed. Then, Section 5 provides an overview of Mitigation Strategies for MEC security in Multi-MNO environments, as a toolbox of solutions available from standards or from industry-led consolidated implementations. Finally, after some analysis of the various threats and on how the various mitigation strategies can address them in gMEC4AUTO architecture, some considerations and possible recommendations on Future Work in Section 6 conclude the report.

4.1.1 GSMA OPG

Since the 5GAA focus of MEC deployments is characterised by Multi-MNO, Multi-OEM and multi-vendor environments, a relevant work in this space is given by GSMA OPG (Operator Platform Group) [26], which is composed of over 40 of the world's leading operators and over 25 key ecosystem partners. The group has introduced the concept of the Operator Platform (OP), where edge compute from operators should be federated and exposed in the same fashion to create a multi-domain capability that could be presented to customers/developers. Moreover, the exploitation of the edge can be enhanced by utilizing network resources (e.g., device location, user plane control, mobility, etc.).

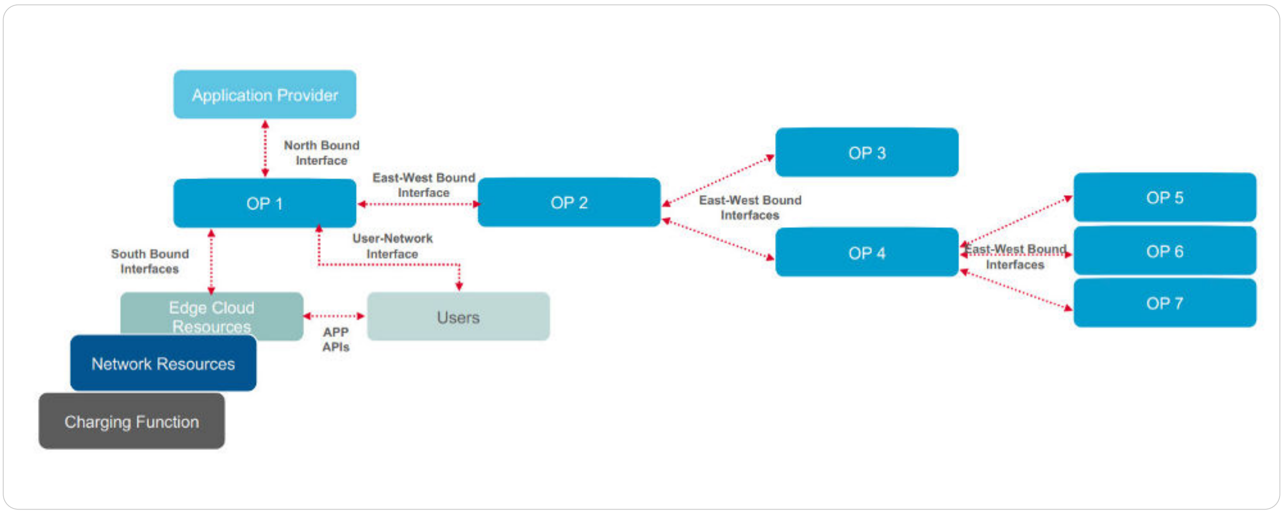


Figure 4.1.1-1: High-level Architecture of the Operator Platform defined by GSMA OPG (source TEC Forum [32])

The OP concept, architecture and core functionality are introduced in initial white papers [33][34] whilst in a second phase a Permanent Reference Document (PRD) [27] specified more in detail the technical requirements, functional blocks and interface characteristics. In this document, clearly automotive use cases (e.g. UC1 - Automotive - Advanced Horizon, UC2 - Automotive – Remote Driving) are key for deriving the OP requirements.

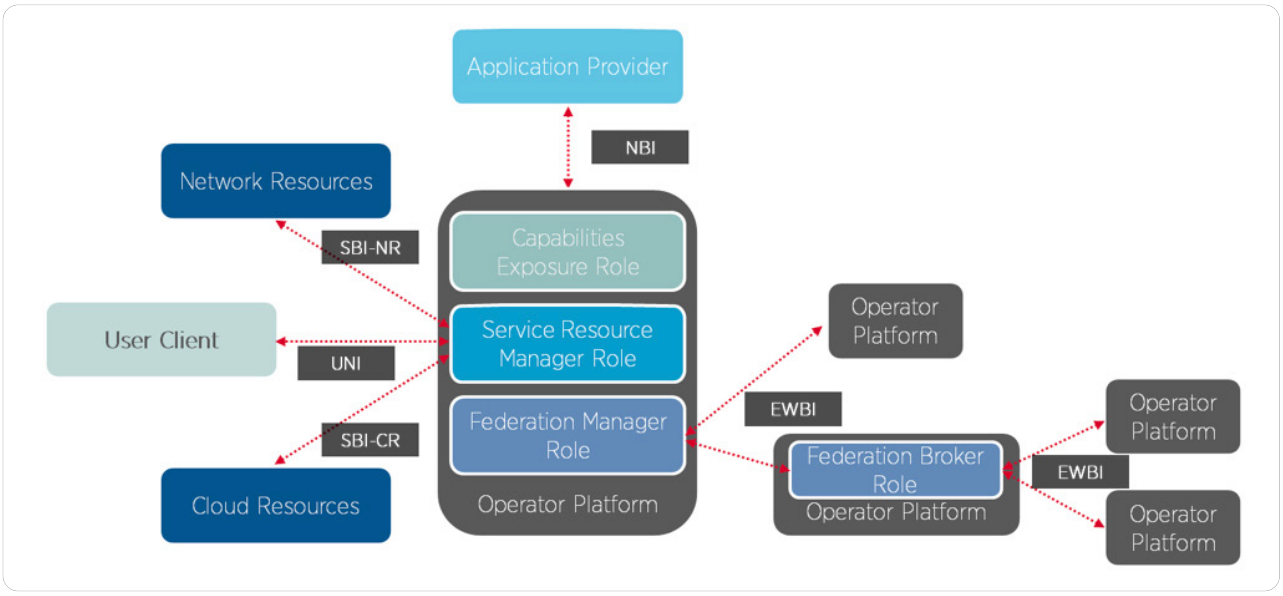


Figure 4.1.1-2: Architecture of the Operator Platform (source GSMA OPG [27])

The latest version of the OPG document, currently focusing on Edge Computing, provides not only a target architecture and requirements to enable an end-to-end delivery chain for different services, but also an extensive analysis (Annex E of the PRD) of the security-related implications of the federation, together with a categorized list of OP threat vectors:

Access Threat vectors

- These are at locations that connect a UE to the OP system. In ETSI ISG MEC, the vulnerabilities are on the RAN link from the UE to the BTS/eNB/gNB, between the UE application and the UE client and in the UE itself.

Architecture Threat vectors

- Vulnerabilities that occur in the overall architecture of a system or its components. Therefore, those vulnerabilities may manifest themselves in OP functions as well as in reference points. The significant categories of threat vectors have to do with validating containers and VMs, both in a particular platform and upon migration to other platforms and with performing traffic steering to applications in a secure manner.

Core Threat Vectors

- They affect the core 5G network, orchestrators, resource managers, controllers, and applications. In OP's case, where implementations of these components map onto Capabilities Exposure and Service Resource manager roles, all of the Core threat vector types appear to be relevant.

Edge Threat Vectors

- cover platform managers, VIMs, MEC platform connectivity and connectivity of MEC apps operated at non-local base stations. These threat vectors appear to map to the EWBI.

Other Threat Vectors

3 areas that do not fit at a specific reference point and which manifest because of functionality, not architecture. For example, charging/billing

Privacy Threat Vectors

3 Data privacy, location privacy, identity privacy; Computational Offloading privacy threats, etc.

Also, in this PRD the GSMA OPG analyzed the security implications toward the SDOs (Standard Developing Organizations) that are identified for the standardization work on federation (ETSI MEC and 3GPP). The following subsections provide a quick overview of the current status in the respective bodies, from a MEC security point of view (figure below shows a possible view of a cross-SDO mapping of the OP architecture as presented in the joint workshop organized by GSMA OPG with ETSI MEC and 3GPP and captured in a more recent white paper [35]), which is in fact showing elements from both ETSI MEC and 3GPP SA6 EDGEAPP architectures, with some indication of the relevance of the various reference points for the OP architecture interfaces. However, standardization work in this area is still ongoing, hence the figure below should be considered as a suitable starting point for a cross-SDO mapping of the OP architecture (at time of writing this deliverable the final mapping is still not finalized but is already giving an idea of the main components involved in the OP, and the possible impacts to the two SDOs from a security point of view).

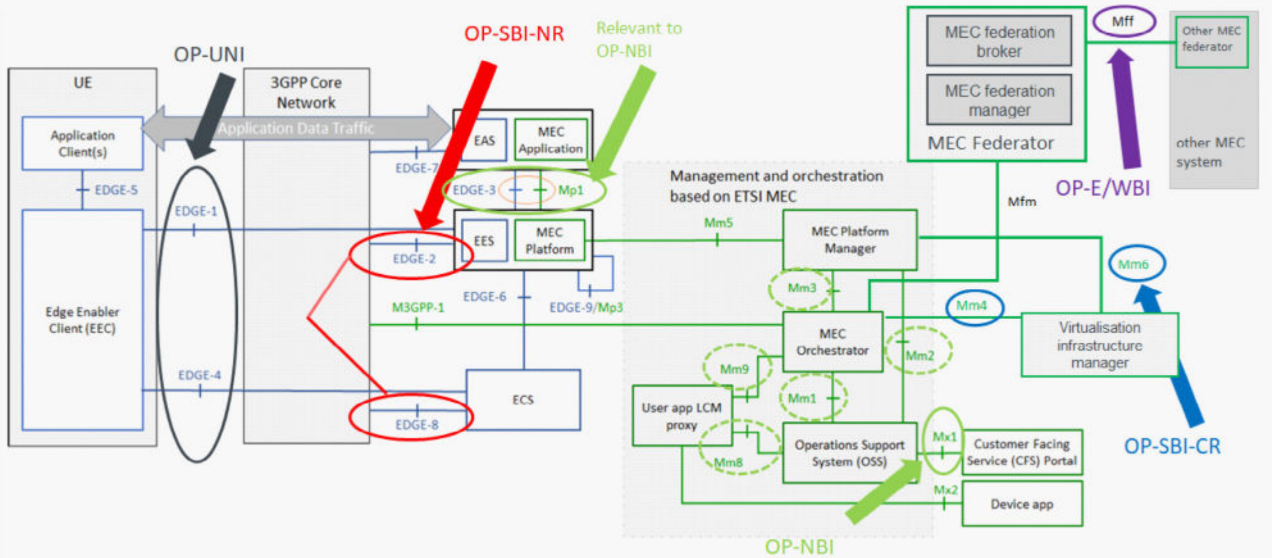


Figure 4.1.1-3: Cross-SDO mapping of the OP architecture (top-down approach) [35])

4.1.2 ETSI MEC

MEC Security is a very comprehensive topic, including many aspects since Edge computing environments are characterized by a complex multi-vendor, multi-supplier, multistakeholder ecosystem of equipment including both HW and SW devices. As a first effort, the ETSI ISG MEC has driven the publication of a white paper, jointly with experts from ETSI NFV SEC, TC CYBER, 3GPP SA3 and other relevant standard organizations. The paper [36] intended to explore security-related use cases and requirements with the aim of identifying aspects of security where the nature of edge computing results in insufficient industry approaches to cloud security.

Based on ENISA 5G Threat Landscape, the potential threats related to MEC include:

- **Abuse of assets**, which mainly involves exploitation of software or hardware vulnerabilities leading to Zero-day exploits, software tampering and system execution hijack which can impact information integrity, service availability etc.; furthermore, APIs serve as conduits that expose applications for third-party integration; as a consequence of that, also APIs are potentially susceptible to attacks like any other software.
- **Compromised supply chain**, vendor and service providers due to tampering of network product (configuration or source code), abuse on third parties' personnel access to MNO facilities and manipulation of network product updates can also result in service unavailability, information destruction and initial unauthorized access.
- **Unintentional damages** that may occur due to misconfigured or poorly configured systems, inadequate designs, and erroneous use or administration of the network, system, and devices can potentially impact service availability and information integrity.

Regarding MEC systems, threat factors can be broadly categorized based on various areas of vulnerabilities: from Platform Integrity to Virtualization and Containerization, Physical security, Application-Programming Interfaces (APIs) and Regulatory issues. More in detail, all MEC security threats can be at various levels (as depicted in the figure 4.1.2-1 below): MEC App / EAS / other applications, MEC platform / EES, NVFI and infrastructure (that may include implicitly also security issues at real estate level), management & orchestration (also possibly including non-standard orchestration frameworks). Also, research work have recently reviewed the status of MEC standardization from a security perspective, analysing the various threats and gaps in this perspective [14].

