Cognitive Heterogeneous Architecture for Industrial IoT

D2.1 Design Specification for CHARIOT Cognitive IoT Architecture (preliminary design)

Document Summary Information

<table>
<thead>
<tr>
<th>Grant Agreement No</th>
<th>780075</th>
<th>Acronym</th>
<th>CHARIOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full Title</td>
<td>Cognitive Heterogeneous Architecture for Industrial IoT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Start Date</td>
<td>01/01/2018</td>
<td>Duration</td>
<td>36 months</td>
</tr>
<tr>
<td>Project URL</td>
<td><a href="http://www.chriotproject.eu">www.chriotproject.eu</a></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deliverable</td>
<td>D2.1 - Design Specification for CHARIOT Cognitive IoT Architecture (preliminary design)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Work Package</td>
<td>WP2: CHARIOT Cognitive IoT Architecture and Platform</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contractual due date</td>
<td>M12 – 31/12/2018</td>
<td>Actual submission date</td>
<td>M12 - 21/12/2018</td>
</tr>
<tr>
<td>Nature</td>
<td>R</td>
<td>Dissemination Level</td>
<td>PU</td>
</tr>
<tr>
<td>Lead Beneficiary</td>
<td>VLTN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Responsible Author</td>
<td>Bill Karakostas</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contributions from</td>
<td>CLMS, EBOS, IBM, ILS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Grant Agreement No 780075.
**Revision history (including peer reviewing & quality control)**

<table>
<thead>
<tr>
<th>Version</th>
<th>Issue Date</th>
<th>% Complete</th>
<th>Changes</th>
<th>Contributor(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>V0.1</td>
<td>06/10/17</td>
<td>10</td>
<td>Initial draft</td>
<td>Bill Karakostas (VLTN)</td>
</tr>
<tr>
<td>V0.2</td>
<td>14/05/18</td>
<td>20</td>
<td>Quality Assurance check</td>
<td>Christos Skoufis (EBOS)</td>
</tr>
<tr>
<td>V0.3</td>
<td>08/12/17</td>
<td>40</td>
<td>List of major components added</td>
<td>Bill Karakostas (VLTN)</td>
</tr>
<tr>
<td>V0.4</td>
<td>07/03/18</td>
<td>60</td>
<td>LL Architectural use cases added</td>
<td>Bill Karakostas (VLTN)</td>
</tr>
<tr>
<td>V0.5</td>
<td>14/05/18</td>
<td>80</td>
<td>Revisions to accommodate new components e.g. security engine, blockchain</td>
<td>Bill Karakostas (VLTN)</td>
</tr>
<tr>
<td>V0.6</td>
<td>14/05/18</td>
<td>80</td>
<td>Quality Assurance check</td>
<td>Christos Skoufis (EBOS)</td>
</tr>
<tr>
<td>V0.7</td>
<td>14/05/18</td>
<td>85</td>
<td>Revisions based on comments and feedback from Consortium partners and QA</td>
<td>Bill Karakostas (VLTN)</td>
</tr>
<tr>
<td>V0.8</td>
<td>14/05/18</td>
<td>90</td>
<td>Revisions to accommodate new components e.g. security engine, blockchain</td>
<td>Bill Karakostas (VLTN)</td>
</tr>
<tr>
<td>V0.9</td>
<td>14/05/18</td>
<td>90</td>
<td>Quality Assurance check</td>
<td>Christos Skoufis (EBOS)</td>
</tr>
<tr>
<td>V1.0</td>
<td>12/11/18</td>
<td>95</td>
<td>Restructuring and formatting to comply with deliverable requirements, addition of 3 Annexes.</td>
<td>Bill Karakostas (VLTN)</td>
</tr>
<tr>
<td>V1.1</td>
<td>14/05/18</td>
<td>100</td>
<td>Quality Assurance and Peer Review checks</td>
<td>Antonis Migiakis (CLMS) Bora Caglayan (IBM) Konstantinos Loupos (ILS) Christos Skoufis (EBOS)</td>
</tr>
<tr>
<td>V1.2</td>
<td>03/12/18</td>
<td>100</td>
<td>Almost final version. Incorporated comments from CLMS, EBOS and IBM</td>
<td>Bill Karakostas (VLTN)</td>
</tr>
<tr>
<td>V2.0</td>
<td>21/12/18</td>
<td>100</td>
<td>Final version for circulation</td>
<td>Bill Karakostas (VLTN) Ioanna Fergadioti (VLTN) Konstantinos Loupos (ILS)</td>
</tr>
</tbody>
</table>

---

1 According to CHARIOT’s Quality Assurance Process:
4 months before Deliverable’s Due Date: 70% should be complete
3 months before Deliverable’s Due Date: 90% should be complete
2 months before Deliverable’s Due Date: close to 100%. At this stage it sent for review by 2 peer reviewers
1 month before Deliverables Due Date Month-1: All required changes by Peer Reviewers have been applied, and goes for review by the Quality Manager
**Disclaimer**

The content of the publication herein is the sole responsibility of the publishers and it does not necessarily represent the views expressed by the European Commission or its services.

While the information contained in the documents is believed to be accurate, the authors(s) or any other participant in the CHARIOT consortium make no warranty of any kind with regard to this material including, but not limited to the implied warranties of merchantability and fitness for a particular purpose.

Neither the CHARIOT Consortium nor any of its members, their officers, employees or agents shall be responsible or liable in negligence or otherwise howsoever in respect of any inaccuracy or omission herein.

Without derogating from the generality of the foregoing neither the CHARIOT Consortium nor any of its members, their officers, employees or agents shall be liable for any direct or indirect or consequential loss or damage caused by or arising from any information advice or inaccuracy or omission herein.

**Copyright message**

© CHARIOT Consortium, 2018-2020. This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both. Reproduction is authorised provided the source is acknowledged.
# Table of Contents

1  Executive Summary .......................................................................................................................... 8

2  Introduction...................................................................................................................................... 9
   2.1  Purpose and structure of this document .................................................................................. 9
   2.2  Mapping CHARIOT Outputs ................................................................................................. 10
   2.3  Deliverable Overview and Report Structure ........................................................................... 11

3  Background and State of the Art ..................................................................................................... 12
   3.1  The current state of IoT ......................................................................................................... 12
   3.2  IoT Safety, Security, and Privacy Challenges ........................................................................ 12
   3.3  Relations between IoT safety, security and privacy/confidentiality ...................................... 13
   3.4  IoT Software Quality ............................................................................................................ 13
       3.4.1  Use of Static Source Code and Binary Analysers for IoT software development ............ 13
       3.4.2  Third-party IoT code .................................................................................................... 14
   3.5  IoT Standards ....................................................................................................................... 14
   3.6  Industrial systems and IoT .................................................................................................... 15
   3.7  IoT Reference Architectures ............................................................................................... 16
       3.7.1  Industrial safety standards ............................................................................................ 16

4  Objectives and scope of the CHARIOT System ............................................................................. 17
   4.1  Introduction ............................................................................................................................ 17
   4.2  CHARIOT Architecture Overview ....................................................................................... 18
       4.2.1  Overview ....................................................................................................................... 18
   4.3  Overview of Architectural Building Blocks ........................................................................... 19
       4.3.1  Description of main system components ...................................................................... 19
   4.4  CHARIOT scope with reference to the OSI model layers ..................................................... 20
   4.5  CHARIOT Data Flow Processing Architecture ..................................................................... 21
       4.5.1  Overview ....................................................................................................................... 21
       4.5.2  Sensor data validation .................................................................................................... 22

5  Structure of IoT data and metadata ............................................................................................... 24
   5.1  Sensor and Device Identity ..................................................................................................... 24
   5.2  The CHARIOT IoT Sensor Data Packet ................................................................................. 24
   5.3  IoT Metadata .......................................................................................................................... 25

6  The CHARIOT Fog network ........................................................................................................... 26
   6.1  Overview .................................................................................................................................. 26
   6.2  Types of Fog nodes ................................................................................................................ 27
   6.3  Trust in the Fog Network ....................................................................................................... 27
   6.4  Performance requirements of the Fog Network ..................................................................... 27
       6.4.1  Fog Network Implementation Options .......................................................................... 27

7  Northbound and Southbound CHARIOT Gateways ....................................................................... 28
   7.1  IoT Gateway Functionality .................................................................................................... 28
   7.2  Industrial IoT Sensors considered in CHARIOT ................................................................. 28
   7.3  ‘Smart’ Sensors versus ‘Smart’ IoT Gateways ...................................................................... 28

8  Blockchain Services .......................................................................................................................... 30
   8.1  Rationale ................................................................................................................................ 30
   8.2  Structure of the CHARIOT Blockchain System .................................................................... 30
       8.2.1  The Role of Blockchain in Firmware Integrity Checking ............................................. 31
       8.2.2  Firmware verification at load time ................................................................................ 32
       8.2.3  Security Engine: Run-time firmware integrity verification .......................................... 32
       8.2.4  CHARIOT Blockchain Implementation Options .......................................................... 33

9  The CHARIOT Privacy and Safety Engines ................................................................................... 34
List of Figures