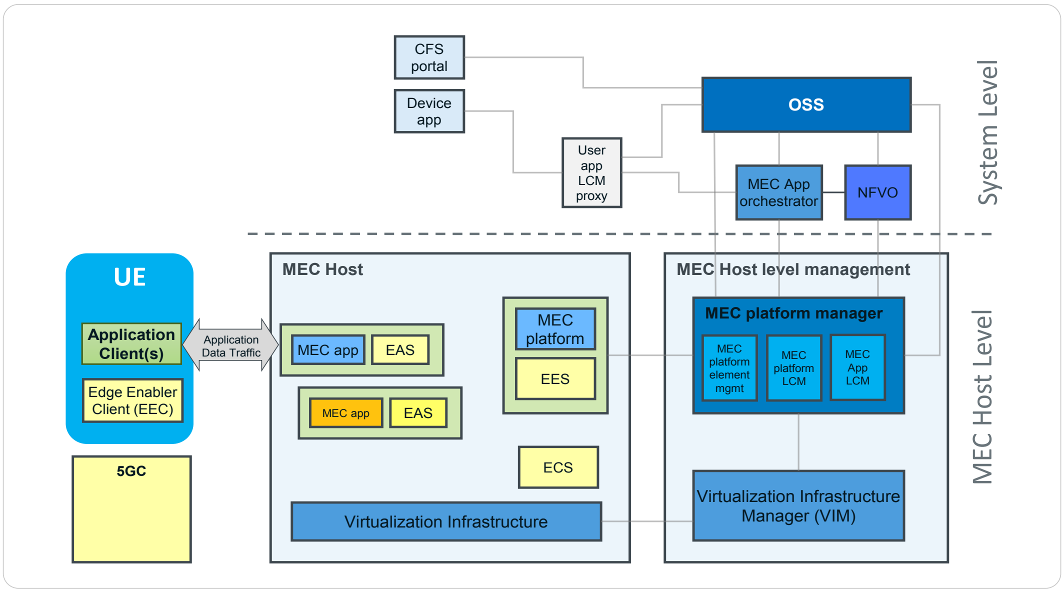


Figure 4.1.2-1 - Functional elements of the synergized architecture that may be subjected to MEC security threats [36])

Currently, a new work item is open (ETSI GS MEC 041: "Study on MEC Security" [36]), with the aim to study security topics and paradigms that apply to MEC deployments. The study will broadly cover the themes of application and platform security, Zero-Trust Networking, and security requirements for MEC Federations. It may also draw upon prior work from other standards and gather requirements from industry associations (e.g. 5GAA), and will identify gaps in ETSI ISG MEC and provide recommendations for new normative work.

4.1.3 3GPP

As mentioned earlier in this section, 3GPP SA6 has defined for the Release 17 the so-called EDGEAPP architecture (see 3GPP TS 23.558 [23]). This architecture is including entities such as Application Client (AC), Edge Application Server (EAS), Edge Configuration Server (ECS), Edge Enabler Client (EEC), Edge Enabler Server (EES).

In this perspective, 3GPP (see TR 33.839 [39]) is studying the security enhancements on the support for Edge Computing in the 5G Core network defined by SA2 (see 3GPP TR 23.748 [41] and 3GPP TS 23.548 [40]), and application architecture for enabling Edge Applications defined in TS 23.558 by SA6 ([23]). Also, security aspects related to edge computing are being defined by 3GPP SA3 for Rel.17 in TS 33.558 [37] and studied (for Rel.18) in TR 33.739 [38].

A key asset in edge computing environments is the possibility to expose and consume service APIs to application level. However, APIs are a well-known subject of multiple attacks types, as they are exposed to external access. The common API Framework (CAPIF) is used by 3GPP as the standardized means to support providing and accessing APIs (and ETSI MEC is fully aligned with CAPIF). From a software development point of view, compliance with CAPIF should be ensured during API design and implementation phases. In this perspective, a further reference is 3GPP TS 33.122 [43] on Security Aspects of Common API Framework (CAPIF) for 3GPP northbound APIs.

Furthermore, 3GPP SA3 is working on two Technical Reports on Security Impacts of Virtualisation (see 3GPP TR 33.848 [44]) and Security Assurance Methodology (SECAM) (see 3GPP TR 33.818] [42]), leading to the introduction of a set of Security Assurance Specifications (SCAS) for 3GPP virtualized network products. Both of these reports explore the additional threats and mitigations required to design, test and deploy functions in a virtual environment. MEC use cases represent many of the higher risk threat scenarios identified by SA3. In particular, 3GPP TR 33.848 [44] is investigating the security consequences of virtualization of 3GPP NFs (see figure 4.1.3-1 below) and targeting Release 18 of the 3GPP specifications. This 3GPP report is applicable to many MEC use cases where the need for additional security controls is higher than in core network data center implemented network functions. It is expected to result in additional ETSI NFV security requirements that can be utilized for MEC.

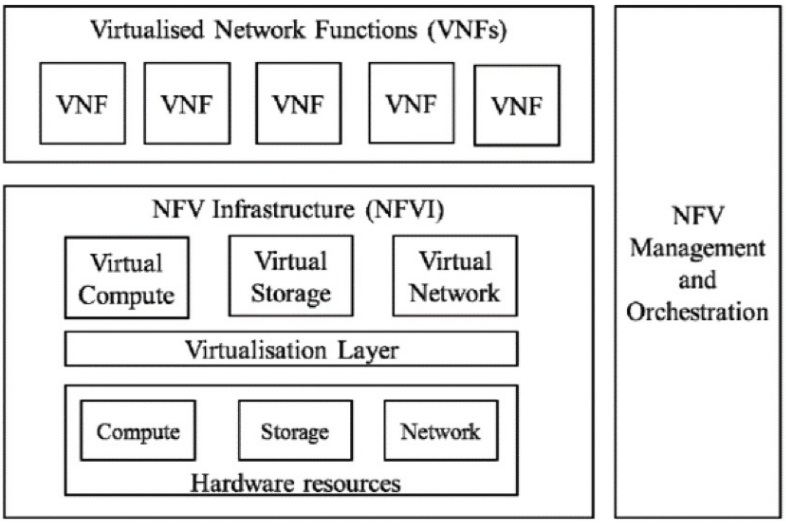


Figure 4.1.3-1 - ETSI NFV high-level architecture (ETSI GS NFV 002) [45])

On the other side, TR 33.818 [42] provides security assurance mechanisms in 3GPP virtualized environment: the document already considers threats related to the integration of ETSI VNF concepts and interfaces within the 3GPP virtualized system, including limited security of the interface between 3GPP VNF and VNFM, the interface between the virtualization layer and hardware, and the interface between virtualization layer and the VIM. Different generic virtualized network products (GVNPs) are defined in the document, including type 1 (implementing 3GPP defined functionalities only), type 2 (implementing 3GPP defined functionalities and virtualization layer), and type 3 (implementing 3GPP defined functionalities, virtualization layer, and hardware layer). Compared to physical network product, GVNP has also two types of logical interfaces, i.e. execution environment interfaces and remote logical interfaces. The remote logical interfaces are interfaces which can be used to communicate with the GVNP from another network node and also include the remote access interfaces to the GVNP for its maintenance through e.g., an Element Management (EM), a Virtualized Network Function Manager (VNFM).

4.2 Security Threats in gMEC4AUTO architecture

In edge environments, data and computing can be in principle distributed across the entire processing chain, from the traditional where the data is gathered from the "edge" devices, routed back to the cloud and/or datacentre, to organizations increasingly looking to perform these functions on physical compute structure at or near the data source itself. Today, security solutions provide encryption when data is in storage and when it is sent across the network, but data can still be vulnerable when actively processed in memory, especially when that data is being processed at the edge where the scale of enterprise-level security is prohibitive. This is especially true for gMEC4AUTO architecture, where automated and connected vehicles must operate securely using data being processed in near-real time, actively from memory.

In the United States, the National Institute for Standards and Technology (NIST)[1], The Department of Defense (DoD)[2], the Committee on National Security Systems (CNSS)[3], National Security Presidential Directives (NSPD)[4], U.S. Code, and a host of other regulations govern cybersecurity. The three major European regulatory documents: ENISA[5], NIS Directive[6], and the EU General Data Protection Regulation (GDPR)[7] establish the framework by which data must be secured. This not only includes static data (e.g., personal privacy information like hospital records) but has extended to dynamic data required by vehicles to operate semi- and fully automated in dynamically changing environments such as presented in the gMEC4AUTO scenarios. It has become imperative to first understand the **threat landscape**, identify how the whole system must comply with regulatory mandates, and how the technology communities can effectively address these challenges today and in the future.

4.2.1. Threat Agents and Attacker Models

Let's begin with an understanding of the threat landscape. We learned that security in the digital transformation age is about two things:

1) hardening the platform with hardware-based security capabilities that establish trust, recovers from attacks gracefully (and degrades from attacks gracefully), with visibility/control of the processes, and
2) a definitive need to protect data (at rest, in flight, and in use) as depicted in Figure 4.2.1-1 below.

[1] https://www.nist.gov/cyberframework
[2] https://dodiac.dtic.mil/dod-cybersecurity-policy-chart/
[3] https://www.hsdl.org/?view&did=16776
[4] https://www.loc.gov/rr/news/directives.html
[5] https://www.enisa.europa.eu/
[6] https://www.enisa.europa.eu/topics/nis-directive
[7] https://gdpr-info.eu/

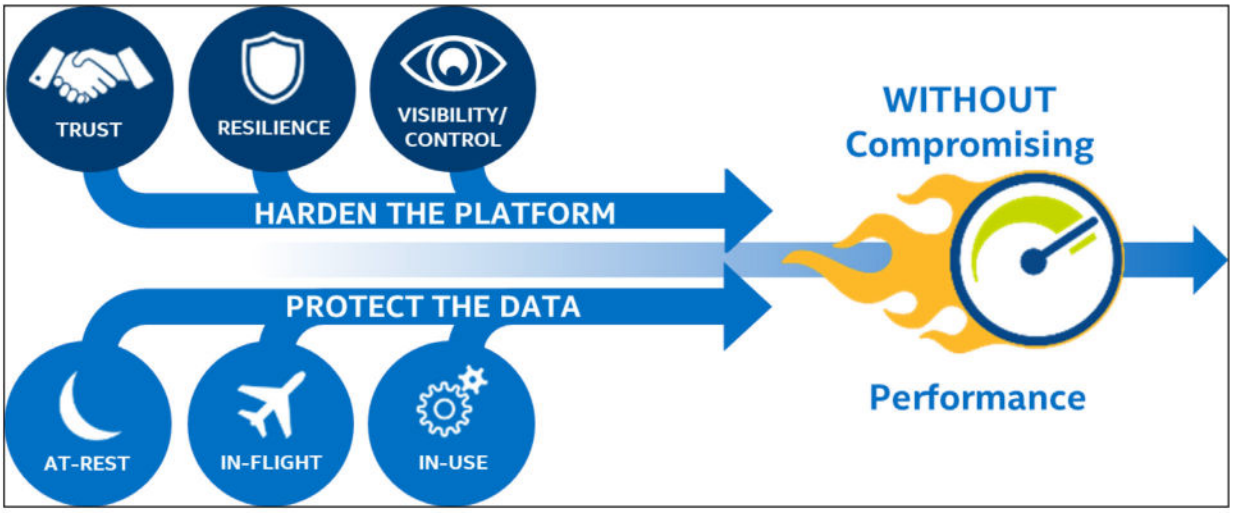
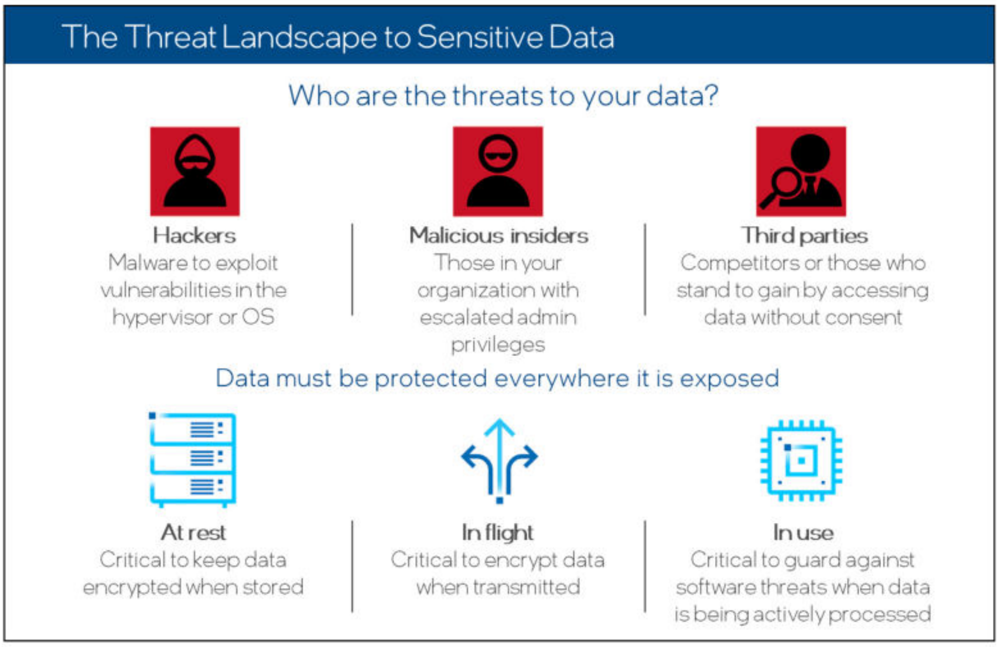


Figure 4.2.1-1: Security in the Digital Transformation Age

All of which cannot compromise the performance of the system. Designing in security is important for developing software and hardware because it becomes more difficult to add security as a system develops. In addition, dealing with existing cybersecurity vulnerabilities and patching them in real-time can be difficult. Morevoer, it will never be as effective as designing systems to be as secure as possible from the beginning. So, we must "get it right" the first time before deployment in a highly contested environment.

We also learned several other valuable lessons about securing our globally distributed infrastructure: software alone, cannot sufficiently protect valuable enterprise assets, and security threats are the product of both, external malicious actors and internal vulnerabilities.



The Threat Landscape to Sensitive Data

Who are the threats to your data?