Table 1: Adherence to CHARIOT’s GA Deliverable & Tasks Descriptions ................................................. 10
Figure 3.1: Typical control architecture ........................................................................................................ 15
Figure 4.2: Architecture Overview ................................................................................................................ 18
Table 2: CHARIOT main architectural components ...................................................................................... 19
Figure 4.3: OSI Layers and CHARIOT ........................................................................................................... 21
Figure 4.4: Stages and flows of IoT data processing ......................................................................................... 22
Figure 4.5: IoT data producers, consumers and queues .................................................................................... 22
Figure 4.6: CHARIOT architectural components as IoT data producers and consumers ............................... 23
Figure 5.1: Structure of the IoT data packet .................................................................................................... 24
Figure 5.2 Describing sensor metadata with SENSORML .............................................................................. 25
Figure 6.1: The CHARIOT Fog network ........................................................................................................ 26
Figure 8.1: the CHARIOT Blockchain network .............................................................................................. 31
Figure 8.2: Workflow for installing and verifying trusted firmware on CHARIOT devices ............................ 33
Figure 9.1: Overview of CHARIOT engines architecture .............................................................................. 35
Figure 11.1: Sequence diagram for the Building Management Use Case ....................................................... 40
Figure 11.2: Sequence diagram for airport security use case. ..................................................................... 41
Figure 11.3: Sequence diagram for Rail Safety Use Case ............................................................................. 41
Figure A1.1 Implementation architecture of the CHARIOT Rail Lab. ............................................................ 45
Figure A2.1: Realisation of a CHARIOT Network using a cluster of Raspberry Pi single board computers. ........ 46
Figure A2.2: Use of the MQTT protocol in the CHARIOT network .............................................................. 47
Figure A2.3: Measuring the performance of the CHARIOT Network ............................................................ 48
Figure A3.1: Gateway / Sensors interconnection ............................................................................................ 50
Figure A3.2: Gateway / Internet communication ........................................................................................... 51
Figure A3.3: Gateway Hardware structure .................................................................................................... 52
List of Tables

Table 1: Adherence to CHARIOT’s GA Deliverable & Tasks Descriptions 10
Table 2: CHARIOT main architectural components 19
Glossary of terms and abbreviations used

<table>
<thead>
<tr>
<th>Abbreviation / Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCS</td>
<td>Blockchain Services</td>
</tr>
<tr>
<td>CSAS</td>
<td>Cloud Sensor Analytics Services</td>
</tr>
<tr>
<td>CFG</td>
<td>CHARIOT Fog Network</td>
</tr>
<tr>
<td>DoA</td>
<td>Description of Action</td>
</tr>
<tr>
<td>EC</td>
<td>European Commission</td>
</tr>
<tr>
<td>GA</td>
<td>Grant Agreement</td>
</tr>
<tr>
<td>GDPR</td>
<td>General Data Protection Regulation</td>
</tr>
<tr>
<td>HMI</td>
<td>Human Machine Interface</td>
</tr>
<tr>
<td>IIOT</td>
<td>Industrial Internet of Things</td>
</tr>
<tr>
<td>IoT</td>
<td>Internet of Things</td>
</tr>
<tr>
<td>ISSE</td>
<td>IoT Safety Supervision Engine</td>
</tr>
<tr>
<td>LL</td>
<td>Living Lab</td>
</tr>
<tr>
<td>MS</td>
<td>Milestone</td>
</tr>
<tr>
<td>NISG</td>
<td>Industrial network (northbound)</td>
</tr>
<tr>
<td>OSI</td>
<td>OSI modelling</td>
</tr>
<tr>
<td>PF</td>
<td>Privacy Engine</td>
</tr>
<tr>
<td>PKI</td>
<td>Public Key Infrastructure</td>
</tr>
<tr>
<td>PLC</td>
<td>programmable logic controllers</td>
</tr>
<tr>
<td>PoC</td>
<td>Proof of Concept</td>
</tr>
<tr>
<td>SAE</td>
<td>Safety Engine</td>
</tr>
<tr>
<td>SAESC</td>
<td>Safety Engine- System context</td>
</tr>
<tr>
<td>SEE</td>
<td>Safety Engine</td>
</tr>
<tr>
<td>SIL</td>
<td>Safety Integrity Level</td>
</tr>
<tr>
<td>SISG</td>
<td>IoT gateway (southbound)</td>
</tr>
<tr>
<td>ST</td>
<td>Sub Task</td>
</tr>
<tr>
<td>T</td>
<td>Task</td>
</tr>
<tr>
<td>WP</td>
<td>Work Package</td>
</tr>
</tbody>
</table>
1 Executive Summary

CHARIOT system’s mission is to prohibit untrusted (possibly malicious, or simply faulty) IoT data crossing over from the IoT network to an industrial network (for example, a rail management system, a building maintenance system, etc.) and create safety, security or privacy risks. To achieve that, the CHARIOT system follows an architectural framework that analyses the IoT system and the data it generates, from a software code quality, and privacy, safety and security risk perspectives. This report is the interim (draft, M12) version of the architecture of the CHARIOT system, the Project’s D2.1 deliverable as an output of WP2 activities and in particular task 2.1. It has undergone several reviews and revisions since its inception in the beginning of the project. The architecture is the result of work carried out in Task 2.1 ('Design specification for CHARIOT cognitive IoT Architecture ') and addresses the Project Objectives as stated in the Description of Action (DoA). The purpose of this document is to provide an architectural technical overview of the CHARIOT project goals, objectives and approach and to steer technical work in Work Packages 1, 2 and 3. It contains statement of the problems tackled by CHARIOT, discussion of the State of the Art in IoT, main architectural components, architectural use cases from the Project’s Living Labs in the areas of building management, airport security and rail safety, reference IoT technologies, and proof of concept implementations. The architecture has been aligned to end-user and industrial use-cases as the architectural use cases for building management, rail safety and airport security, considering that a system architecture should not be the design for a particular application rather than the design artifact and a blueprint for systems that will be asked to fulfil specific missions within a domain (in our case the domain is 'Industrial IoT' and the systems will have to realize safety security and privacy roles). Thus, the design of the architecture is the result of analysing and understanding the industrial IoT domain and its characteristics and the impact they have on privacy, security and safety system properties. This is here reported as the “architectural use cases” in chapter 11.

This report includes an overview of the CHARIOT related background and the related state-of-the-art analysis recognizing the importance and analyzing recent IoT safety, security and privacy challenges as well as their relations. It also analyses the IoT software quality requirements and risk as they relate to mandating the use of source code analysis across their development teams (during development, quality assurance, and security auditing), utilizing binary analysis for 3rd party code analysis and design and development activites with a “security-first” philosophy aligned with the CHARIOT static code (WP1) and binary analyses (WP2-WP3). IoT standards as well as industrial IoT systems, and reference architectures are also described and referenced in relation to the above activities. Later in the report, the system architecture is presented that will enable CHARIOT to act as an active monitor and filter of IoT data that prevents such data creating a safety, security or privacy risk for the industrial system while leverage CHARIOT dependable solution for trusted IoT data it itself architected in a way that operates in a trusted and dependable manner. In this scope the architecture, scope, data flow and processing as well as IoT data and metadata related to sensors are also described. The CHARIOT Fog network is also presented as a new computational paradigm proposed to compliment cloud-based processing and is relevant to the characteristics and needs of IoT and that connects computational and storage resources in a local network that is physically closely located to the industrial system. The architecture includes also the north/south CHARIOT gateways and their functionalities and connectivities as well as the blockchain rationale, role, operational attributes and standing inside the CHARIOT overall architecture and system (firmware verification, security engine etc). The report also includes analysis of the privacy and safety engines as the intelligent cognitive units applying domain specific rules relevant to the industrial system domains as well as IoT analytics under the concept of the CHARIOT Fog Network, as a set of locally available computing, storage and networking resources. Collectively these are known as the CHARIOT Cognitive Platform. Closing, the report presents the three main use cases that demonstrate the main principles of the CHARIOT architecture in the domains of building management, rail safety and airport security, corresponding to the Project’s Living Labs’ use cases.

Keywords: IoT architecture, IoT functional safety, IoT security, IoT privacy, Blockchain, IoT network, wireless sensor, smart sensor, Fog network, IoT gateway
2 Introduction

2.1 Purpose and structure of this document

According to CHARIOT’s Description of Action (DoA): “Task 2.1 will specify a meta-architecture – that accommodates and integrates with all leading IoT technologies, platforms and software stacks and allows IoT devices to augment with cognitive capabilities. The architecture will be specified in terms of several orthogonal concerns.”

This document is the output from task 2.1 ‘Design specification for CHARIOT cognitive IoT Architecture’ according to CHARIOT Project Workplan. It documents the architecture of a technical system that the Project’s DoA coins ‘cognitive IoT Architecture’. The meaning of the term ‘cognitive’ in the context of CHARIOT is explained in the following sections.

This report contains a high-level description of the architecture of the CHARIOT system. This includes key architectural components their key functionalities, role in the overall architecture and inter-dependencies. It also includes the description of interfaces to other systems, external to CHARIOT, including the IoT System and the Industrial System. Additionally, key architectural orthogonal concerns or constraints regarding issues such as for example performance, scalability, data processing architectural styles and implementation options are discussed. This approach allows for different realizations of the architecture for different industrial environments, using different IoT technologies. This report provides high level architectural goals and constraints, as a blueprint for the detailed design and implementation of the various subsystems and components under specific tasks of Work Packages 1, 2 and 3. The report presents proof of concept implementation approaches and reference IoT technologies in Annexes I and III.

In addition, architectural use cases are used to explain how the proposed architecture applies to the Project’s Living Labs Use cases. A revised version of this report will be submitted in Month 25 to accommodate experience and feedback from the Project’s Living Labs (Work Package 4). At this stage of the project, the CHARIOT architectural models presented in this report have guided the development and integration of CHARIOT subsystems in the following tasks:

- Task 1.2 (‘Method for coupling pre-programmed private keys on IoT devices with a Blockchain system’). The Blockchain part of the Security Engine has been developed integrated with other components of the CHARIOT system (Southbound Gateway, Dispatcher, etc.).
- Task 2.2 (‘CHARIOT IoT Cognitive Platform’). In this ongoing task, the CHARIOT Platform has been designed and prototyped according to the architectural guide presented in this report.
- Task 2.3 (‘Interface to IoT Gateways and components’). In this ongoing task, various types of sensors that will be utilized in the Living Labs have been connected to the CHARIOT Reference IoT Gateway (discussed in Annex III of this report).
- Task 2.4 (‘Task 2.4 Security & Blockchain’). In this ongoing task the Blockchain subsystem (already designed and implemented as part of Task 1.2) will be interconnected with other components and services such as the CHARIOT Security Engine (Task 3.3 ‘Firmware Security integrity checking with embedded neural networks’).
- Task 3.2 (‘Privacy Engine based on PKI and Blockchain technologies’). The architecture specification guides work in this ongoing task.
- Task 3.1 (‘Design specification of IPSE’). The architecture specification guides work in this ongoing task.
2.2 Mapping CHARIOT Outputs

The purpose of this section is to map CHARIOT’s Grant Agreement commitments, both within the formal Deliverable and Task description, against the project’s respective outputs and work performed.

Table 1: Adherence to CHARIOT’s GA Deliverable & Tasks Descriptions

<table>
<thead>
<tr>
<th>CHARIOT GA Component Title</th>
<th>CHARIOT GA Component Outline</th>
<th>Respective Document Chapter(s)</th>
<th>Justification</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DELIVERABLE</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D2.1 Design specification for CHARIOT cognitive IoT Architecture’</td>
<td>A report containing high-level architectural models, architectural use cases and high-level functional descriptions of the CHARIOT architecture including the IoT security, IoT interoperability and Integration, and the Cognitive Components Module.</td>
<td>Sections 5 to 11 provide overview and details of the architectural components and subsystems. Section 12 presents the architectural use cases.</td>
<td>The architecture is presented in a logical layered approach. Key components/subsystems are then further explained in dedicated sections.</td>
</tr>
<tr>
<td><strong>TASKS</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST 2.1.1 “IoT security concern description. Security micro-services built specifically for IoT-based solutions”</td>
<td>Security micro-services built specifically for IoT-based solutions</td>
<td>Section 9.2.3 Security Engine: Run-time firmware integrity verification</td>
<td>CHARIOT uses blockchain for security purposes, i.e. authenticating the integrity of the IoT sensors’ firmware against information stored in a blockchain.</td>
</tr>
<tr>
<td>ST 2.1.2 IoT interoperability and integration of contextual data sources</td>
<td>“Leading the industry at partnering with outside data providers (e.g. Weather Company)”</td>
<td>Section 6.3 (‘IoT Metadata’).</td>
<td>CHARIOT architecture enriches sensor data with metadata such as regarding the sensor’s location to implement intelligent (‘cognitive’) sensor data validation.</td>
</tr>
<tr>
<td>ST 2.1.3 IoT cognitive concern specification</td>
<td>Specification of Understanding, Reasoning, Learning capabilities</td>
<td>Section 10, (‘The CHARIOT Privacy and Safety Engines’), Section 11: (‘IoT Analytics’).</td>
<td>CHARIOT intelligently analyses sensor data against patterns learned from historical data. Then, it deploys the learned rules about sensor data on runtime safety and privacy Engines.</td>
</tr>
</tbody>
</table>
2.3 Deliverable Overview and Report Structure

The report is organized as 11 main sections, a references (bibliography) section, and 3 Annexes. Section 1 is an executive summary, and Section 2 (this section), provides an introduction, the relation to the Project Grant agreement tasks and objectives, and a map to the subsequent report sections. Section 3 contains a discussion of the State of the Art in IoT and related areas, while Section 4 provides an overview of the CHARIOT system architecture, the patterns and style of processing used. The following sections discuss the main architectural components in more detail:

- Section 5 discusses the structure of IoT data and metadata/context.
- Section 6 discusses the data processing architecture employed in CHARIOT.
- Section 7 presents the Southbound and Northbound interfaces (‘gateways’) of the CHARIOT system.
- Section 8 discusses the blockchain components and services used for sensor authentication and validation by the Security Engine.
- Section 9 presents the Privacy and Safety Engines that apply business rules to filter incoming IoT data.
- Section 10 discusses external services such as Cloud-based IoT data analytics.
- Section 11 presents architectural use cases that correspond to the Project’s Living Labs in the areas of building management, airport security and rail safety.
- Section 12 contains the summary and conclusions of the work carried out in Project Task 2.1.
- Finally, Annexes I to III contain an application of CHARIOT to rail safety, a proof of concept CHARIOT system implementation, and description of the reference IoT Gateway used in the Project.
3 Background and State of the Art

3.1 The current state of IoT

The Internet of Things (IoT) can be defined as a worldwide network of interconnected devices. IoT technologies and applications are rapidly gaining traction in all consumer and business areas. IoT has the potential to revolutionize the way things are currently done and bring radical innovation in many business processes in ecommerce, healthcare, manufacturing and other domains. Gartner estimates that there will be 21 billion things connected to the Internet by 2020\(^2\). However, due to its nature, IoT also introduces several types of risks, mainly due to:

- the very large numbers of IoT devices that interact concurrently
- the dynamic nature of IoT systems, with the continuous addition of new devices, services and capabilities
- the large amounts of data generated by IoT
- the conflict between the need to provide wide access to IoT data, versus respecting the privacy of the owners or subjects of such data
- the provenance of IoT firmware, software and hardware, and quality standards
- the need for cross network connectivity (WiFi, cellular mobile communications, intranet, Internet) that inadvertently exposes IoT to online vulnerabilities.

It is widely agreed that IoT must become more trustworthy before it can be deployed in safety critical systems. In IoT and in the cybersecurity area more general, there are two parallel trends of security by design versus security monitoring (Román-Castro et al, 2018). An IoT system can be designed to adhere to security (and to safety and privacy) principles. This can make it more trustworthy but not totally secure, as this is an unattainable property of an ICT system. Thus, security by design in IoT needs to be augmented with dynamic security checking measures to further increase the trust in the IoT system. The same principle is applied by CHARIOT to the safety and privacy system attributes.

*IoT therefore generates new opportunities but also creates new challenges with respect to trustworthiness (Bojanova & Voas, 2017)* CHARIOT proposes an integrated set of technologies that make industrial IoT installations inherently more safe, secure and privacy preserving, by addressing both static and dynamic security, safety and privacy enforcement. Because the CHARIOT technologies are acting in an integrated manner to ensure the security, safety and privacy properties of the IoT system related to the industrial installation, they constitute a technical system. The architecture of the CHARIOT system is the subject of this report.

3.2 IoT Safety, Security, and Privacy Challenges

The functional safety of a system ensures that a system does not harm its environment. Safety is ensured by implementing safety mechanisms in software and/or hardware. In the IoT context, on the software side, the Operating System/Middleware/applications running on the sensors must support safety in its design and architecture, so that functional safety requirement can be more easily implemented.

Security properties ensure, that the system is not harmed/attacked by its environment. IoT security standards are required to protect IoT connected systems against cybercrime and national security threats and help to ensure that the system is trustworthy and trusted (UKGOV). As IoT is built on top of the Internet, security issues pertaining to the Internet also apply in IoT (Lee, 2017). As the number of connected IoT devices increases, the number of the end-to-end connection points increases exponentially, and the number of potential security vulnerabilities also increases proportionally. Protecting the IoT is however, a complex and difficult task. The

---

number of attack vectors available to malicious attackers might increase exponentially, due to global connectivity ("access anyone") and accessibility ("access anyhow, anytime") of the IoT (Lee, 2017). IoT data and devices must therefore have security built-in by design, but also be scrutinized for their behavior and data at runtime.

### 3.3 Relations between IoT safety, security and privacy/confidentiality

As security problems can evolve to serious safety problems, security can also be seen as an indispensable safety concept. One aspect of security is to keep data confidential and ensure its integrity. For example, if data confidentiality and integrity are not ensured, a medical device may harm the patient with the wrong dosage. If data or the system is not available because of a Denial of Service (DoS) attack to the medical system, the medical device may not function at all.

With these considerations in mind, we need to develop design methodologies that address safety, privacy and security in the architecture, in an integrated manner.

Some good practices that have been taken into account in the design of the CHARIOT architecture include:

- A white list policy for inter-component-communication. Unrestricted communication is prohibited and needs to be allowed explicitly, defined at system configuration-not at runtime. CHARIOT system components that are ‘internal’ do not have external interfaces and only communicate with other internal components over secured (encrypted) connections. They are not allowed to access sensors directly in order to receive data.
- Instead, CHARIOT’s Southbound Gateway is responsible for connecting to sensors and for filtering out invalid sensor data and for rejecting data from non-authenticated sensors, or from sensors that run unauthorized firmware.
- Use of cryptography before exchanging data between CHARIOT and the outside world.
- Security of IoT data: IoT data entering the CHARIOT system that need to be stored for some time period are encrypted (‘encryption at rest’).

### 3.4 IoT Software Quality

As IoT applications become more sophisticated, and open to network and Internet connectivity, the risks of built-in security vulnerabilities are increasing. Despite this, awareness of the risks associated with insecure code is still low among IoT developers and QA teams, and not seen as a priority (Castro et al., 2018).