Hackers
Malware to exploit vulnerabilities in the hypervisor or OS

Malicious insiders
Those in your organization with escalated admin privileges

Third parties
Competitors or those who stand to gain by accessing data without consent

Data must be protected everywhere it is exposed

At rest
Critical to keep data encrypted when stored

In flight
Critical to encrypt data when transmitted

In use
Critical to guard against software threats when data is being actively processed

Figure 4.2.1-2: The Threat Landscape (related to data protection)

We’ve identified seven potential threats to secure operations in this environment. Note, this list is not exhaustive, but represents major categories of threats most likely encountered. The reader should keep in mind, that these threats are always changing, adapting to protection methodologies, and also these security measures are continuously adapting and changing to provide a robust threat mitigation posture to engage in this new landscape.

Editor’s Note: in this section we are only treating data in the threat landscape. However also treats related to applications and workloads are important and might be considered as FFS.

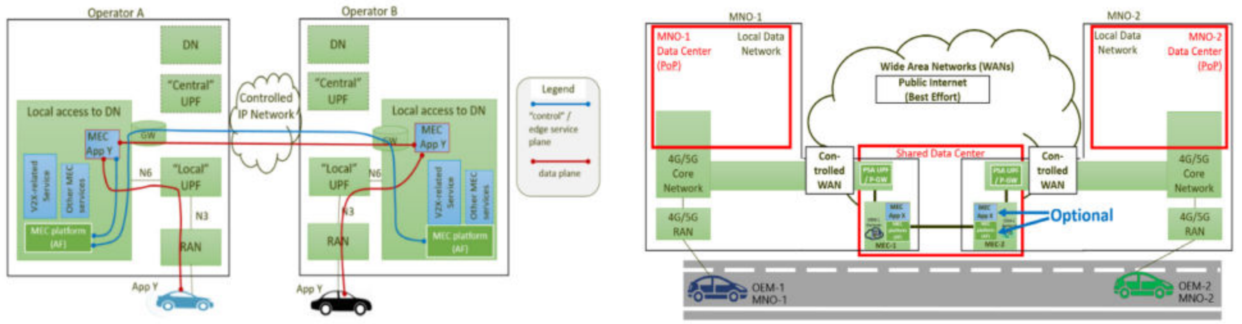


Figure 4.2.1-3: Reference MEC architecture in 5GAA MEC4AUTO [28]: (left) MNO setup; (right) Neutral Host setup

Now, tailoring the security threats to the gMEC4AUTO architecture (depicted in the figures below, both considering the MEC deployments in MNO Setup and Neutral Host Setup), the above-described security landscape should be referring to MEC security in the multi-MNO scenarios analysed in 5GAA. Main aspects to be always considered, when referring security threats specifically for the gMEC4AUTO architecture, are:

- Workloads are outside the PLMN trusted domain, but running on external ECSP domains
- Mutual trust between MEC Applications and MEC platforms, meaning that 1) in principle the edge application from MNO A should be considered as if it would be running on an “hostile” environment (MNO B) and also vice-versa, i.e. 2) a platform operated by MNO B is hosting “unknown” applications which may endanger the system;
- Security threats are also related to the communication links in this figure (both data plane and control plane!), meaning that all relevant communication channels can be untrusted, in principle;

Furthermore, also devices can be a further source of security issues. Examples: the car, but also the VRU, including smartphones and other connected devices.

In the following, we provide also multiple examples of security treats relevant in this context:

3 Malware injection attacks (e.g. SQL)
3 Man-in-the-middle attacks
3 Denial of Service (DoS) attacks
3 Advanced Persistent Threat (APT)
3 Ransomware
3 Other attacks

4.2.1.1 Malware

Malware attacks are the most common type of cyberattacks. Malware is defined as malicious software, including spyware, ransomware, viruses, and worms, which gets installed into the system when the user clicks a dangerous link or email. Once inside the system, malware can block access to critical components of the network, encrypt system and data, damage the system, and gather confidential information, among others. It is naive to think a connected and automated vehicle would be any less vulnerable to such attacks. For instance, a fan controller on the main compute board receives a firmware update. The update can be uploaded either by Bluetooth communications or through hard contact in a service garage. Embedded in the firmware update is malicious code. The malicious code now has access to the main compute board. A malware attack is insidious in that the culprit lies in wait, gradually degrade the system performance, and avoid detection through carefully masked actions: a slight voltage drop here and there, all individually within tolerance but in aggregate exceeds limitations. The end results can be catastrophic, fatal in fact. Once injected (in this case in the car), a malicious software can indeed communicate with the other elements in the architecture (Figure 4.2.1-3) and propagate its effects to other cars and/or to network infrastructure.

5GAA_gMEC4AUTO_SecurityThreat ST#1 – Malware

4.2.1.1.1 SQL Injection

As a particular case of malicious software, a Structured Query Language (SQL) injection attack occurs when cybercriminals attempt to access the database by uploading malicious SQL scripts. Once successful, the malicious actor can view, change, or delete data stored in the SQL database.

Connected cars are expected to heavily use the low-latency, high-bandwidth, and network-slicing features of 5G as cellular networks that take advantage of the next-generation technology standard roll out across the globe. 5G networks will serve as the modern wireless infrastructural backbone that will work together with advances in artificial intelligence and machine learning, for both onboard and in-cloud data processing, to bring more autonomous features to connected cars. These technological advances are either currently under development or already being implemented.[8]

[8] https://documents.trendmicro.com/assets/white_papers/wp-driving-security-into-connected-cars.pdf

With advances in connected cars come important concerns of cybersecurity too. It is worth noting, however, that the motivation of cybersecurity as it relates to connected cars is not limited to securing autonomous driving. The automotive industry will continue to create cars, but it will also broaden its offerings to include a variety of mobility services for different use cases. These are encapsulated in the mobility-as-a-service (MaaS) system, which uses end-to-end digital solutions providing private and public vehicle users an easy and streamlined way of traveling. One of the key technologies of this emerging model is connected and automated driving, which emerges now the need to focus on the cybersecurity of connected cars.[9]

An SQL injection attack is relatively easy to accomplish, but has very deleterious effects. The connected car ecosystem is extremely complex, with potentially millions of endpoints and end users. The complexity of this ecosystem, with its immense size and many functions, makes for large and at times unpredictable attack surfaces. Although they primarily communicate wirelessly, connected cars heavily depend on the networked ITS infrastructure for communications. In our threat modelling exercise, we focused on attacks that could be launched remotely against and/or from the victim vehicles. The primary target of this attack type is the head unit or middleware that runs the car.[10]

4.2.1.2 Man in the Middle

Man in the Middle (MitM) attacks occur when cyber criminals place themselves between a two-party communication. Once attackers interpret the communication, they may filter and steal sensitive data and return different responses to the user.

This can be a prevalent means by which a nefarious actor can co-opt a semi- or autonomous vehicle. At the begin of 2022 for example, *"a security researcher in Germany managed to get full remote access to more than 25 Tesla electric vehicles around the world. A security flaw in the web dashboard of the EVs left them wide open to attacks. (The researcher warned Tesla, and the software has since been patched.)"*.[11]

Man-in-the-Middle (MitM) attacks can be in principle performed in any elements in the architecture (Figure 4.2.1-3), but obviously the easiest way exploits vulnerabilities in the devices and vehicles, and also in wireless communication links.

5GAA_gMEC4AUTO_SecurityThreat ST#2 – Man in the Middle

4.2.1.3 Denial of Service

Denial of Service (DoS) attacks flood systems, networks, or servers with massive traffic, thereby blocking the system to fulfil legitimate requests. Attacks can also be based on many infected devices to attack the single target system. This is also known as a Distributed Denial of Service (DDoS) attack. Although this appears to be the realm of data centres and large server farms, DDoS attacks can attack any device connected to the grid.

[9] ibid

[10] ibid

[11] https://venturebeat.com/2022/05/15/car-hack-attacks-its-about-data-theft-not-demolition/

From a cybersecurity point of view, the vehicle can be simplified interpreted as a client application of the entire V2X system. Most new high-end cars can already connect in multiple ways, including cellular, Bluetooth, Wi-Fi, and USB, and in the future via V2X. And, while the IT industry has been in the networking business for decades, dealing with security and developing risk mitigations and best practices, car makers are relatively new to these issues. Case in point: hacking the Jeep Cherokee.[12]

5GAA_gMEC4AUTO_SecurityThreat ST#3 – Denial of Service

4.2.1.5 Advanced Persistence

The economic impact of advanced persistent threats is immeasurable and requires constant (and consistent) vigilance. An advanced persistent threat (APT) is a stealthy threat actor, typically a nation state or state-sponsored group, which gains unauthorized access to a computer network and remains undetected for an extended period[13,14]. In recent times, the term may also refer to non-state-sponsored groups conducting large-scale targeted intrusions for specific goals.[15]

Although this type of attack seems outside the norm for the automotive industry, ENISA identifies it as categorically relevant. Attempts to "tune" the vehicle driving characteristics, for example to enhance performance. The car hacker's handbook, for example, is advertised to users as mean to perform "car mods" or "discover undocumented features". ENISA provides the description of this attack as follows:[16]

In the case of an alteration by the legitimate user, the scenario could consist in getting a direct connection to car components, then trying to persistently alter the behaviour of a given ECU. The objective may be for example vehicle tuning, bypass of the geo-fencing on a corporate vehicle. The user may also use diagnostic equipment, which may also be used by other categories of attackers, for example in a garage. The steps would then consist in obtaining a legitimate or illegitimate access to diagnostic equipment, then exploiting a vulnerability in the diagnostic equipment to persistently alter the behaviour of an ECU. In a garage context, such an attack may be related to business intelligence as much as an attack on the vehicle itself.[17]

5GAA_gMEC4AUTO_SecurityThreat ST#5 – Advanced Persistence

4.2.1.6 Ransomware

Ransomware is a type of malware attack in which the attacker locks or encrypts the victim's data and threatens to publish or blocks access to data unless a ransom is paid. Ransomware and other cyber-attacks are the enemy of today's data-driven organization. Attacks are increasingly destructive, driving the cost per attack into the millions. Cyber

[12] https://spectrum.ieee.org/why-the-next-denial-of-service-attack-could-be-against-your-car
[13] "What Is an Advanced Persistent Threat (APT)?". www.kaspersky.com. Retrieved 11 August 2019.
[14] "What Is an Advanced Persistent Threat (APT)?". Cisco. Retrieved 11 August 2019.
[15] Maloney, Sarah. "What is an Advanced Persistent Threat (APT)?". Retrieved 9 November 2018.
[16] See Car Hacker's Handbook by Craig Smith
[17] https://www.enisa.europa.eu/publications/cyber-security-and-resilience-of-smart-cars/@@download/fullReport

threats take on many different forms and attackers use multiple techniques and platforms. It's not a matter of "if" an organization will be targeted by cyber criminals but "when."

Recent research conducted by Future of Automotive Security Technology Research (FASTR) has concluded that connected cars could be the next targets for ransomware hackers/developers. FASTR which technically acts as a consortium of automotive manufacturers, software makers for automotive industry and suppliers, discovered in its research that as soon as a connected car connects to the internet, the entire vehicle gets exposed to threat surface.[18]

5GAA_gMEC4AUTO_SecurityThreat ST#6 – Ransomware

4.2.1.7 Other treats: Zero Day Exploits

Although Zero-Day exploits are a small percentage of the overall attack picture today, they are trending exponentially since 2019. A zero-day threat (sometimes called a zero-hour threat) is one that hasn't been seen before and doesn't match any known malware signatures. This makes it impossible to detect by traditional signature-matching solutions. It may exploit a previously unknown software vulnerability (sometimes called a zero-day vulnerability), or it may be a new malware variant delivered by traditional means.

X-Phy describes a case, where a hacker has identified a serious vulnerability in self-driving cars that would give him access to remotely control self-driven cars. He created a patch update and then sends it over the network specifically to several IP addresses that he spoofs, because he was able to intrude the network. The attacker has taken a zero-day attack to the next level. While a simple zero-day attack would have been to exploit a zero-day vulnerability before a patch is rolled out, he smartly created a malicious patch himself. Through this advanced zero-day attack, less people are likely to suspect the attack and more people will consider the update genuine. The update is installed on several cars. The hacker therefore gets control over several of these cars' key features, like the engine and the wheels. At this point, the hacker is able to control almost everything in the car and could take it to any direction.[19]

Although X-Phy's example describes a possible and hypothetic threat, an actual case (different from this example) occurred in 2015 when Charlie Miller and Chris Valasek found the vulnerability in the onboard computer of a Jeep Cherokee. People have been talking for a long time about attacks on such systems if the attackers have access to a diagnostics jack. However, a remote attack on a car's critical systems remained a purely theoretical scenario about which experts have warned for a long time (including experts from Kaspersky Lab). Many hoped that the car manufacturers would recognise the risk of such vulnerabilities being exploited and take preventive measures. The investigators gained access through the onboard entertainment system not only to non-critical settings but also the car's controls like the brakes and accelerator. The investigators plan to publish the technical details of the hack, but the overall scheme of things is already known.[20]

5GAA_gMEC4AUTO_SecurityThreat ST#7 – Zero Day Exploits

[18] https://www.cybersecurity-insiders.com/connected-cars-are-vulnerable-to-ransomware-attacks/

[19] https://x-phy.com/zero-day-exploit-automotive-industry

[20] ibid

4.3 Privacy Threats in gMEC4AUTO architecture

4.3.1. Data Privacy

A certain dataset belonging to a certain user could be disseminated (for any application design reasons) across various MEC servers of the system, and especially in multi-MNO scenarios targeted by 5GAA they can be administered by different players. This data exposure poses a potential privacy issue, since it becomes difficult to track and properly protect the flow of personal and privacy-sensitive information.

Moreover, in a design characterised by the presence of virtualized functions and UE mobility, data can be anywhere in the MEC infrastructure and it is not possible to identify a priori where a MEC application or piece of user data physically resides. So, without appropriate restriction (or traceability measures) on function locations or data locations, privacy sensitive information could move between different trust domains and even different legal jurisdictions (related to possibly different data regulations), making it hard to protect it. In addition, without appropriate lifecycle protection, sensitive information of one virtualized MEC Application could be leaked to another MEC Application reusing the storage resource. So, for example, if a MEC Application moves from one host to another or is terminated, and the previous resources are allocated to another MEC Application without being fully cleared, this could lead to a compromise of privacy sensitive data or keys.

5GAA_gMEC4AUTO_PrivacyThreat PT#1 – Data Privacy

4.3.2 Identity privacy

Identity is the key to safeguarding private information in the cyber-space. Achieving mutual authentication with anonymity and untraceability are crucial for data security and user privacy.

Authentication procedures can imply privacy breaches associated with the disclosure of user information at layers that are not intended to consume certain identity attributes.

In the MEC paradigm, identities from one trust domain will need to mutually authenticate entities across different trust domains. This means that the need for interfaces with secure authentication and authorization mechanisms increases, as does the complexity and operational costs of the public key infrastructures required for the identity and key management. Solutions employed so far based on PKI infrastructures have proven to be technically feasible, but they are centrally managed and domain-specific. On the other hand, MEC is a multi-stakeholder environment with several trust domains and cross-domain touch points and so far, it is not clear if PKI solutions can be a commonly accepted cross-domain IDM concept.

5GAA_gMEC4AUTO_PrivacyThreat PT#2 – Identity Privacy

4.3.3 Location privacy during service migration

To cope with UE mobility, gMEC4AUTO has analysed an Edge Computing service continuity solution divided in three stages: application service retention, application instantiation and data migration, and application service redirection. During the migration stage, when the UE switches to the target RAN, the MEC Orchestrator chooses the target MEC, triggers application instantiation, and applies data migration from the source MEC to the target MEC. The MEC Orchestrator selects the new Edge node for the UE according to the position of the UE.

In this case, there is a risk of user location privacy leakage if a malicious eavesdropper tracks the service migration trajectory, either this is for supporting continuity between Edges within a single MNO network or Multi-MNOs. The location privacy risk stems from the fact that the service migration trajectory overlaps a lot with the user movement trajectory with existing service migration policy. Such an eavesdropper can be a hacker that has gained authority of the MEC system, or an untrusted MEC provider interested in tracking users of a certain service.

5GAA_gMEC4AUTO_PrivacyThreat PT#3 – Location Privacy during service migration

4.4 Trust Concerns in gMEC4AUTO architecture

Edge deployments constitute a complex multi-vendor, multi-supplier, and multi-stakeholder ecosystem lacking a central entity that implements system-wide security assurances or accepts full liability if things go wrong. As a result, this brings to the surface the issue of mutual trust between stakeholders, which we identified in Section 4.2.1, meaning that we cannot make assumptions about the trustworthiness of participating entities and we have to move to the discussion of what is needed to prove that an actor is trusted or not. In fact, this could be solved with a PKI where publicly trusted CAs collaborate cross-borders and areas, but as stated before, it is not clear if PKI solutions can be a commonly accepted cross-domain IDM concept.

Indeed, looking at current architectures, there is an implicit assumption on the trustworthiness of the communicating parties, and there are not enough considerations of the presumable trustworthiness of the environment and the provenance of the services running. At the same time, there are emerging new attack vectors targeting cross-layer vulnerabilities, security misconfigurations and vulnerable and outdated components, which lead us to the need of weakening the above assumptions on trustworthiness.

In this context, MEC introduces the vision of the three-tier paradigm for application design, i.e. client app (e.g. at the vehicle), MEC App (at the edge server) and backend server application (at the remote server, e.g. data centre): The deployment of these three operational layers (with respect to the traditional client-server paradigm) introduces new locations at the edge, where sensitive data is communicated and processed that may adhere to varying security requirements and operate under different trust domains, thus, resulting in new attack vectors to be considered.

For instance, let’s consider two MEC Application Servers (AS1 and AS2 respectively) offering different sets of services: A client (vehicle) that is registered to AS1 and routes its data to the respective edge server, might need to “switch” to AS2 for consuming a specific functionality of interest (e.g., trajectory mapping). However, how can the vehicle client be aware on the security and trust level of the domain operated by AS2? In fact, in principle the edge application from AS2 needs to be considered as possibly misconfigured (running in a “hostile” environment offered by AS2) that once leveraged by the vehicle might endanger the entire system, and, thus, already on-boarded applications from AS1. The same concern also holds from MEC Application Servers’ point of view: AS2 is not aware of the level of trust of the applications (from AS1) running on the vehicle, thus, should inherently be considered as (possibly) compromised.

This poses new trust considerations that need to be captured prior to enabling the service continuity mode that is envisioned through the introduction of MEC. Essentially, the introduction of all these actors requires a paradigm shift from (single) trusted entities (i.e., vehicles) to **trusted networks** considering also the MEC layer. Towards this direction there are already existing solutions for addressing the trust considerations in MEC based on trusted computing technologies (e.g. TPMs, TEEs, or in the form of a RTOS Hypervisor) for enabling device status assurances [36]. However, this doesn’t suffice to address all the trust considerations in the case of gMEC4AUTO. Trying to combine security claims from Roots-of-Trust (RoTs) embedded in each vehicle towards the creation of a network RoT capable of assessing the trust level of all network entities is still unexplored and needs to consider the following aspects and possible threats.

4.4.1 Establishment of Trust among different Application Servers

MEC applications offered by different Application Servers (ASs) are usually grouped together into different trust domains adhering to varying security requirements. This translates to different sets of security assurances that need to be provided by a vehicle that wants to consume such MEC applications offered by different providers. In the current security architectures, such security assurances can be offered through the integration of trusted computing mechanisms (i.e., remote attestation). These security assurances (based on the attestation of a computing platform’s integrity) can directly translate into trust in a vehicle’s capability to protect its information and functional assets (MEC applications) against varying sets of threats.

However, since the deployment of such attestation procedures might be costly, different MEC Application Providers might require different types of security assurances that in turn might lead to a lesser trust domain, within which a vehicle needs to operate, as it is moving between different MEC Application Servers (AS).

For instance, AS1 could require strict security assurances, e.g. verification of the underlying hardware capabilities and operational correctness of running software packages. Another AS2 in this scenario could require a less stringent security assurances where only integrity guarantees of the underlying OS are required. Therefore, for vehicles operating in trust domains with less security requirements (e.g., consuming MEC applications offered by AS2), attackers might exploit “transient malware” capabilities (i.e., subtle software bugs, bitflips, etc.) to compromise the data assets of the vehicle which in turn can lead to the compromise of the applications of the MEC AS1.

In addition, the need for continuous interactive attestation capabilities in such MEC infrastructures (cf. Section 5 on the mitigation measures needed for Multi-MNO environments), considering multiple Verifier workers that should also exchange security reports on a vehicle, can also open up the attack space to exploits due to synchronization issues. For instance, such transient malware can manipulate the verification process only during the attestation of a system property and then command the malware to put the device back in an expected (operational) state before it erases itself. This results in essentially bypassing the authentication process while having a very short time window for being able to detect such exploits. One prominent example is the Time-Of-Check-Time-Of-Use (TOCTOU) attack [1] for which even state-of-the-art remote attestation mechanisms cannot protect against.

Furthermore, privilege escalation attacks can enable an attacker to get access to MEC functions which are above their intended level of privileges. In the context of CCAM (Cooperative, connected and automated mobility)[21], moving to a different MNO Service Provider that enables safety-critical functions (i.e., collision avoidance through intersection assistance), can allow an attacker to manipulate this functional asset by having compromised a less critical operation exploiting such attack path propagation techniques.

5GAA-gMEC4AUTO_TrustConcern_TC#1 – Establishment of Trust among different Application Servers

4.4.2 Secure and Stateful Applications Migration

Continuing in this direction, current deployments consider the instantiation of virtualized Trusted Execution Environments (TEEs) as safeguards for enabling the secure (isolated) execution of safety-critical functions. However, this isolation only targets local execution protection as part of the underlying host (either in the vehicle or the MEC). Considering one of the main benefits of a MEC-enabled infrastructure for supporting the offloading and migration of resource-intensive tasks from the vehicle to the MEC Server (or between MEC Servers), this requires additional trust measures to be set in place. These measures must protect against attacks that try to compromise workload states[22] running inside the TEE as extracted for been migrated to another TEE agent.

Currently, such a migration needs to be mediated by an application (running in the host memory (“untrusted world”)) that can interact with the TEE in order to extract the current state to be migrated (encrypted by the TEE’s internal key). Besides this being a disruptive operation that greatly impacts the operation of a safety-critical operation, it enables an additional attack vector for adversaries to gain access to the “trusted world” of the TEE through this governing application and also compromise the underlying key [3]. Examples include locality-based attacks that can monitor and spoof the Application Performance Counters (APC) caching memory blocks for the mapping between the virtual and physical resources leveraged by the TEE.

[21] Cooperative, connected and automated mobility (CCAM): https://transport.ec.europa.eu/transport-themes/intelligent-transport-systems/cooperative-connected-and-automated-mobility-ccam_en

[22] See also: https://www.intel.com/content/dam/www/public/us/en/documents/white-papers/ecu-consolidation-white-paper.pdf

The latter is especially important during the live migration of workloads between the MEC three-tier architecture. Workload migration requires visibility and interpretation of the state that a device is in, so that it can be "replicated" to the other endpoint where it needs to be instantiated (defined also in the context of "equivalence" requirement for third generation architectures). Attackers may exploit gaps and differences in the models used for interpreting the state of a TEE, so as to bypass this equivalence checking and not only compromise the state but also lead to inconsistencies between the two collaborating TEE agents running in different locations.

5GAA-gMEC4AUTO_TrustConcern_TC#2 – Secure and Stateful Applications Migration

4.4.3 Data Location and Lifecycle

As described also in Section 4.3.5, the integration of MEC-enabled infrastructures introduced multiple locations where sensitive data is shared and processed. Besides privacy issues that this data asset exchange entails, it also poses new trust considerations. In a fully decentralised environment with multiple data sources and data processing units being hosted in different layers of the application stack (from the vehicle to the MEC and the backend Cloud Server), it is difficult to have a high level of certainty where sensitive data might be located. With MEC applications and network functions by design, the applications can run anywhere in the overall infrastructure. Thus, it is of paramount importance to be able to enhance data provenance so as to attach assertions auditing the integrity and correctness of the data lifecycle for safety critical applications. However, these cryptographic assertions are now managed by the virtualized TEE safeguards instantiated in each one of the actors in the MEC paradigm. In a long-term view, keys might be the target of exploitation (cf. Section 4.4.2), and attackers can also exploit lesser trust domains (with no or minimal integrity guarantees on the host functional assets) to recover data.

Furthermore, multiple functions (MEC applications) running in tandem with the TEE Guard also entails threats due to internal workload visibility. A MEC application, originating from an MNO with a smaller LoA, might attempt to manipulate the permission layer at the virtualization layer so as to get access to internal (sensitive) data structures of other workloads running.

5GAA-gMEC4AUTO_TrustConcern_TC#3 – Data Location and Lifecycle

4.4.4 Continuous Authorisation and Authentication

Towards the establishment of trustworthy operational environments, there is a reasonable expectation that all actors establish secure and authenticated communication channels for secure information exchange. In decentralized MEC-enabled infrastructures, this requirement is enriched to also capture the traversing of vehicles between different MNO service providers, thus, resulting in continuous authentication enabled through the exchange of self-issued certificates – leveraging the root-of-trust capabilities of the virtualised TEE safeguard – that embody the output of the needed security claims as attributes. Essentially, this translates to the modelling of trustworthiness controls that need to be exchanged, prior to establishing the desired level of trust for authenticating and authorizing MEC applications, as securely (cryptographically) produced vehicle-issued claims to be integrated to the created certificates as attributes. Such (self-issued) certificates need to be signed by the local functional asset containing dedicated key storage and needs to be protected so that only this specific task can have access to the required key material.

With virtualized implementations, as the ones leveraged in MEC, such key restriction usage policies become much more complicated due to the numerous applications interacting with the virtualized TEE. An (even remote) compromised host can use this TEE as an “oracle” providing valid (signed) certificates, thus, resulting in getting access to safety-critical MEC applications without the necessary privileges. Current security mechanisms furnish the use of crypto primitives towards creating a verifiable chain of trust, but this chain of trust targets essentially authentication, authorisation and identity management purposes and it does not capture chain of trust from the service to the execution environment and to the device hardware and software authenticity and integrity. So, we need to expand the security principle with the “never trust always verify” concept and we need to provide certifiable mechanisms (leveraging HW-based keys and key management and protection measures over virtualized infrastructures) to bootstrap mutual trust between all participating entities in the ecosystem.

5GAA-gMEC4AUTO_TrustConcern_TC#4 – Continuous authorization and authentication

5 Mitigation Strategies for MEC security in Multi-MNO environments

5.1 Data Plane encryption

User data traffic in gMEC4AUTO architecture is transferred across various domains (multiple MNOs, other DNs, interconnections between MEC systems, and finally over radio interface) (see Figure 5.1-1 below). Encryption of data plane is thus required in all these domains, where various available solutions can be exploited, mainly coming from 3GPP specifications for 5G systems.