Adequate levels of software integrity can only be achieved if teams are able to eliminate both accidental coding errors and intentional design-in vulnerabilities, through efficient analysis techniques suitable for today’s highly complex applications. It is recommended that IoT software development teams can start by:

- Mandating the use of source code analysis across their development teams – during development, quality assurance, and security auditing.
- Utilizing binary analysis for 3rd party code analysis.
- Design and develop with a “security-first” philosophy.

### 3.4.1 Use of Static Source Code and Binary Analysers for IoT software development

Modern static-analysis tools are popular because they have proven to be effective, they are simple to introduce, and they can be used by development, QA, and security audit teams. Furthermore, in contrast to dynamic testing,
the code analyzed is never executed, so there is no additional test case development overhead and static analysis can be applied very early in the development process.

When programmers use static analysis as soon as code is written, bugs and security vulnerabilities can be found and eliminated even before the unit testing or integration testing phases begin. The earlier a defect is found, the cheaper it is to fix. This represents one of the major advantages of automated static code analysis.

Static-analysis tools for source and binary have the ability to detect vulnerabilities before an IoT product is shipped, dramatically reducing security threats and corporate exposures that can cost organizations millions of euros.

More recently, new methods and tools for IoT software development and V&V have been proposed (Schmittner et al., 2015). For example, specialized cross-compilers can help avoiding some safety defects. Such software compilers can add some meta-data (cryptographic signature, and/or “proof-carrying code”) in the binary code, hence permitting that binary executable to be suitably filtered or authenticated by gateways.

3.4.2 Third-party IoT code

Over the recent years, third-party code has become a major factor in the software development industry. It is now used throughout software development in all applications, from highly sensitive government applications, to security-intensive financial systems, to safety-critical applications, to consumer and mobile applications.

According to a report from VDC Research, the majority of software that runs on embedded devices is now developed by external sources, not in-house development teams. Some of this is open-source, but in embedded applications, nearly 30% of code is third-party commercial software – so the source is often unavailable. Such components include graphics toolkits, cryptography libraries, and communications middleware (network, USB, Bluetooth), which make up nearly 70% of the common embedded attack vectors.

To deal with the issue where third party IoT code is used but its source code is not available, binary code analysis (both static and dynamic) can be utilized. This is one of the measures taken by the Security Engine of the CHARIOT system as explained in subsequent sections.

3.5 IoT Standards

IoT is a relatively young Information and Communication Technology (ICT), for which standards only recently started to emerge. Current IoT standards are often overlapping and not widely adopted. Of course, older standards for ICT are applicable to IoT. For example, standards like ISO/IEC 27001 and ISO/IEC 27002 addressing governance, risk and compliance issues related to information security, apply to IoT, as are cybersecurity standards such as ISO/IEC 27031 and ISO/IEC 27035. There are also ISO/IEC standards defining encryption and signature mechanisms that are relevant to IoT. Safety standards such as ISO 10377, are also relevant and applicable to the safe deployment of IoT. Finally, software development standards such as IEEE standards for requirements, design, maintenance of software, are relevant for the design, development, deployment and maintenance of IoT installations.

---

4 http://iot-design-zone.com/article/knowhow/2435/addressing-iot-impact-on-software-engineering
5 https://www.vdcresearch.com/Coverage/IoT-Tech/reports/18-RO.html
3.6 Industrial systems and IoT

Industrial Internet of Things (IIoT)\(^6\) describes systems that connects and integrates industrial control systems with enterprise systems, business processes, and analytics.

We define as *industrial* systems those manufacturing plants and installations in domains like energy, telecommunications, transport and industrial production.

*CHARIOT Project is primarily concerned with IIoT so an understanding of the dependencies between IoT and the Industrial System is essential.*

![Figure 3.1: Typical control architecture](image)

As shown in Figure 3.1, industrial systems perform processes that consume resources and produce, or otherwise manipulate, resources such as energy, manufactured products, transport products and so on. The correct execution of the process is achieved with the use of *controllers* which employ *sensors* to measure parameters of the state of the process, as well as *actuators* that alter some parameters (variables) of the process. A controller is a system in its own right, consisting of components such as Human Machine Interfaces (HMI), desktop PCs, as well as specialised hardware such as PLCs, servo controllers and drives. The focus of CHARIOT is on sensors deployed in the environment of the industrial system that capture and transmit data relevant for the control of the system process. Of particular interest, are sensors that have communication and networking capabilities and that can be accessed remotely, i.e. over the Internet. This essentially constitutes IoT in the context of industrial systems (IIoT).

Many of the existing industrial installations of sensors pre-date IoT which is mostly a phenomenon of the past decade, although it originates in research carried out in the 90s, which culminated in the term Internet of Things to be coined by MIT in 1999. However, originally, industrial systems did not use IoT technologies, gradually IoT started to penetrate industrial system installations in overhauls, upgrades and re-placement of older technologies. IoT in industrial installations results in systems that are easier to connect, remotely manage and interoperate, amongst other benefits. Introducing IoT in industrial systems, however, in addition to benefits also brings risks. The risks are the results of the unintended consequences of introducing IoT in an industrial system, i.e. the risks of making such system less safe, secure or private for its stakeholders. The reasons of such unintended consequences are multiple. IoT through its connectivity opens the industrial system to new attack vectors (routes) that can be exploited by malicious actors. IoT data can become corrupted due to non-malicious

(such as sensor malfunctioning or program errors) or malicious causes, presenting the industrial system with incorrect data that can cause it to function incorrectly and create safety hazards. Industrial system data may become inadvertently exposed on the Internet, creating a privacy risk. Also, IoT designers developing IoT technologies are rarely security and privacy experts meaning that such systems might have not been designed with security, safety or privacy in mind.

3.7 IoT Reference Architectures

Reference models and reference architectures for IoT, provide a description of greater abstraction than what is inherent to actual systems and applications. These are more abstract than system architectures that have been designed for a specific application with particular constraints and choices.

CHARIOT is not concerned with the architecture of IoT per se. It however acknowledges the existence of IoT Architectural standards, defined by standards organisations or in the context of R&D projects. Some of such standards include:

- The IoT-A project [http://www.iot-a.eu/arm](http://www.iot-a.eu/arm) that defines an IoT domain (called a Reference Model), and a common foundation for building interoperable IoT system architectures.
- FI-WARE, a middleware platform, driven by the European Union, for the development and global deployment of applications for Future Internet. One main characteristic of FI-WARE is that its API specification is open and royalty-free.
- IEEE P2413 - Standard for an Architectural Framework for the Internet of Things (IoT)[8]. This standard defines an architectural framework for the Internet of Things (IoT), including descriptions of various IoT domains, definitions of IoT domain abstractions, and identification of the commonalities between different IoT domains such as healthcare, automotive, and others.

CHARIOT is IoT standard agnostic: It can handle any type and architectural style of IoT device and network, provided suitable Southbound Gateway(s) are configured to connect the IoT system to CHARIOT.

3.7.1 Industrial safety standards

Quality assurance methods are required to ensure that IoT does not deteriorate the safety of an industrial system beyond a specific level of risk. Such quality assurance methods are codified as standards by international organisations like ISO, IEEE, IEC and others.

Industrial safety standards require the certification of software, which means that application code must be thoroughly tested and documented. These standards define several levels of criticality, which are measured by the severity of the hazards caused by application failure. In case of EN-50128 and IEC-61508 the criticality is ranked from Safety Integrity Level (SIL), with 0 being the most basic up to SIL-4 being the most stringent level. The avionic safety standard DO-178B/C defines Design assurance levels rank from E (basic) to A (strict). A good design approach is to separate non-interacting applications with different SIL levels, allowing mixed criticality levels in the system. This approach is market proven and has undergone several certifications according to industrial standards like IEC61508, EN50128, DO 178B and ISO26262[9].

---

7 A more extensive discussion of IoT standards as well as of safety and security standards is provided in deliverable D1.1 ‘Classification and use guidelines of relevant standards and platforms’.
9 [https://webstore.iec.ch/preview/info_iec61508-1%7Bed2.0%7Db.pdf](https://webstore.iec.ch/preview/info_iec61508-1%7Bed2.0%7Db.pdf)
4 Objectives and scope of the CHARIOT System

4.1 Introduction

CHARIOT system’s scope can be visualized as a layered pyramid shape that builds stacked capabilities to achieve its mission which is to make the industrial system resilient to safety, security and privacy risks related to the IoT.

At the bottom layer of Figure 4.1 lay the raw data collected from IoT sensors. Such data represent the state of the industrial system and of its environment. These are the data on which the industrial control system will base its decisions. It is paramount therefore that sensor data obey several qualities regarding authenticity, correctness, accuracy and timeliness, i.e. that the data are trusted in decision making.

The next layer of the Figure 4.1 pyramid is concerned with the trustworthiness of IoT data. Both static and dynamic techniques used in this layer ensure the authenticity of the sensor data, and also their correctness against some verifiability measures.

The following layer of Figure 4.1 interprets the sensor data within a system context, i.e. they analyse the significance of sensor data in terms of system properties such as safety, security and privacy. This is the reason that CHARIOT calls its system architecture ‘cognitive’, i.e. because it requires the use of domain knowledge and reasoning capabilities. This requires the understanding of issues such as where the data were collected, the entities they refer to, the purpose the sensor data will be used to and the consumers of the data.