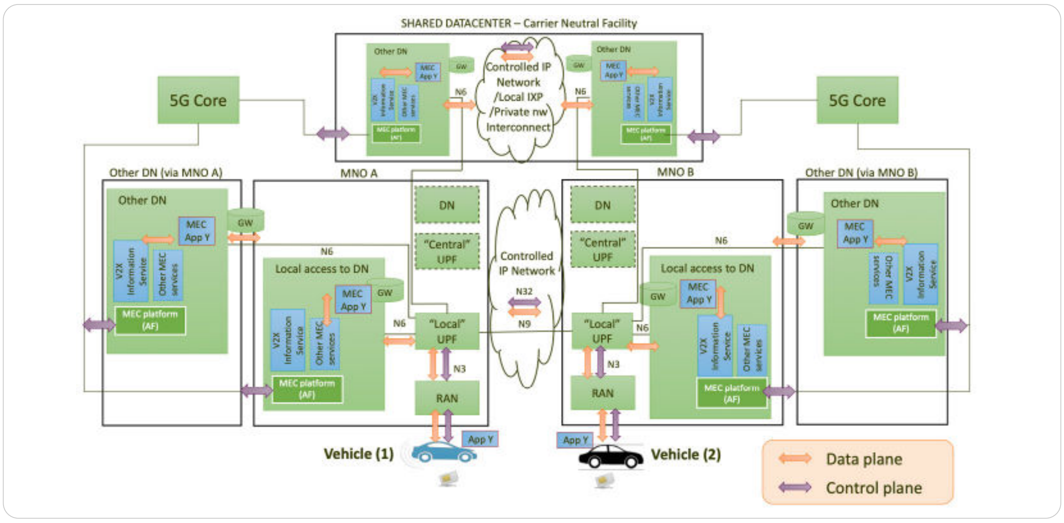


Figure 5.1-1: Data plane and Control Plane in reference MEC architecture in 5GAA gMEC4AUTO [31], including both variants of MNO setup and Neutral Host (NH) setup

More in detail, Data Plan (DP) traffic is given by the follow components:

- Communication over **Uu**, **PC5** and **N3** (i.e. traffic over 5G radio interfaces and toward UPF):
    - The 3GPP standards specify four pairs of algorithms that are used to ensure the confidentiality and integrity of communications between the UE and the RAN and the AMF respectively. The encryption and integrity protection algorithms used by the UE are the same in 5G as in 4G. NEA (5G Encryption Algorithm) is a ciphering algorithm, while NIA (5G Integrity Algorithm) is an integrity algorithm.
    - The encryption of Uu link between UE (in the car) and the gNB is specified by 3GPP in TS 33.501; PC5 data plane encryption over 5G NR (between UEs and between UE and RSU) is defined in TS 33.536: here, the NR PC5 Encryption Key (NRPEK) and NR PC5 Integrity Key (NRPIK) are used in the chosen confidentiality and integrity algorithms respectively for protecting PC5-S signalling, PC5 RRC signalling, and PC5 user plane data.
    - When it comes to the UE traffic through RAN and toward the UPF, N3 is the reference point, which security requirements are specified in TS 33.501. The GPRS tunnelling protocol for the user plane (GTP-U) supports multiplexing of the traffic from different PDU sessions by tunnelling user data over N3 interface (i.e., between the 5G access node and the UPF) in the core network. GTP encapsulates all end-user PDUs and provides encapsulation per-PDU-session. According to TS 33.501, in order to protect the traffic on the N3 reference point, it is required to implement IPsec ESP and IKEv2 certificate-based authentication. IPsec is mandatory to implement on the gNB. On the core network side, a SEG may be used to terminate the IPsec tunnel.
- Communication over **N6** (traffic from UPF to MEC, which is sitting at the DN): here the reference 3GPP specification is TS 33.501 which states that data between UPF and DN is encrypted (Clause 12 and Clause T.2).
- The communication between MEC Application and MEC platform may include many kinds of data, including potentially user context data and information needed by the application to run; ETSI ISG MEC standardizes a variety of MEC services by specifying implementation agnostic, RESTful APIs using HTTP. When it comes to API consumption, The ETSI GS MEC 009 specification [20] defines design principles for RESTful MEC service APIs (e.g., V2X API). These general principles defined in MEC 009 apply for all the APIs using **Mp1** reference point between MEC Applications and MEC platform (thus applicable also to service producing MEC Applications exposing their services via the MEC platform). Here, the specification mandates support for HTTP over TLS (also known as **HTTPS**) using **TLS** version 1.2 (as defined by IETF RFC 5246). TLS version 1.3, defined by IETF RFC 8446, should be also supported. The specifications explicitly prohibit the use of HTTP without TLS or TLS versions preceding version 1.2.
- From a 3GPP perspective, the consumption of MEC APIs is realized via **CAPIF** (Common API Framework). When the NEF supports CAPIF for external exposure as specified in clause 6.2.5.1 in TS 23.501, then CAPIF core function shall choose the appropriate CAPIF-2e security method as defined in the sub-clause 6.5.2 in TS 33.122 for mutual authentication and protection of the NEF-AF interface.

- Communication over N9 (supporting the controlled IP network, to connect multiple operator domains): here again the communication between different MNOs is assumed to be encrypted; the 5G System architecture introduces Inter-PLMN UP Security (**IPUPS**) at the perimeter of the PLMN for protecting user plane messages (ref. TS 23.501). The IPUPS is a functionality of the UPF that enforces GTP-U security on the N9 interface between UPFs of the visited and home PLMNs. Note that IPUPS can be activated with other functionality in a UPF or activated in a UPF that is dedicated to be used for IPUPS functionality (ref. TS 23.501, clause 5.8.2.14).

5GAA-gMEC4AUTO_Mitigation_Strategy MS#1 – Data Plane encryption

5.2 Security on Control Plane

Control plane traffic in gMEC4AUTO architecture is transferred across various domains (multiple MNOs, other DNs, interconnections between MEC systems, and finally over radio interface) (see above Figure 5.2-1). Encryption of control plane is thus required in all these domains, where various available solutions can be exploited, mainly coming from 3GPP specifications for 5G systems (in this perspective, a major reference in 3GPP is represented by TS 33.501, which specifies the 5G security architecture, i.e., the security features and the security mechanisms for the 5G System and the 5G Core, and the security procedures performed within the 5G System including the 5G Core and the 5G New Radio).

More in details:

- The 3GPP standard specifies the use of **IPsec** and **(D)TLS** for some of the communications between the gNBs and the 5GC or between entities of the 5GC. For non-service based interfaces between the (R)AN and the 5GC and inside the 5GC, the connection is expected to be secured using IPsec with protection profiles defined in clauses 4 and 5 of TS 33.210. Among other interfaces, all service based interfaces shall be protected using TLS. Currently, the 5G core network functions support state-of-the-art security protocols like TLS 1.2 and 1.3 to protect the communication at the transport layer, and the **OAuth 2.0** framework at the application layer to ensure that only authorized network functions are granted access to a service offered by another function.
- When it comes to MEC management system ETSI GS MEC 010-2 defines the RESTful resources and operations over reference point Mm1 and Mm3 APIs for **application package management** and **application life cycle management**. Here, as well as other MEC service APIs, and in accordance with MEC 009, these management APIs shall support HTTP over TLS (also known as HTTPS) using **TLS** version 1.2 as defined by IETF RFC 5246 [10]. TLS 1.3 (including the new specific requirements for TLS 1.2 implementations) defined by IETF RFC 8446 [14] should be supported. HTTP without TLS shall not be used. Versions of TLS earlier than 1.2 shall neither be supported nor used.
- Considering the needed information exchange between MEC (acting as an AF) and **5G core network**, the 3GPP specification TS 33.501 mandates the usage of **TLS** to provide integrity protection, replay protection and confidentiality protection for the interface between the NEF and the AF. In fact, in typical cases, MEC is outside the PLMN trusts domain, and thus all communication from MEC Platform and 5GC should be carried out via NEF. Here, the support of TLS is mandatory. More in detail, after

the authentication, NEF determines whether the AF is authorized to send requests for the 3GPP Network Entity. The NEF shall authorize the requests from AF using **OAuth**-based authorization mechanism, the specific authorization mechanisms shall follow the provisions given in RFC 6749.

- One of the most critical information exchanged during management procedures is the UE identity. When it comes to radio interface between UE and RAN, the 3GPP standard specifies three encryption schemes to obfuscate the SUPI (which is used to identify the UE between network functions); the **SUPI obfuscation** mechanism can be achieved using two elliptic curve-based protection schemes (called ECIES and both using AES-128 for the symmetric encryption of the of the data).
- The 5G System architecture also introduces a Security Edge Protection Proxy (**SEPP**) as an entity sitting at the perimeter of the PLMN for protecting control plane messages. The SEPP enforces inter-PLMN security on the N32 interface.

5GAA-gMEC4AUTO_Mitigation_Strategy MS#2 – Security on Control Plane

5.3 Security of containers

MEC deployments are based on virtualized infrastructure, where ETSI NFV framework is a standard reference, being adopted also by 3GPP for 5G systems. In NFV systems, MEC components can be seen as VFNs (Virtual Network Functions). However, practical implementation of MEC Applications are not limited to VM (Virtual Machine), which in some case could be not suitable for the needs of mobility and flexibility at the edge, for example for life-cycle management of applications [21]. So, MEC applications can run on alternative virtualization technologies, such as containers. By the way, as ETSI NFV is also working on alternative virtualization technologies, the MEC work should be aligned with NFV where applicable.

State of the art solutions on container support are already covered by many initiatives, where also security of containers in some cases is being considered (even if still the industry standards are working on this):

- **3GPP:** in the context of security, the relevant WG in charge of this work is SA3. The TR 33.848 is a technical report on Security Impacts of Virtualisation. Also, TR 33.818 (Security Assurance Methodology (SECAM)) is leading to the introduction of a set of Security Assurance Specifications (SCAS) for 3GPP virtualized network products. It analyses threats related to the integration of ETSI VNF concepts and interfaces within the 3GPP virtualized system (e.g. interface between 3GPP VNF and VNFM, virtualization layer and HW, and between virtualization layer and the VIM).
- **ETSI NFV:** the specification ETSI GR NFV-SEC 009 seeks to provide methods, capabilities, procedures and assurances of various strengths based on requirements and available technologies and techniques - that safeguard Virtual Machines or Containers running on a virtualization host. ETSI GS NFV-SEC 012 takes this further by defining additional security requirements for sensitive functions (or components within larger functions), such as the use of Hardware Mediated Execution Environments (HMEEs). Sensitive components include cryptographic tunnel end points, security functions and Lawful Interception functions. A list of technologies and measures from various domains is provided to meet the requirements of the various

use cases, including memory inspection, secure logging, OS-level access control, secure storage, etc. NFV security WG is also exploring further features in the current release to make the NFVI secure enough for wide range deployment scenarios and use cases. The related specifications include container security defined in ETSI GS NFV-SEC 023, isolation and trust domain defined in ETSI GS NFV-SEC 026 and NFVI security assurance defined in ETSI GR NFV-SEC 027.

- **ETSI MEC:** Container technology is increasingly valued by MEC application developers. MEC will likely be introduced step by step into containerized MEC applications and container management platforms. Containers are used to package various applications and provide a unified development, testing, and production environment for upper-layer applications. The study ETSI GR MEC 027 focuses on identifying the additional support that needs to be provided by MEC when MEC applications run on alternative virtualization technologies, such as containers. The document collects and analyses the use cases relating to the deployment of such alternative virtualization technologies, evaluates the gaps from the currently defined MEC functionalities, and identifies new recommendations. From a security perspective, currently in ETSI MEC there is a Work Item on MEC Security (ETSI GR MEC 041), studying security topics and paradigms that apply to MEC deployments. The study will broadly cover the themes of application and platform security, Zero-Trust Networking, and security requirements for MEC Federations. It may also draw upon prior work from other standards and gather requirements from industry associations (e.g., 5GAA). It will identify gaps in ETSI MEC and provide recommendations for new normative work.

- **SASE:** Secure Access Service Edge (SASE) was brought up by Gartner in 2019 to define a category of hardware and services used to enable edge security. MEF (Metro Ethernet Forum), as a global industry association of network, cloud, and technology providers, is working on ”standardizing” SASE offering for such managed edge service providers. In this context, MEF 118 (published in October 2022) has introduced a “Zero Trust Framework for MEF Services”, where each virtual machines, containers or microservices composing a MEC Application could be considered an Application Actor for which Policies can be applied. Finally, Application Actors also include their respective Application Programming Interfaces (APIs) since they are a part of the Application. This Zero Trust Framework thus defines Actors and associated Service Attributes used to create Policy criteria. These Service Attributes can subsequently be changed via negotiations but all such changes must be subject to continuous monitoring and tamper proof Audit Event logging requirements. The Service Attributes must be encrypted (recommended NIST SP 800-175B), both at rest and in transit.

5GAA-gMEC4AUTO_Mitigation_Strategy MS#3 – Security of containers

5.4 Identity and authentication

In general, authentication between the communication endpoints is needed in order to mitigate spoofing of messages. As we mentioned in several points above, TS 33.501 mandates that mutual authentication and transport security between network functions is based on TLS 1.2 and 1.3. This is complemented by token-based authorization based on OAuth 2.0, as we will see in the next section.

The usage of both TLS and OAuth 2.0 relies on the use of a **Public-Key Infrastructure (PKI)** that has to be in place. In a PKI, a Certificate Authority (CA) issues certificates to each of the communication endpoints guided by proper (identity) management functions and policies. The public/private key pairs associated with the certificate can then be used for the asymmetric cryptography used in mutual authentication and signing/verifying of tokens that is required for using TLS and OAuth 2.0. However, TS 33.501 does not examine the case of different trust domains and multi-vendor NFs and leaves several details open on how to provision certificates and how to setup commonly trusted certification authorities (CAs).

In general, in this heterogeneous environment of gMEC4AUTO, identities are assigned to each entity in a single trust domain but permits entities to mutually authenticate other entities across different trust domains. At the same time, considering the flexibility of the MEC architecture, achieving mutual authentication should also consider privacy requirements such as anonymity and untraceability. Such mechanisms have been discussed by academic work [5], but they have not been elaborated sufficiently in industry and standards.

5GAA-gMEC4AUTO_Mitigation_Strategy MS#4 – Identity and authentication

5.5 OAuth 2.0

The MEC platform should authenticate all MEC application instances and only provide them with the information for which the application is authorized. MEC specifications mandate the use of the OAuth 2.0 for authorization of access to RESTful MEC service APIs defined by ETSI ISG MEC. The implementation of the OAuth 2.0 authorization protocol uses the client credentials grant type according to IETF RFC 6749 and with bearer tokens according to IETF RFC 6750.

OAuth 2.0 has the optional provision for scopes which may be defined at the level of resources, combinations of resources and methods, or combinations of resources and methods with specific values for parameters, or values of attributes in the payload body. For subscriptions, the subscription type can be used to scope the authorization and is expressed as a string named the permission identifier. Definitions of permission identifiers thus accompany MEC service specifications. The available authorization scopes for a service are made known to clients during service discovery.

In place of OAuth 2.0 standard bearer tokens, AA entities in MEC deployments may also hand out JSON Web Tokens (JWT) (IETF RFC 7519). JWT have a compact representation and an extensible structure to directly encode application defined claims and entitlements into the access token. These may include OAuth scopes or roles to restrict access to MEC services. JWTs may be signed by the AA entity to become tamper proof and encrypted to not leak any application metadata. Thus, JWTs are self-contained and self-verifiable access tokens. A practical benefit of using JWT bearer access tokens is they allow fully decentralized and stateless enforcement of API security, not needing to continually query the AA entity upon requests to MEC services, thereby yielding a performance gain.

5GAA-gMEC4AUTO_Mitigation_Strategy MS#5 – OAuth 2.0

5.6 User Application LCM proxy

ETSI GS MEC 003 [22] specifies the MEC architecture, where a functional block is introduced, called User application lifecycle management proxy (UALCM proxy), to deal with requests coming from device applications via the external reference point Mx2.

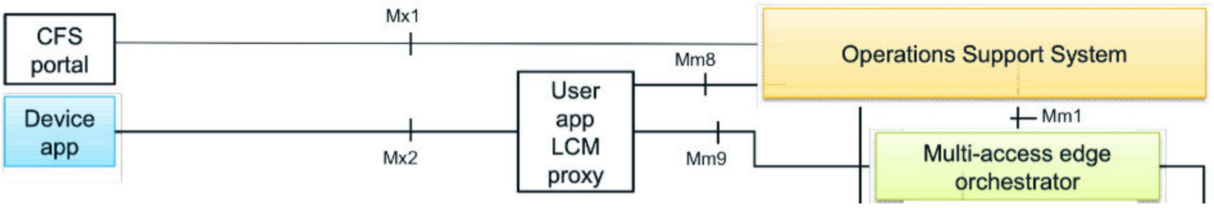


Figure 5.10-1: User app LCM proxy (at the MEC system level) [22]

In fact, device applications can request on-boarding, instantiation, termination of "user applications" and when supported, relocations in and out of the MEC system (note: in MEC 003, the "user application" is simply defined as a MEC application that is instantiated in the MEC system in response to a request of a user via an application running in the device, i.e. the device application). This UALCM proxy is also allows informing the device applications about the state of the user applications, and finally authorizes requests from device applications in the device (e.g. UE, laptop with internet connectivity) and interacts with the OSS and the MEO for further processing of these requests.

As a consequence, the UALCM proxy (available when supported by the MEC system) is practically filtering incoming requests (from Mx2) directed to the MEC system (respectively Mm8, toward the OSS, and Mm9, toward the MEC Orchestrator). This block is also present in the synergized architecture supported by ETSI MEC and 3GPP SA6 [23], as part of the management and orchestration based on ETSI MEC. Also, the usage of UALCM proxy (UALCMP) is specified on MEC 016 [24], describing how the device application interacts with the UALCMP over the UE application Interface. In this perspective, the device application presents the access token to the UALCMP with every request in order to assert that it is allowed to access the resource with the particular method it invokes. The access token is included in the «Authorization» request header field as a bearer token according to IETF RFC 6750. Also for this reference point Mx2, OAuth 2.0 is applied on UE application interface, where an authentication and authorization entity is assumed to be available for both the REST client, i.e. the device application, and the REST server represented by the UALCMP.

5GAA-gMEC4AUTO_Mitigation_Strategy MS#6 – User Application LCM proxy

5.7 Security Credential Management

Public-key-infrastructures (PKI) can be considered a standard method to model trust between different entities and this generic concept has been used in C–V2X systems to protect the direct communication of the PC5/V5 interface between the C-V2X devices in the application layer. IEEE 1609.2 [2] defines the format and processing of the security message of the C-V2X system. The corresponding C-V2X security management system in US, Europe and China is based on the above standard and designed according to their actual conditions and management requirements.

More specifically, the Security Credential Management System (SCMS) [3] is a product of vehicle OEM consortia and the US Department of Transport (USDOT), and the European Cooperative Intelligent Transport Systems (C-ITS) is developed by CEN and ETSI with support from the European Commission [4]. Furthermore, 5GAA has evaluated the SCMS and the C-ITS system designs and it has concluded that they can be improved to take advantage of cellular connectivity. The effort to identify potential design simplifications to increase efficiency and harmonize technologies across regions has resulted in an updated system design for large-scale deployment and cross-regional interoperability called “Efficient Security Provisioning System” (ESPS) [5].

Broadly speaking, in all of the above systems, Privacy and Cyber Security features have been realized by design by defining the Certificate and Security Policy based on PKI management and pseudonymizing of the messages.

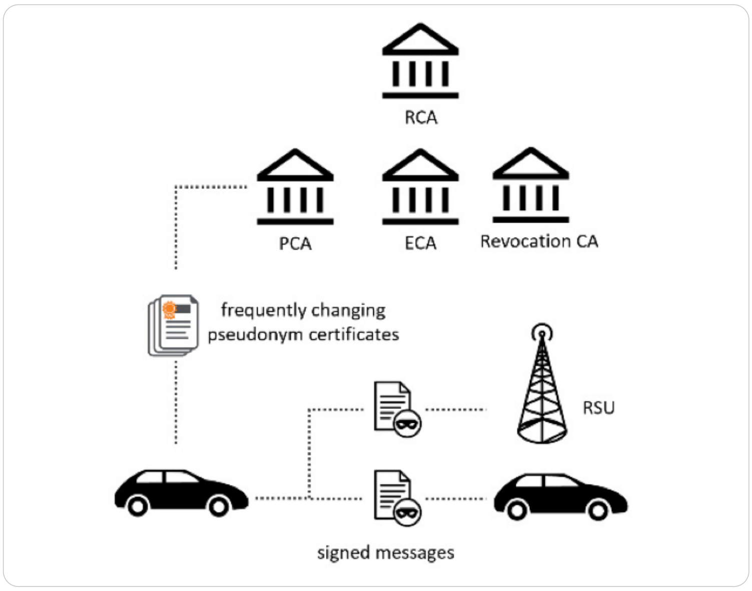


Figure 5.8-1: A V2X security solution based on PKI

The question then becomes, how the vehicles are provided with the set of pseudonyms. In the PKI approach, a set of certification authorities (CAs) provide credentials to the vehicles. In the general case, there is a set of different authorities with distinct roles:

- Root Certificate Authority (RCA): This entity is the trust anchor of the PKI that is responsible for issuing certificates to sub-CAs. The certificate of the RCA is signed by itself.

- Enrolment Certification Authority (ECA): This entity is responsible for registering vehicles and issuing long-term certificates. Entities with enrolment certificates can then apply to other CAs, like for example to the pseudonym CA for issuing pseudonym certificates.
- Pseudonym Certification Authority (PCA): This entity is responsible for issuing certificates that do not contain any identifying information.
- Certificate Revocation CA: responsible for issuing certificate revocation list for all kinds of certificates.

Adopting this mechanism, C-V2X can achieve several privacy properties as explained by [6]:

- Minimum disclosure: The amount of information revealed by a user in a communication is kept to the minimum and is no more than what is required for the normal operation of the system.
- Conditional Anonymity: Individual vehicles are anonymous within a set of potential participants. If a vehicle deviates from system policies, the corresponding long-term identity can be retrieved by the PKI entities, and revoked temporarily or on a permanent basis.
- Unlinkability: No entity is able to link the different pseudonyms of a specific vehicle with each other.
- Forward and backward privacy: The revocation of a credential does not affect the unlinkability of previously signed messages. Also, if an attacker recovers the identity of the sender of a particular credential, it does not affect the privacy of other messages signed by the same sender.

Regarding the Uu Interface, the C-V2X security adopts the security mechanism (3GPP TS33.401, v.15.6.0 2018) provided by the mobile cellular system. The core network of the 5G system does not introduce new network elements and network functions for the C-V2X system. Therefore, the core network of the 5G system adopts the security mechanism provided by the 5G system to realize the interface security related to C-V2X. Again here, certificate-based authentication with TLS 1.2/1.3 and X.509 certificates is the preferred solution for authenticity checks in 5G, utilizing the existence of a certificate authority (CA) that it securely and reliably keeps care of the validity of the digital identities in the network.

5GAA-gMEC4AUTO_Mitigation_Strategy MS#7 – Security Credential Management

5.8 Misbehavior Detection

Misbehavior Detection (MBD) is another concept to detect and exclude faulty or malicious agents from a system and this is, too, highly related to trust management. Indeed, the PKI solution described above is not enough on its own to guarantee that the information in V2X packets is indeed correct and trustworthy, i.e., bad actors and/or malfunctioning devices with valid credentials can flood the V2X network with bad and even harmful data, which is what we call misbehavior. More concretely, misbehavior within the V2X network refers to the willful or inadvertent transmission of incorrect data.

Incorrect information in V2X packets could be caused by a number of things, including but not limited to the malicious attacks discussed earlier. Faulty onboard components, temporary malfunction of a sensor, such as GPS not working inside a tunnel, or cameras failing to detect objects in poorly lit condition, could result in misbehaving packets as well. Whatever the cause or motivation, the net effect is the same – the information sent out in a V2X packet is substantially different from the physical reality, and is beyond the nominal margin of error considered acceptable by the industry.

At a high level, the approach to misbehavior detection and remediation is as follows:

3 Local Misbehavior Detection (LMBD): Detect misbehavior locally at the device level.

3 Misbehavior Reporting: Report the misbehavior to a central authority in the Public Key Infrastructure (PKI) called a Misbehavior Authority (MA).

3 Global Misbehavior Detection (GMBD): Investigate and corroborate misbehavior reports at the MA level.

3 Misbehavior Remediation: Take remediation actions, such as revocation, deny-listing, software update, etc.

Among all SDOs, the European Telecommunications Standards Institute (ETSI) is leading the standardisation effort for specifying a cybersecurity system against V2X misbehaviour attacks. The outcome of this effort are two documents identified as ETSI Technical Report (TR) 103 460 [5] and Technical Specification (TS) 103 759 [6]. The latter is a standard under development that defines a V2X misbehaviour detection and reporting system for a subset of V2X message types: Cooperative Awareness Message (CAM) and Decentralized Environmental Notification Message (DENM).

5GAA-gMEC4AUTO_Mitigation_Strategy MS#8 – Misbehavior Detection

5.9 Attestation and Hardware Root of Trust

Trusted (or Confidential) Computing is a core enabler for measuring and validating the trustworthiness and resilience of a system (through hardware enhancement) against a wide set of attack vectors targeting its integrity and the confidence of other C-ITS actors for reliance on that entity to fulfil specific responsibilities. In this context, both authentication (a process of ensuring that the computing platform can prove that it is what it claims to be) and attestation (the process of proving that a computing platform – or a set of devices in the case of collective attestation schemes - is trustworthy and has not been breached) are prominent security constructions for creating assurances on platform integrity state and the ability to protect data in accordance to various security levels and policies. More specifically, attestation is the process through which a remote challenger can retrieve verifiable information regarding a platform's configuration and execution state (as described in TCG PC Client Specific Implementation Specification [3]). These can be divided into two general classes. On the one hand, **static attestation protocols** capture the memory content of a device, typically at boot time or load time of a software to be verified. This allows to detect whether the software (or their configuration files loaded) have been manipulated. **Runtime attestation schemes** capture the dynamic state of a Prover by tracing the way the software on the Prover is actually executing. This allows to detect attacks that only change the dynamic behaviour of a program but not the program itself, such as Return-Oriented programming (ROP) attacks. In all those schemes the common trust indicators are those software measurements that capture the behaviour of a system (comprising the Chain of Trust) to be validated against trusted behavioural profiles.

However, the information in the measurements log is not sufficient to enable a trustworthy assessment of the platform's integrity state. The measurement log is generated by the very software running on the platform being assessed. Therefore, trust in data contained in the measurements log must be ensured as well as the freshness of the state information. This implies that the mechanism used to capture the state on the prover device must be trustworthy. Further, the state information must provide all information for the verifier to make judgments regarding the integrity of the prover device, i.e., depending on which types of attacks should be detected all relevant information about the prover's state must be reported (e.g., to detect ROP attacks information about the prover's execution paths must be included). All these are supported by using inherently trusted primitives enabled into the host platform. The primitives are called Roots of Trust (RoT) which are trust components supporting services including: software measurement service for the Root of Trust for Measurement (RTM); software measurement and measurement validation service for the Root of Trust for Verification (RTV); access controlled and tamper evident or tamper resistant protected storage service for the Root of Trust for Storage (RTS) certification service (providing cryptographic proof that a set of data originates from the RTS) for the Root of Trust for Reporting (RTR). These can be SW-, HW- and or virtual-based depending on the security requirements need to be considered in the underlying platform, i.e., security leveraging HW-based keys offers stronger guarantees on the device integrity and can provide enhanced resilience and code confidentiality through protected secret keys. Examples of such RoT solutions include TPM (Trusted Platform Module), TEE (Trusted Execution Environment), DICE (Device Identifier Composition Engine) and PUFs (Physical Unclonable Functions), and others.