The analysis of the sensor data is carried out by what CHARIOT calls the safety and privacy engines. These can be seen as continuously running processes that process the streams of data arriving from the sensors, reject untrusted data and calculate the degree of risk that such data represent to the Industrial System. Data that represent risks that are below a threshold are assumed to be safe to be passed on to be used in the normal operations of the industrial system, while data that are deemed to represent a safety, security or privacy risk are redirected to specific sinks so that appropriate action can be taken if necessary.

Thus, CHARIOT acts as an active monitor and filter of IoT data that prevents such data creating a safety, security or privacy risk for the industrial system. It achieves that through passive and active layers of processing and
filtering the IoT data. It acts in other words, as a safety, security and privacy protection ‘net’ around the industrial system. However, for CHARIOT to be a dependable solution for trusted IoT data it itself must be architected in a way that operates in a trusted and dependable manner.

Thus, the CHARIOT architecture utilizes design principles which assure the trustworthiness and dependability of its functionality. This means that CHARIOT system is resilient to cyberattacks on itself and also resilient to failures, while it is also dependable in the way it handles and processes sensor data. How the dependability properties of CHARIOT are achieved is the subject of the following sections.

4.2 CHARIOT Architecture Overview

4.2.1 Overview

CHARIOT system architecture is not intended as an architecture for building or integrating IoT systems, in the way that for example, FI-WARE (http://fiware.org) is. It is an architecture for bridging or interconnecting IoT systems and industrial systems, in a safe secure and privacy preserving manner. It therefore aims to make the IoT safer, more secure and more privacy preserving, in other words more trusted, in an industrial context. The closest parallel of CHARIOT is the cybersecurity intrusion avoidance and detection systems that operate on the perimeter of a cyber-physical system. However, CHARIOT goes beyond cybersecurity as it also addresses privacy and functional safety of systems.

Figure 4.2 shows a high-level view of CHARIOT system architecture and of its contexts. The main subsystems of CHARIOT are identified in the diagram of Figure 4.2. CHARIOT utilizes three types of networks to connect its components, systems and services.

The first type of network is the IoT Sensor network. Although it is conceptually a single logical network, in practice several types of networks, both wired and wireless may be used to connect the sensors and transmit their data. Sensor data are collected by an IoT Gateway. The Gateway initially authenticates the sensors and rejects data from sensors that cannot be authenticated. The IoT Gateway (or another component called the Dispatcher), in addition to collect the sensor data it may carry out different functions such as ‘cleaning’ (e.g. correcting) the data, re-formatting it, tagging it aggregating it, timestamping it, and then transmitting it to the CHARIOT system.
While the diagram of Figure 4.2 shows a single IoT Gateway in practice several such gateways can be used. Once data are transmitted to the CHARIOT system, they are directed to the processing subsystems that analyse the data from a safety security and privacy perspective and either forward the data to the industrial system for consumption or alert the relevant subsystems of the industrial system for abnormalities, exceptions and inconsistencies.

### 4.3 Overview of Architectural Building Blocks

Below we list the main building blocks of our architectural approach.

In general, all IoT architectures have limits to their applicability as well as different degrees of suitability for different domains/problems. So, for example, an IoT architecture for personal healthcare may not meet the requirements for IoT in the automotive domain. The CHARIOT architecture was conceived for industrial systems that have been interfaced to or extended with IoT sensors, or where designed from the beginning with IoT as the core element of their functionality. Such systems depend on the reliable operation of their IoT element. In particular, if such systems have a safety critical role they can act in an unsafe manner if the IoT becomes unreliable (either maliciously or unintentionally due to environment effects or internal flaws and deficiencies).

#### 4.3.1 Description of main system components

<table>
<thead>
<tr>
<th>Component Acronym</th>
<th>Component Name</th>
<th>Mission</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISSE</td>
<td>IoT Safety Supervision Engine</td>
<td>To ensure that no sensor data will be sent to the safety control system and result in a potentially unsafe operation. ISSE acts as a dispatcher of IoT validated data to the CHARIOT Northbound gateway</td>
</tr>
<tr>
<td>SAE</td>
<td>Safety Engine</td>
<td>A Computational Engine running on the CHARIOT Fog Network that checks the safety properties of incoming IoT data ensuring that such data cannot cause the Industrial Control system to perform unsafe operations.</td>
</tr>
<tr>
<td>SAESC</td>
<td>Safety Engine- System context</td>
<td>The Safety Engine has access to a safety model of the controlled system, e.g. the safety control functions, topology, etc.</td>
</tr>
<tr>
<td>SEE</td>
<td>Security Engine</td>
<td>At runtime evaluates the meta-data contained in the sensor firmware (“proof-carrying techniques” or “cryptographic signature”) and compares with the fingerprint stored in the blockchain.</td>
</tr>
<tr>
<td>PF</td>
<td>Privacy Engine</td>
<td>Its mission is to prohibit the transmission of privacy sensitive data from the gateway. Prior to releasing sensor data to the other engines, some privacy screening must be applied (this makes sense for some types of data in some problem domains). The privacy filter/engine utilises some privacy rules directly encoded in the module or obtainable from an external source.</td>
</tr>
<tr>
<td>Components</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>HMI</td>
<td>Human machine interface Interfaces to the information systems used by the CHARIOT human operators to monitor and control the status of the CHARIOT system.</td>
<td></td>
</tr>
<tr>
<td>SISG</td>
<td>IoT gateway Transmits sensor data to the systems of the CHARIOT Fog Network; manages sensors connected to it. Performs sensor configuration updates.</td>
<td></td>
</tr>
<tr>
<td>NISG</td>
<td>Industrial Network Transmits sensor data validated by the CHARIOT system to the Industrial System.</td>
<td></td>
</tr>
<tr>
<td>CFG</td>
<td>CHARIOT Fog Network The Network providing computational resources used by the CHARIOT Engines and Services</td>
<td></td>
</tr>
<tr>
<td>BCS</td>
<td>Blockchain Services Blockchain services provided by the CHARIOT Fog Network, utilised for sensor firmware verification purposes</td>
<td></td>
</tr>
<tr>
<td>CSAS</td>
<td>Cloud Sensor Analytics Services External Cloud services providing cognitive capabilities to the CHARIOT Engines</td>
<td></td>
</tr>
</tbody>
</table>

### 4.4 CHARIOT scope with reference to the OSI model layers

CHARIOT straddles several technological domains for computing and networking. It is useful to understand its scope and position with respect to other established architectural models. OSI is a layered model for networking as shown in the diagram of Figure 4.3.
According to the OSI model layers, CHARIOT subsystems can be considered as operating at the following layers:

- **Physical layer**: It corresponds in our case to the wireless and wired networks of sensors. In this layer, different communication protocols such as Zigbee, Bluetooth, 6LoWPAN are used by the IoT Network.
- **Data Link**: Ethernet network(s) to implement the CHARIOT Fog Network.
- **Network**: IP is the protocol for data transmitted within the CHARIOT Network.
- **TCP/IP**: is the transport protocol utilized for inter-component communications in the CHARIOT network. Variants of TCP/IP may be considered when performance (e.g. real time) needs to be accommodated, or for interconnection with the Industrial Network. Popular industrial Ethernet protocols like EtherNet/IP can be used here to allow consistent data and communication to the Industrial network via the Northbound Gateway. There, CHARIOT IoT data can be transmitted to the HMI and desktop PCs (industrial information systems), the PLCs, HMIs, servo controllers and drives (control level) and to the device level distributed I/O and field components (automation level).
- **Session/presentation/application**: On top of TCP/IP layer, higher level protocols may be utilized for communicating with external HMI systems such as control/monitoring workstations. SNMP protocol may be used for CHARIOT network device management. MQTT protocol may be utilized for communicating to external services such as Cloud analytics.

### 4.5 CHARIOT Data Flow Processing Architecture

#### 4.5.1 Overview

In a data flow architecture, the whole software system is seen as a series of transformations on input data, where data and operations are independent of each other. In the data flow approach, the data enters into the system and then flows through the modules one at a time until they are assigned to some final destination (output or a
data store). The connections between the components/modules of the system can be implemented as I/O stream, I/O buffers, piped, or other types of connections. The mission of CHARIOT is to create a trusted IoT environment: one where the data coming from the IoT system can be trusted, i.e: they cannot cause safety, security and privacy risks.

4.5.2 Sensor data validation

Data validation is an essential step to improve data reliability and to make IoT data trusted.

For each sensor measurement, data are usually represented by one-dimensional time series. These values, known as raw data, need to be validated before further use to assure the reliability of the results obtained when using them. Communication problems system, lack of reliability of sensors, deliberate acts of misuse or other inherent errors often arise, generating missing or false data during certain periods of time. Missing or false data must be detected and either flagged as such to the receiver or replaced by estimated data.

Such conditions are handled by CHARIOT through appropriate procedures over several stages, that detect missing or erroneous IoT data and stop them from entering the Industrial System and cause safety, privacy or security risks. Thus, CHARIOT implements several data validation approaches towards increasing the levels of trust to the IoT system and data.
As shown in Figure 4.5, sensor data are packaged with metadata providing additional contextual information and are stored by the data producer in a queue from which they are picked by various data consumer processes. The decoupling between producers and consumers of IoT data packets enables performance and scalability required by the need to transmit and process large volumes of IoT data at a fast rate.