All approaches have in common that they typically use a challenge-response protocol to ensure freshness of attestation (and to minimize the TOCTOU [1] problems). At the same time, non-interactive protocols have been proposed as well to mitigate DoS attacks. They rely on a trust anchor to act as reliable entity to trigger the capturing of the prover's state.

The security stack of existing attestation mechanisms is summarized in the following table:

Layer	Key Security Measures
Management	Secured APIs
Data	Encryption, Signature, Metadata
OS	Secure Boot, Hardening
Trusted Component	UEFI, TPM, HSM, Sanctum, SPDM

5GAA-gMEC4AUTO_Mitigation_Strategy MS#9 – Attestation and Hardware Root of Trust (HW RoT)

5.10 “Chip-to-Cloud” Assurance Solutions in NFV environments

Validating trust is the exercise of going through the various measures of trust applicable for a particular trust relationship, evaluating the levels of trust assurance and, if they meet the criteria set, validating that trust relationship. Trusted Computing and attestation-enabled schemes are also extended to provide the means for assessing the trustworthiness and Level of Assurance (LoA) of not only embedded systems and edge devices but that also of MEC Services deployed through Virtual Functions (VFs). This trust continuum between vehicle and MEC leverages secure “chip-to-cloud” assurance solutions based on the use of attestation mechanisms for verifying the NF environment including both all the infrastructural elements, and the software components running on them, and especially the MEC Services actually deployed.

To also capture the different security requirements posed by different Application Providers (APs), there are security services (offered by the underlying RoT) that can translate to varying Levels of Assurances for a service’s integrity. This unlocks the efficient establishment of trust relationships and the secure on-boarding and hand over of vehicles, even between APs belonging into different trust domains, by mapping specific software and hardware measurements to a pre-defined scale of trustworthiness based on the attack vectors that can be detected by assessing a specific measurement. For instance, capturing the static content of a VF’s memory can enable configuration integrity but no execution integrity which in turn can be validated through (runtime) control-flow attestation for having security guarantees against ROP attacks but not DOP attacks. This scale consists of a set of specific Levels of Assurance (LoA) that can be applicable to establish the trustworthiness of a particular set of components (e.g. system or platform), according to the nature of the requested service, the threats being considered, and the applicable policies at all levels, from legal requirements to commercial SLAs. Six Levels of Assurance have already been defined in ETSI [4] for NFVs, named with a number that characterizes each one in a relative scale of trust, where a higher number implies a higher degree of trust.

This ability also captures the need for enforcing specified trust relationships for and between the virtualization resources for End-to-End Trust Lifecycle Management. This essentially facilitates decisions on trust made on a multitude of parameters including the geographical and regulatory location of the AP; hardware capabilities and provenance of the underlying physical resources; software capabilities and provenance of those processes and items comprising a MEC Service; execution chain of trust; time elapsed since last audit; ownership of hardware; ownership of software; appropriate use of encryption techniques; software hardening; and physical security.

5GAA-gMEC4AUTO_Mitigation_Strategy MS#10 – “Chip-to-Cloud” Assurance Solutions in NFV environments

5.11 Secure Migration Service

Existing MEC deployments integrate RoTs (especially virtualization-based TEE extensions) for verifying that services are offered by trustworthy systems. However, in the context of services that target the resource and execution capabilities enrichment of edge devices (through offloading), trust verification mechanisms also support the secure migration of tasks and states of “edge-running” workloads from the vehicle to the MEC Server (or between MEC Servers) (Section 4.4.2). Such a functionality is currently supported only through centralized architecture requiring the existence of a trusted backend-server for facilitating the key exchange between the two enclaves (instantiated in different hosts) that wish to share data and/or states (data and state migration). Otherwise, the target enclave will not be able to unseal the received data. Besides the challenge of establishing a symmetric key between two enclaves that might be running on different hardware capabilities (thus, be susceptible to interoperability issues in a MEC-enabled environment) such a solution also makes strong trust assumptions on the backend-server which if compromised (or “honest-but-curious”) could compromise the intermediate key establishment process.

5GAA-gMEC4AUTO_Mitigation_Strategy MS#11 – Secure Migration Service

6. Recommendations for MEC automotive deployments in Multi-MNO environments

After an overview (in Section 4) of the main threats identified for the gMEC4AUTO architecture, Section 5 provided an overview of Mitigation Strategies for MEC security in Multi-MNO environments, as a toolbox of solutions available from standards or from industry-led consolidated implementations. Here, the goal is to analyze the various threats and how the various mitigation strategies can address them in gMEC4AUTO architecture, with the aim to draw some considerations and possible recommendations on Future Work. In Section 5 a number of mitigation strategies have been identified:

- MS#1 – Data Plane encryption
- MS#2 – Security on Control Plane
- MS#3 – Security of containers
- MS#4 – Identity and authentication
- MS#5 – OAuth 2.0
- MS#6 – User Application LCM proxy
- MS#7 – Security Credential Management
- MS#8 – Misbehavior Detection
- MS#9 – Attestation and Hardware Root of Trust (HW RoT)
- MS#10 – "Chip-to-Cloud" Assurance Solutions in NFV environments
- MS#11 – Secure Migration Service

6.1 Security considerations

ST#1 (Malware): relevant strategies in this context are: MS#9, #10, #3, #5. In the context of malware attacks, some available tools can be applicable in the context of gMEC4AUTO architecture, including the usage of attestation, HW RoT, "Chip-to-Cloud" assurance Solutions in NFV environments and securitization of containers, to protect the virtual and physical infrastructure that are hosting the data. Also, authentication mechanisms using OAuth 2.0 can be suitably verify the identity of who has access to the actual data to be protected. In this perspective no specific recommendations are made, but the simple consideration that current tools available can concur to mitigate this thread. Also, no specific gaps are identified, on the technology side. Instead, international standard bodies (e.g. ETSI, 3GPP) should work more and progress on the normative work in order to produce interoperable standards that can be used in federation scenarios targeted by gMEC4AUTO. In summary, the above strategies available can provide a good mitigation against risk of malware, to guarantee a certain level of security. However, to determine which level of security (and associated to a certain level of confidence) is sufficient wrt certain application use case (and related business needs), it is certainly something out of the scope of the present document.

ST#2 (Man in the Middle): relevant strategies in this context are: MS#10, #1, #5. Here, the above strategies available can provide a good mitigation against risk of "Man in the Middle". No specific recommendations are derived, from 5GAA perspective, nor are any particular gaps identified, as at a first place the above available tools should be adopted in multi-stakeholders MEC environments, to guarantee a certain level of security. However, also in this case, to determine which level of security (and associated to a certain level of confidence) is sufficient wrt certain application use case (and related business needs), it is something out of the scope of this document.

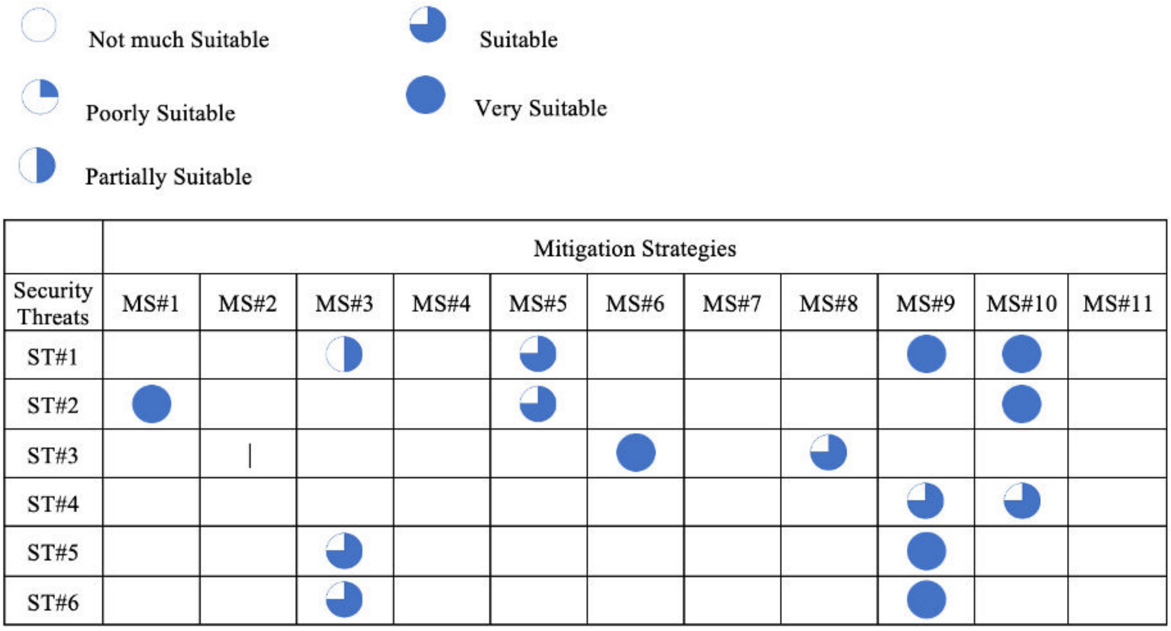
ST#3 (Denial of Service): relevant strategies in this context are: MS#6, #8. Also here, the above strategies available can provide a good mitigation against risk of "Denial of Service" (DoS) or Distributed-DoS (DDoS). Here, the specific advantage of MEC environments (wrt to traditional internet environments with centralized computing and storage entities), is that traditional cloud configurations are more prone to Denial of Service (DDoS) assaults and power outages. Instead, by its decentralized nature, edge computing disperses processing and storage, making systems less susceptible to outages and downtime. Since most procedures occur locally, hackers cannot intercept data in transit. Even if an intruder hacks a single machine, the attacker can only access the data on that computer. In this context, the available tools listed in the above are believed as appropriate to mitigate this DoD/DDoS risks, especially in the scenarios targeted targeted by gMEC4AUTO. Nonetheless, it is worth mentioning that there are also threats and related counter-measures for DNS attacks, as well as in traditional non-MEC environments. In summary, no specific recommendations are derived, from 5GAA perspective, nor are any particular gaps identified, as at a first place the above available tools should be adopted in multi-stakeholders MEC environments, to guarantee a certain level of security. However, also in this case, to determine which level of security (and associated to a certain level of confidence) is sufficient wrt certain application use case (and related business needs), it is something out of the scope of this document.

ST#4 (Advanced Persistence): relevant strategies in this context are: MS#9, #10. This security threat is pertaining more to the vehicle, as cars can be often subjected to attacks. So, the threat is actually less related to network/cloud/edge. In any case, the above strategies available can provide a good mitigation against risk of "Advanced Persistence". No specific recommendations are derived, from 5GAA perspective, nor are any particular gaps identified (with the same considerations as above, on the level of security and related confidence).

ST#5 (Ransomware): relevant strategies in this context are: MS#9, #3. This kind of threat is also typical for any other IT environments where data needs to be protected, both in rest and in movement. So, the solutions using MEC should leverage the same mitigation strategies conceived for any other data center or cloud computing system, using datasets from customers and/or treating any set of user data (both in rest and in movement). Thus, in general, the above strategies available can provide a good mitigation against risk of "Ransomware". No specific recommendations are derived, from 5GAA perspective, nor are any particular gaps identified (with the same considerations as above, on the level of security and related confidence).

ST#6 (Zero Day Exploits): relevant strategies in this context are: MS#9, #3. Also this kind of threat is also typical for any other IT environments where data needs to be protected, both in rest and in movement. Same considerations as above (on the level of security and related confidence) apply.

In summary, in order to present the reader a map of **security threats** and possible coverage of existing tools and available mitigation strategies, the table below shows a qualitative assessment of the *suitability* of these tools to the specific needs of MEC environments, and especially the even more specific needs of the gMEC4AUTO targeted scenarios. The table represents also a way to give a picture of complementarities and potential overlaps, showing where more elaboration of the tools might be needed, and/or more standard work could be needed or simply further discussed (in these cases, such investigation is out of the scope of the present document, but a dialogue among experts on MEC and on cybersecurity fields may be needed). Here below is the legenda related to the following table:



○ Not much Suitable

◔ Poorly Suitable

◑ Partially Suitable

◕ Suitable

● Very Suitable

	Mitigation Strategies										
Security Threats	MS#1	MS#2	MS#3	MS#4	MS#5	MS#6	MS#7	MS#8	MS#9	MS#10	MS#11
ST#1			◑		◕				●	●	
ST#2	●				◕					●	
ST#3		|				●		◕			
ST#4									◕	◕	
ST#5			◕						●		
ST#6			◕						●		

6.2 Privacy considerations

PT#1 (Data Privacy): relevant strategies in this context are: MS#1, #2, #3, #5. In the context of data privacy threats, some available tools can be applicable in the context of gMEC4AUTO architecture, starting from the encryption (including both strategies for Data Plane and Control Plane), to authorization mechanisms (e.g. via OAuth 2.0) and security of containers. However, in the context of gMEC4AUTO still the gap is about identifying (and standardizing) the information exchange among the federating entities, so that data privacy is ensured. Being able to track the flow of data is also an important aspect.