Figure 4.6 illustrates how the CHARIOT Engines pickup sensor packets from the queue and validate them against safety, security and privacy rules embedded in the Engines. This allows sensor data packets that violate some security, safety or privacy rule to be rejected.
5 Structure of IoT data and metadata

5.1 Sensor and Device Identity

IoT sensors and devices are assumed to have a unique identity within the network they exist. Such identity can be based on the network card unique number (MAC address) combined with other identifying information encoded in the node such as a serial number or a CPU id. Unique identity is not a secure property for authenticating the node- it is used for node management/house-keeping by the relevant management services at the Fog or IoT network. It is important that a consistent identification scheme is used across the IoT and Fog networks and the Cloud, otherwise the engines and services operating at each network will not be able to synchronize the data that they exchange and share. One mechanism to achieve that is for the IoT Gateway to advertise the identifiers of connected sensors over a secure channel accessible by the Fog Network and the Cloud Analytics.

A parallel node identification scheme used for node authentication is the digital certificate. For example, X.509 certificate can be used for authentication of network devices.

Additionally, every node in the CHARIOT Fog network is assigned a digital certificate by the administrator of the CHARIOT system.

5.2 The CHARIOT IoT Sensor Data Packet

![Figure 5.1: Structure of the IoT data packet](image)

Header (with internal id timestamp etc for housekeeping duties)

Meta + context data

Raw data (payload).

Meta + raw can be combined if a schema approach is used e.g. as self describing XML, Json,...

As explained in the previous section an IoT Gateway transmits sensor data to the CHARIOT Network that are picked by data producers, converted to packets and stored in a queue. Although it is possible to transmit individual values captured by individual sensors in discrete ‘packets’, the overheads would be in most cases unacceptable. The Dispatcher therefore packages aggregated and enriched sensor data in a single unit we call the IoT Sensor Data Packet.

As shown in Figure 5.1, the IoT Sensor Data Packet conceptually consists of:

A header: containing information used for validation of the packet

A payload: containing information about the contents of the data packet. This includes information about the sensor/sensors that produced the data, timestamp of when the data were produced and processed by the Gateway, aggregation information (i.e. the identifiers set of sensors from which data were collected, the ranges of values they represent, the type of aggregation e.g. average value, median value, timeseries of values etc.).
The general goal is to keep the size of the IoT Sensor data Packet small to avoid performance problems in the CHARIOT Network. On the same token the payload of the data packet must be easily accessible to inspection by CHARIOT Engines and therefore contain self-defining data as much as possible.

To achieve compactness a binary format for representation of structured data such as binary XML or binary JSON (BSON) can be utilized where appropriate.

5.3 IoT Metadata

IoT Metadata therefore are essential for the intelligent processing of IoT data by the CHARIOT Engines. A standard such as SensorML\(^\text{10}\) can be used to tag sensor data with metadata as illustrated by the example of Figure 5.2 Describing sensor metadata with SENSORML

---

6 The CHARIOT Fog network

6.1 Overview

Fog is a new computational paradigm that has been proposed to compliment cloud-based processing and is relevant to the characteristics and needs of IoT (Dastjerdi & Buyya, 2016). Briefly, the Fog network connects computational and storage resources in a local network that is physically closely located to the industrial system. This reduces the latency in data transmission, compared to transmitting to the Cloud and also ensures that the resources required to process the data are dedicated always available, compared to Cloud where resources are shared between multiple clients. However, a Fog network has typically fewer resources compared to a Cloud system as there may be technical (sufficient network bandwidth for example), physical (insufficient space) and financial constraints limiting the size of resources that can be made available to the Fog.

In CHARIOT the Fog network is used primarily for the following purposes:

- To store sensor data locally, for local analytics.
- To provide computational resources for executing blockchain services
- To provide computational resources for executing the safety security and privacy engines.

In physical terms, the CHARIOT Fog network is a wired network that runs either the standard TCP/IP protocols or an industrial version of it such as EtherNet/IP.

The Fog Network provides hardware/software resources that are either physical or virtual machines. For the virtual machines, popular virtualization technologies such as VMs or Docker containers can be used.

The Fog Network connects to other services and networks (Figure 6.1):

- The Industrial Network via the industrial (‘Northbound’) gateway. This communication consists mainly of the sending of control signals (reports, alarms, notifications) to the Industrial Network.
- The Cloud services. The analytics services of the Fog Network transmit analytics queries to the Cloud analytics services and receive back results.
- The IoT sensor network via the IoT (‘Southbound’) Gateway.

Figure 6.1: The CHARIOT Fog network
6.2 Types of Fog nodes
A node is an autonomous unit of computation. It runs on virtual or physical resources of the Fog Network and provides services. The following types of nodes exist in the CHARIOT Fog Network:

- Computational Nodes on which run the CHARIOT Engines.
- Fog Network Administration Nodes, monitoring for example the operation of other services such as the blockchain and the CHARIOT engines.
- Nodes providing blockchain related services.
- Network devices such as gateways routers, switches.

6.3 Trust in the Fog Network
All participating nodes in the CHARIOT Fog network are assumed to be trusted. The network is secure and the only entries to the network from other networks are through the ‘northbound‘ (industrial network gateway) and the ‘southbound’ IoT Gateway.

Sensor Data (‘packets’) coming into the network through the IoT Gateway are scrutinized by a chain of Engines as explained in Section 4.5.

The firmware checking authentication method, ensures the trustworthiness of the providers of sensor data. Thus, a node will consider broadcast packets or other types of messages with sensor data that are signed with the public key of nodes that are in its whitelist and will ignore other packets.

6.4 Performance requirements of the Fog Network
Performance is essential requirement for the Fog network and comes under the following considerations:

- Network performance (latency, bandwidth)
- Processing of sensor data packets and of other exchanged messages
- Local Fog storage capacity and read/write performance
- Performance of the Cloud service (latency, throughput, SLA/availability).

6.4.1 Fog Network Implementation Options
Because of performance and other advantages, Ethernet has emerged as the dominant standard for the physical layer of many industrial protocols such as EtherNet/IP, Ethernet TCP/IP, Modbus TCP/IP and Profinet. Using Ethernet, it will be easier to interconnect the Fog Network with the Industrial Network and connect to devices attached to it, such as PLCs, HMIs, field I/O etc. This will minimise problems such as impedance due to the heterogeneity of the two networks.
7 Northbound and Southbound CHARIOT Gateways

7.1 IoT Gateway Functionality

A gateway is a computational node present in both the IoT and the Fog/Cloud networks, acting as a bridge between the IoT network and the CHARIOT Fog Network. The IoT Gateway also acts as an IoT device manager, i.e. it controls which sensor is permitted to connect to the IoT Network and transmit data to the CHARIOT system. The IoT Gateway maintains a 'whitelist' of sensors that are allowed to connect to the CHARIOT IoT network. This whitelist is stored in a secure storage area on the IoT Gateway such as a secure flash storage area controlled by a microprocessor. The members of the ‘whitelist’ are public keys of the sensors connected to the Gateway or where key based authentication is not possible, another uniquely identifying (but not cryptographically secure) identifier. Where a secure key encryption is possible, the IoT gateway employs a secure communication protocol that authenticates the sensors using symmetric or asymmetric key cryptographic techniques. This is used to filters valid sensor data by rejecting data that are transmitted by sensors that are not authenticated.

The Gateway pre-processes valid sensor data by performing aggregation and other simple statistical operations and packages them together with meta-data such as location and time, before it broadcasts the sensor data packet to the Fog network and also uploads it to the Cloud via the appropriate Cloud service interface.

Additionally, the IoT Gateway is a security critical node because it allows network traffic from the IoT network to cross over to the Fog network, and ultimately, to the Industrial Network. It must be ensured that the IoT Gateway only runs authorized firmware. The integrity verification of the IoT Gateway’s firmware is described in the following sections.

7.2 Industrial IoT Sensors considered in CHARIOT

Many types of industrial sensors will be tested in CHARIOT’s Living Labs. The list below is non-exhaustive:

- Analogue sensors (Temperature, Humidity, Air pressure). These sensors do not include any processor but only the transducer & signal conditioning circuits. They are wired connected and emits raw analogue signals.
- Digital sensors (Wired): They typically use the MODBUS or RS-232 protocols, and can communicate with Gateway at baud rate range typically between 2400 and 115200 bps. Typical MODBUS sensors connected at Gateway are Power Metering, UPS / Battery management, Power Generator management, PLCs, Smart Locks and others.
- Wireless Sensor (Bluetooth Low Energy v. 4.2). Battery operated sensor. Data rate is up to 1Mbit.

7.3 ‘Smart’ Sensors versus ‘Smart’ IoT Gateways

The boundaries between the functionalities performed by sensors and gateways determine the conditions under which sensors are deemed to be ‘smart’. A sensor for example can perform some of the functionalities of the IoT Gateway itself, by for example correcting, formatting and aggregating some of the data that it measures. Usually, such sensors are complete computing units (System on a chip- Soc) that include a microprocessor. These sensors can also implement natively (in hardware or software running on the embedded microprocessor) cryptographic functions. This makes easier to implement for example secure keys and certificates for sensor authentication and for secure transmission of data between the sensor and the Gateway.

In cases where the sensor lacks the above features (‘dumb’ sensor), the IoT Gateway is expected to implement all required functionalities. As smart sensors have also capabilities for connecting directly to the Internet, the boundaries between sensors and IoT Gateways are blurred.
CHARIOT’s architecture addresses industrial systems where security is paramount and proposes that sensors interconnect to other networks only via gateways for the purpose of system management and governance. The CHARIOT architecture is designed to accommodate both ‘smart’ and ‘dumb’ sensors. A reference IoT gateway used by CHARIOT is described in Annex III.
8 Blockchain Services

8.1 Rationale

Blockchain services are services that create and maintain a distributed ledger containing trusted immutable data about the configuration and state of the CHARIOT system. This includes for example the verified identities and firmware of key nodes in the CHARIOT network such as the IoT gateways and sensors. The purpose of the Blockchain Layer in CHARIOT is to ensure trust in the participating nodes in the following sense:

Only authorised by the CHARIOT administrator IoT devices can be installed. No untrusted IoT device is permitted to transmit to CHARIOT, ensuring that no unauthorised devices can impersonate authorised ones and inject illegal data.

CHARIOT uses the technology of a blockchain for the following purposes:

- To register the valid/authorised IoT devices/sensors on the network at any time,
- To guarantee that the nodes run the correct operating system/firmware, i.e. that the code they execute has not been modified intentionally or not by non-authorised actors.

Moreover, the CHARIOT blockchain is designed with built-in redundancy so it is both fail-proof and tamper-proof.

8.2 Structure of the CHARIOT Blockchain System

A blockchain system in CHARIOT primarily comprises a set of peer nodes running on the CHARIOT Fog network. Peers are a fundamental element of the network because they host distributed ledgers with authenticated and immutable information about the state of the CHARIOT system.

The Blockchain network is a network of trusted nodes that runs within the CHARIOT Fog network. Each blockchain peer node maintains a copy of the ledger. Peers synchronise their copies of the ledger with each other by using the relevant consensus protocol supported by the particular blockchain technology used. Each peer node provides an interface that allows other entities in the CHARIOT Fog network to ask queries used for authentication of other nodes/devices and of the data they supply.

The CHARIOT Blockchain must be implemented using a permissioned not a public blockchain.

There are several nodes comprising the blockchain network for the purpose of performance and resilience. This means that as long as the majority of nodes in the blockchain are uncompromised, the blockchain will provide trusted data to the clients.
The ledger containing the state of the data is distributed for the same purpose, i.e. even when a number of blockchain nodes become unavailable, or become untrusted, as long as the majority of the nodes remains intact, the integrity of the blockchain (the CHARIOT system state) is guaranteed. Standard mechanisms of blockchain based on digital keys, transactions and blocks can be used to ensure the integrity of the blockchain.

8.2.1 The Role of Blockchain in Firmware Integrity Checking

The main purpose of the blockchain is to act as a single point of reference for authoritative information about the state of the CHARIOT system. This means that all participating nodes (i.e. network devices such as the gateways as well as the sensors) are trusted not to have been tampered with.

To understand the significance of this in an IoT context, it must be appreciated that IoT networks and their configurations do not remain static for too long. The reasons for that are many folds:

- IoT devices are added and/or removed from the network.
- The firmware code that runs on the devices may be changed i.e. due to upgrade to an improved version.
IoT device insertion and removal is centrally managed at different levels:

- At the IoT gateway level, sensors may become attached or detached.
- At the CHARIOT Fog Level devices may be added or removed, or the code for the various services updated.

One typical malware attack is to replace the firmware of a network device with a malicious one with the purpose of diverting it to different destinations (data theft), injecting unauthorised, i.e. false data, or use the device to launch attacks to other network devices (e.g. DDOS attacks).

CHARIOT has proposed the firmware integrity checking service that ensures the trustworthiness of the firmware running on CHARIOT devices. A high-level description of the software integrity checking is as follows:

- IoT devices submit sensor data to the CHARIOT Fog network.
- The device providing such data must be authenticated against the information held in the blockchain ledger. This is achieved through a request-response challenge where the data provider is challenged with a request to compute a response that utilises knowledge about its firmware. Valid responses are stored in the blockchain ledger and are only known to the blockchain peers.

In the following section we describe how firmware of CHARIOT network devices is verified at load time and at runtime.

8.2.2 Firmware verification at load time

We assume that all CHARIOT Network devices are located at installation time with trusted firmware that is loaded at manufacturing or installation time using a secure cryptographic chip such as TPM, that is present in most computing platforms, ‘CHARIOT’ compliant firmware is then securely loaded on the device. This customised firmware calculates responses to authentication challenges that is used by the verifier (Blockchain node) to detect changes to previous configurations.

8.2.3 Security Engine: Run-time firmware integrity verification

The purpose of runtime firmware integrity verification is to ensure that CHARIOT connected IoT devices are running untampered firmware. The verification is performed by the CHARIOT Security Engine, a trusted node of the CHARIOT Fog Network. Techniques such as remote firmware attestation are used. Remote firmware attestation is typically based on a challenge-response protocol. This approach requires the adaptation of the device firmware, i.e. the installation of ‘CHARIOT compatible firmware’ on the device.

Upon receiving the challenge, the device must compute a response that is based on a combination of the state of its firmware and some random factor that ensures that challenges are constantly varied and responses cannot be eavesdropped and replayed by a malicious third party.
Figure 8.2: Workflow for installing and verifying trusted firmware on CHARIOT devices

Figure 8.2 illustrates the workflow for developing and installing CHARIOT compliant firmware on CHARIOT devices. At runtime this is utilized for authentication of the CHARIOT devices by a Runtime Monitoring Tool using a challenge-response protocol. IoT devices that do not pass the request-response challenge are flagged by the Security Engine so that remedial action can be taken, i.e.:

- The device is blocked.
- The device is rebooted.
- The device’s firmware is remotely updated.
- The device is physically removed from the network and sent for forensic analysis.

8.2.4 CHARIOT Blockchain Implementation Options

There are several blockchain technologies available today of both commercial and research prototype status, such as HyperLedger\(^\text{11}\), Etherium\(^\text{12}\) etc. The choice of a suitable blockchain engine will be determined by factors such as the privacy aspects of the blockchain. The CHARIOT blockchain must be permissioned, i.e. operated by a controlled group of trusted nodes.

While this report referred to a single blockchain ledger in practice multiple ledgers could be operated depending on how the system state is partitioned. Currently, the reference Blockchain technology used by CHARIOT is Hyperledger.

The system overheads introduced by the blockchain must be considered in the provision of resources to the Fog network. The blockchain operation should not unreasonably tax the computational resources of the CHARIOT Fog network. Basic blockchain operations such as transaction validation, assembly into blocks and ledger, peer consensus, maintenance should be done as efficiently as possible. The number of participating peers should be dictated by the size of the IoT Network and the number of transactions/devices that need to be validated while fault tolerance must be maintained.

\(^{11}\) [https://www.hyperledger.org/](https://www.hyperledger.org/)

\(^{12}\) [https://www.etherium.org](https://www.etherium.org)
9 The CHARIOT Privacy and Safety Engines\textsuperscript{13}

9.1 Introduction

The CHARIOT Engines are the intelligence (‘cognitive’) processing units shown in Figure 3.1 in the Introduction section. The term ‘intelligence’ implies that these engines do not operate based on low level fixed processing behaviour, but instead apply domain specific rules (e.g. relevant to the industrial system domain) and a continuously updated understanding of the IoT system’s behaviour, mined from IoT data by the Cloud IoT analytics system and encoded as rules.

To understand the concept of the CHARIOT Engines, the analogy from the networking domain between shallow and deep packet inspection performed by devices such as network routers will be used.

Shallow inspection, i.e. inspection of the packet’s header but not the content, in the CHARIOT case is carried out by the IoT Gateway and the Security as described in previous sections. Shallow inspection functions will discard IoT data that do not originate in a trusted source (e.g. a sensor).

Deep inspection on the other hand, will analyse the contents of the IoT data packet and by applying domain specific rules will compute a risk level that the contents of the packet represent. Risk level is calculated in terms of the unwanted consequences from delivering of the packet to the industrial network. Risks can be further classified as follows:

- Safety risks: If the IoT data packet reaches the Industrial Network it may cause the performance of an undesirable function which can result in damage to assets or humans.
- Security risks: If the IoT data packet transmitted by an insecure IoT device reaches the Industrial Network it can cause a violation of security rules, i.e., the compromise of IT assets.
- Privacy Risks: If the IoT data packet reaches the Industrial Network it may cause inadvertently the disclosure of private (personal) data to unauthorized actors.

Examples of such situations include:

- Safety violation: The IoT data contain values for a system variable that is outside the normal/expected range. Although the source of such data is trusted the validity of the value cannot be verified. If the industrial system controller acts upon this value it can lead to catastrophic failure/accident.
- Security violation: An insecure IoT device transmits sensitive data in an unencrypted format. If this information reaches unauthorized actors in the Industrial system, the security of the whole IoT system may be compromised.
- Privacy violation: The IoT data packet contains images of people who have not consented to their publication. If such images are accessed by unauthorized actors in the Industrial Network, the privacy rights of those people will be violated.

The workflow for IoT packet inspection is illustrated in Section 9.1

\textsuperscript{13} The detailed specification and design of the CHARIOT Engines is the responsibility of Work Package 3.
9.2 Safety Engine

The Safety Engine inspects IoT data packets to calculate their risk level from a safety perspective. As explained in Section 3.6, the Controller of the Industrial System uses the values for control variables provided by the sensor network in order to perform control actions, through actuators. Control actions with a safety objective are performed by the Controller or in some cases by a dedicated Safety Controller. The Safety Engine needs to be aware of the context of IoT data, for example the sensors’ locations, in order to know which of such data will be utilised in safety related functions. Moreover, it will need to know the parameters of the safety control system such as tolerances, ranges and thresholds of control variable values. This information is used to encode safety rules that can evaluate the risk level of safety related sensor data that lie outside the set safety ranges. The purpose of the Safety Engine therefore is to prevent sensor data that for some unknown reason (malfunction or malicious act) are anomalous, to cause a potential safety hazard. The Safety Engine represents an additional layer of safety to the built-in functionality of industrial safety control systems.