PT#2 (Identity Privacy): relevant strategies in this context are: MS#1, #2, #3, #5, #6, #4. In the context of identity privacy threats, some available tools can be applicable in the context of gMEC4AUTO architecture, e.g. additional mechanisms for identity management at application level, in the establishment of federation, and finally also during operations. ETSI MEC and 3GPP are already providing token-based mechanisms to preserve the privacy of the user, simply by using a token representing that UE, instead of transmitting the actual User ID (see also MEC 014 "UE Identity API" [25]). However, still standards need to further work on this area, in order to fully cover the specific needs or preserving user identity, while at the same time being capable to offer ubiquitous and global MEC services in multi-MNO, Multi-OEC and multi-vendor environments. In this context, there is also the concurrent need from operators and service providers to establish agreements on how to treat user identity in the context of federation. However, this aspect is outside of the scope of the present document, and pertains to the level business agreements related to the MEC Federation (a recent example is given by the request from some operator to the European Commission to establish a Joint Venture[23] for offering a privacy-led, digital identification solution to support the digital marketing and advertising activities of brands and publishers; such initiatives, if established may generate a secure, pseudonymized token derived from a hashed/encrypted pseudonymous internal identity linked to a user's network subscription which will be provided by participating network operators).

PT#3 (Location Privacy during service migration): relevant strategies in this context are: MS#1, #2, #3, #5, #6, #4. In the context of location privacy threats, some available tools can be applicable in the context of gMEC4AUTO architecture, e.g. additional mechanisms for identity management at application level, in the establishment of federation, and finally also during operations. Also in this case, even if there is some ongoing work in the standards to address these needs, still more work is needed on this area, in order to fully cover the specific needs or preserving location privacy, while at the same time being capable to offer ubiquitous and global MEC services in multi-MNO, Multi-OEC and multi-vendor environments. In particular, the need to offer MEC services across a federation could be in contrast with the concept of "separation of concerns" expressed by service providers in GSMA OPG [26] (see also the requirements for Operator Platform (OP) and related security concerns [27]), where *"the OP does not expose its internal topology and configuration, Cloudlets' physical locations, internal IP addressing, and real-time knowledge about detailed resource availability"*. In general,

[23] M.10815 https://ec.europa.eu/competition/mergers/cases1/202302/M_10815_8844242_215_3.pdf
See also the Press Release https://ec.europa.eu/commission/presscorner/detail/en/IP_23_721

according to the PRD, *"The OP provides information on the geographical Region(s) where the edge cloud service is available"*, thus in principle this requirement is a good basis for ensuring location privacy in federated environments. However, more in specifics, mobility and service migration may imply the need to deal with privacy also in these scenarios; as a consequence, from a practical perspective, location privacy threats during service migration are still not completely defined/addressed, and still more standardization work is needed on this area.

In summary, in order to present the reader a map of **privacy threats** and possible coverage of existing tools and available mitigation strategies, the table below shows a qualitative assessment of the suitability of these tools to the specific needs of MEC environments, and especially the even more specific needs of the gMEC4AUTO targeted scenarios. Also in this case, the table represents also a way to give a picture of complementarities and potential overlaps, showing where more elaboration of the tools might be needed, and/or more standard work could be needed or simply further discussed (in these cases, such investigation is out of the scope of the present document, but a dialogue among experts on MEC and on cybersecurity fields may be needed). Also here, the same legenda as above is related to the following table:

	Mitigation Strategies										
Privacy Threats	MS#1	MS#2	MS#3	MS#4	MS#5	MS#6	MS#7	MS#8	MS#9	MS#10	MS#11
PT#1	●	◕	◑		◑						
PT#2	●	◕	◑	◕	◑	◕					
PT#3	●	◕	◑	◕	◑	◕					

6.3 Trust Concerns

TC#1 (Establishment of Trust among different Application Servers): relevant strategies in this context are: MS#9, MS10. Validating trust in the Vehicle-MEC continuum is not a simple exercise of assessing the integrity of each actor as a standalone component, but rather attesting the entire service graph chain considering also all the infrastructure elements of the MEC Service Provider. This translates to capturing all the trust relationships between all stakeholders and actors considering also the different trust domains where each MEC AS operates (Section 4.4.1). While the existing trust assessment mechanisms we discussed in MS#9 and MS#10 cover the latter requirement through the provision of attestation enablers offering different Levels of Assurances (LoAs), these neither do capture the entire design space of possible interactions between MEC-internal components (MS#10) nor the new (and more advanced) types of threats that aim to bypass the verification process through obfuscating their existence so that current types of measurement logs cannot detect them (MS#9). More specifically, the need to enforce specific trust relationships for and between the virtualization and physical resources poses the challenge of MEC End-to-End Trust Lifecycle Management which has not received the proper attention till now. Envisioned solutions need to facilitate decisions on trust made on a multitude of parameters including the geographical and regulatory location of the AP; hardware capabilities and provenance of the underlying physical resources; software capabilities and provenance of those processes and items comprising a MEC Service; execution chain of trust; time elapsed since last audit; ownership of hardware; ownership of software; appropriate use of encryption techniques; software hardening; and physical security.

Furthermore, the prominence of exploits including "transient" malware (Section 4.4.1) that try to manipulate cross-layer vulnerabilities will require the enrichment of the software and hardware measurements that can be provided by the underlying RoT as part of its Chain-of-Trust (MS#9). This should not only cover load- and run-time memory and software integrity but will need to consider hardening of all access management policies that govern the interaction and communication between internal processes (of varying locality) and data.

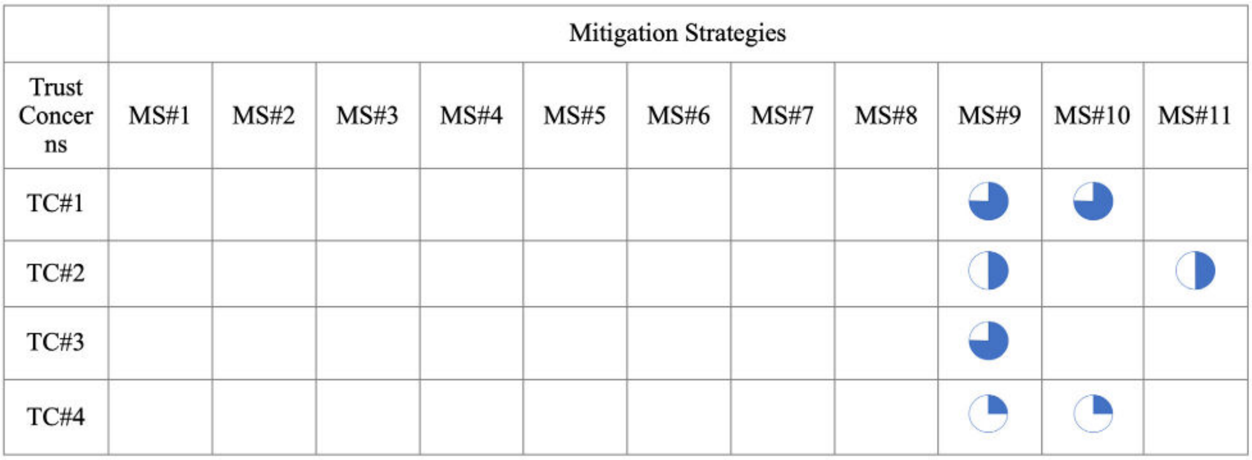
TC#2 – (Secure and Stateful Applications Migration): relevant strategies in this context are: MS#9, MS#11. Even though the maturity of migration technologies that we discussed in MS#11 can offer reliable solutions for flexible migration, the migration of workloads in a decentralized manner and with minimal trust assumptions, which serve verticals with demanding trust and security properties introduce an exceptional challenge. This is because, services which capitalise on hardware-enabled trusted execution environments (MS#9) lose the ability for live migration [15]. This limitation poses significant challenges in the realisation of live migration techniques and possible solutions will need to consider the realization of new key management schemes to be integrated as part of the underlying RoTs and security monitors. Confidentiality and integrity are to be supported through secret sharing crypto primitives and security constructions. More specifically:

1. identity and status of VFs are to be verified and traced via run-time in a trust manner.
2. secret sharing is to be securely delivered to the migrating VF from its affiliated peers via trusted links. A secret will be split within a local control node's TEE and the shares will be delivered to identified VFs via secure communication channel. The secure channel will be built by the TEE key. Since each device will have an installed trusted anchor, it shall be able to use its unique secret key to set up secure channel, like SSL, with other entities. Via the channel, the shares are free from being compromised and manipulated by network attackers during its delivery.
3. Secret Sharing (SS) supports TEE key migration from a broken node to others, to guarantee secure VF migration and replaceability.

TC#3 – (Data Location and Lifecycle): relevant strategies in this context are: MS#9. As we discussed in Section 4.4.3, it is important to be able to enhance data provenance via assertions auditing the integrity and correctness of the data lifecycle. This becomes even more important for safety-critical applications, like in C-ITS. However, these cryptographic assertions are now managed by the virtualized TEE safeguards discussed in MS#9 and are instantiated in each one of the actors in the MEC paradigm. A remaining gap is that attackers can exploit lesser trust domains (with no or minimal integrity guarantees on the host functional assets) to recover data. Furthermore, a MEC application, originating from an MNO with a smaller LoA, might attempt to manipulate the permission layer at the virtualization layer so as to get access to internal (sensitive) data structures of other workloads running.

TC#4 – (Continuous authorization and authentication): relevant strategies in this context are: MS#9, MS10. The mitigation strategies discussed in MS#9 and MS#10 do not capture the dynamic nature of current and future C-ITS ecosystems where the trust state of each actor is constantly changing and, thus, affecting the establishment of new trust relationships (or the existing ones). What is needed is a trust architecture and a dynamic trust assessment methodology enabling vehicles to continuously assess the level of trust they can place on the MEC when consuming its services. Been aligned also with the emerging Zero-Trust paradigm [12], such trust relationships need to capture the safety-critical nature of C-ITS services and the presence of multiple actors (multi-MNO and multi-MEC providers).

In summary, in order to present the reader a map of **trust concerns** and possible coverage of existing tools and available mitigation strategies, the table below shows a qualitative assessment of the suitability of these tools to the specific needs of MEC environments, and especially the even more specific needs of the gMEC4AUTO targeted scenarios. Also in this case, the table represents also a way to give a picture of complementarities and potential overlaps, showing where more elaboration of the tools might be needed, and/or more standard work could be needed or simply further discussed (in these cases, such investigation is out of the scope of the present document, but a dialogue among experts on MEC and on cybersecurity fields may be needed). Also here, the same legenda as above is related to the following table:



	Mitigation Strategies										
Trust Concerns	MS#1	MS#2	MS#3	MS#4	MS#5	MS#6	MS#7	MS#8	MS#9	MS#10	MS#11
TC#1									◕	◕	
TC#2									◑		◑
TC#3									◕		
TC#4									◔	◔	

7. Conclusions

In this technical report, the 5GAA approach to MEC security, privacy and trust, from automotive perspective, is following the work started in MEC4AUTO [28] (and continued in gMEC4AUTO work item), where the reference architecture is targeting MEC systems deployed in Multi-MNO, Multi-OEM and multi-vendor environments. As a consequence, this document targeted a very specific and tailored scenario, thus covering a smaller part of the entire "galaxy" of cybersecurity. It thus analysed the main threats from security, privacy and trust perspectives, and provide an overview of the most relevant mitigation strategies available in the industry, by finally evaluating them in terms of suitability for the 5GAA gMEC4AUTO architecture and targeted use cases. Finally, the report highlighted possible gaps and future work in that perspective.

One of the conclusions we can draw is that standards still need to further work on this area, in order to fully cover the specific needs or preserving user identity, while at the same time being capable to offer ubiquitous and global MEC services in Multi-MNO, Multi-OEM and multi-vendor environments. Also, some more work is needed, in order to fully cover the specific needs or preserving location privacy, while at the same time being capable to offer ubiquitous and global MEC services in Multi-MNO, Multi-OEM and multi-vendor environments. In particular, the need to offer MEC services across a federation could be in contrast with the concept of "separation of concerns" by OPG. This fundamental trade-off should be first resolved by industry associations like GSMA (and also vertical market representatives like 5GAA) to effectively drive the standardization work on proper directions that can address industry needs.

Another conclusion that comes out of our analysis is that trusted computing is a core enabler for measuring and validating the trustworthiness and resilience of a system against a wide set of attack vectors targeting its integrity and the confidence of other C-ITS actors. In this context, both authentication and attestation are prominent security constructions for creating assurances on platform integrity state and the ability to protect data in accordance to various security levels and policies. While there are existing trust assessment mechanisms that cover the requirement through the provision of attestation enablers offering different Levels of Assurances (LoAs), there is need for new solutions that capture both the entire design space of possible interactions between MEC-internal components and the new (and more advanced) types of threats that aim to bypass the verification process. Another significant challenge concerns the realization of live migration techniques. Possible solutions there will need to consider new key management schemes to be integrated as part of the underlying root of trust and security monitors. Overall, what is needed is a trust architecture and a dynamic trust assessment methodology enabling vehicles to continuously assess the level of trust that they can place on the MEC when consuming its services.

In summary, the identified gaps encourage the industry to work more on these topics, where the requirements from automotive domain are very specific and tailored to the need to provide global MEC deployments supporting secure C-V2X services.

5GAA is a multi-industry association to develop, test and promote communications solutions, initiate their standardisation and accelerate their commercial availability and global market penetration to address societal need. For more information such as a complete mission statement and a list of members please see https://5gaa.org

5GAA Automotive Association