9.3 Privacy Engine

The Privacy Engine implements another type of domain specific IoT related intelligence. It inspects sensor data packets and associates their payload with known domain entities for which privacy restrictions regarding the use of their data apply. The rules encoded in the Privacy Engine determine which data packets do not represent a privacy risk and are therefore allowed to enter the Industrial Network and also to be stored in the Cloud Analytics Service.
As with the other CHARIOT Engines, the privacy engine requires domain knowledge, i.e. the ability to associate data and metadata encoded in the IoT sensor data packet with domain entities.

The Privacy Engine therefore flags proactively privacy risks before violations actually occur.

### 9.4 Privacy, Security and Safety Supervision Engine (IPSE)

As per diagram of Figure 9.1, IPSE acts as the proxy/dispatcher between the Security, Safety and Privacy Engines and the Industrial Gateway. IPSE permits only sensor data packets that have been cleared from safety, security and privacy risk perspective, to reach the Industrial Gateway and thus enter the Industrial Network. IPSE therefore acts as a pass-through connector for trusted IoT data. The Industrial Gateway is responsible for any further conversions (e.g. translations, aggregation, etc.) of received data packets before they enter the Industrial Network.

### 9.5 CHARIOT Engine Management in the Fog Network

The CHARIOT Engines discussed in the previous section represent critical functionality of the CHARIOT system. Not only they must be trusted but they must also be resilient, i.e. fault tolerant. Continuous and uninterrupted operations of the engines must be ensured by the CHARIOT Fog Network. Updates to the Engines, for example to their rule based must be carried out while the Engines continue to operate. Appropriate management services of the Fog Network must monitor the operation of the Engines. Ideally, depending on the resources available in the Fog Network, each Engine should be hosted on a separate physical or virtual server to ensure independence of their operation from common disruptions.
10 IoT Analytics

There’s a growing need to analyze fast data at the network’s edge. These IoT devices range from smartphones to streetlights to cars, and all the way up to large industrial systems such as gas turbines. The challenge is how to process the huge amounts of IoT data generated by such devices and make sense of it. Fog/edge computing has been proposed to be integrated with Internet of Things (IoT) to enable computing services devices deployed at network edge (Lin et al, 2017).

Not only volumes but velocity of IoT data is a problem: One major challenge is that business have to overcome obstacles with network traffic: By the time data from hundreds or thousands of IoT devices is transmitted to a central location or the cloud for quick analysis the data can lose its value, even if transmission only takes a few minutes. Real time response in milliseconds or even lower is crucial for many systems for industrial automation and control.

10.1 Edge and Fog IoT Analytics

According to a report by IDC Futurescape\(^\text{14}\), by 2018, some 40 percent of IoT-created data will be stored, processed, analyzed, and acted upon close to, or at the edge, of a network - an approach known as *edge computing*. Use cases for edge computing include amongst others, offshore oil rigs, parking spaces, smart lighting and parking — almost any application with remote devices or applications that require a lot of bandwidth. A Pervasive Computing journal article pointed out that high-end metropolitan-area networks have only a bandwidth of 100 Gbps, which can support uploads of 1080p streams from only 12,000 concurrent users at YouTube’s recommended upload rate of 8.5 Mbps\(^\text{15}\). Thus, to support a million concurrent uploads it would require 8.5 Terabytes per second. Examples such as the above highlight the need for IoT data processing close to the actual IoT devices.

10.2 Fog Analytics in CHARIOT

CHARIOT architecture introduces the concept of the CHARIOT Fog Network, a set of locally available computing, storage and networking resources. Collectively these are known as the CHARIOT Cognitive Platform\(^\text{16}\). The CHARIOT Platform provides networking, storage and computational resources to the CHARIOT subsystems, components and services.

10.3 Cloud Analytics in CHARIOT

For IoT data where immediate analysis is not crucial, and/or the volumes exceed the capacity of the Fog network, a backend Cloud based IoT analytics service is utilised. The IoT Cloud Analytics Service is a remote service accessible from the Fog Network through secure channels. The IoT Cloud Analytics service performs two main functions:

- Stores and analyses (using appropriate machine learning techniques) sensor data and metadata.
- Communicates mined rules back to the CHARIOT Fog Network which are utilised by the CHARIOT Engines.


\(^{15}\) https://www.rtinsights.com/why-edge-computing-and-analytics-is-crucial-for-the-iot/

\(^{16}\) Delivered by Task 2.2 ‘CHARIOT IoT Cognitive Platform’
The Cloud Analytics Engine shares domain knowledge with the CHARIOT system in the following ways:

- Common identification scheme for sensors. It allows the two systems to agree on the naming of sensors and about which sensor(s) the data refer to.
- Common typology/classification of sensor types. It allows sensor data and metadata to be classified accordingly.
- Some shared knowledge of the industrial domain. It allows the sharing of descriptions of sensor location knowledge (topology) for instance.
- Agreement on some standard data processing functions such as MIN, MAX, AVG etc, allowing both descriptions of sensor data uploaded to the Cloud service as well data processing (analytics) queries to be submitted by the CHARIOT services to the Analytics Service.

For this purpose, a common sensor description language such as Sensor Model Language (SensorML) can be utilised.

The Cloud Analytics Service becomes more effective as more sensor data (both valid and invalid) are stored and analysed. The Service is used in an off-line mode where the results of common analytical queries are cached in the Fog Network and are continuously synchronised off-line with the Analytics Service.

Cloud Analytics services are the responsibility of subtask 2.2.4 ‘Analytics and Cognition Services’.
11 Architectural Use Cases

The following three use cases demonstrate the main principles of the CHARIOT architecture in the domains of building management, rail safety and airport security, corresponding to the Project’s Living Labs’ use cases. The use cases purpose is to highlight scenarios where CHARIOT engines and functions are combined to achieve a safety/security/privacy goal. Similar (but more detailed) use cases are expected to be defined by the Project’s Living Labs in Work Package 4 (‘Demonstration of concept in Living Labs’).

11.1 Building Management

11.1.1 Use Case: Detection and mitigation of unsafe HVAC malfunctioning scenario.

Faults due to malicious acts or due to equipment malfunctioning can have safety implications for heating ventilation and air-conditioning (HVAC) systems.

In this use case, the HVAC system of an office building is supervised by CHARIOT. This means that the functioning of existing HVAC sensors deployed in the building is monitored and evaluated by CHARIOT engines.

In this scenario, the air temperature sensors from a particular office area transmitted values constantly fluctuate between the values of 10 and 40 degrees Celsius. As a result, the HVAC controller alternates between cooling and heating up the office space, but due to wide discrepancies between the two temperature ranges the whole HVAC controller could malfunction. This would result in the ventilation unit to stop operating and quickly the office to be filled with unsafe levels of carbon dioxide, creating a safety hazard.

Without CHARIOT functionality in place, it would take longer to detect the deteriorating air quality in the office and the health and safety of occupants would be compromised.

The CHARIOT system however can achieve an early detection, diagnosis and mitigation as follows:

The IPSE Engine detects that sensors transmit temperature values that are within the normal ranges, and that the sensor’s firmware is up to date. However, the rate of value fluctuation as checked against historical data from the Cloud IoT analytics service indicates an anomalous condition.

CHARIOT IPSE Engine also collects status reports from the Safety and Privacy Engines, confirming that no security violations have been detected. However, the Safety Engine calculates the risk that the malfunctioning of the temperature sensor can result in the breakdown of the HVAC system with potential health and safety consequences. Thus, IPSE prevents the erroneous sensor values to enter the Industrial Network (Building Management System). The Safety Reports diagnosing the abnormality are picked up by the Maintenance Department that further investigates the cause of malfunctioning on the sensor.

The sequence diagram of the CHARIOT operations is illustrated in Figure 11.1.
11.2 Airport Security

11.2.1 Use case: Privacy and security violation scenario using hijacked IP cameras.

This use case involves privacy intrusion of airport passenger data with the use of hijacked (remotely managed) IP cameras. According to this use case scenario, the firmware of some of the security IP cameras operating at the airport has not yet patched to a secure level, making the cameras vulnerable to remote attacks using HTTP commands before any password authentication checks take place. Without CHARIOT, it is possible that before the intrusion is detected, the hijacked cameras have already been used to transmit images and video of airport passengers thus violating their privacy rights. Also, transmission of images or videos of safety critical areas in the airport can represent a security risk.

With CHARIOT an early detection, diagnosis and prevention or mitigation is possible. Checking Firmware validity against data stored in the CHARIOT Blockchain, the Security Engine reports a security violation. The Safety and Privacy Engines also monitor the IoT data for abnormal activity. They analyse sensor data packets and detect the presence of images in data transmitted by IP cameras, indicating that malicious agent has possibly infiltrated the LAN where the IP cameras are connected to, seeking to hijack (remotely control) the cameras. The Privacy Engine confirms that the transmission of such data cameras will constitute a privacy violation hazard.

IPSE diagnoses a potential security event with security, safety and privacy implications. It instructs the IoT Gateway to disconnect the affected cameras from the network. It communicates its diagnosis to the IT operations centre in order to take appropriate actions.

The above is illustrated in the sequence diagram of Figure 11.2.
11.3 Rail Safety System

11.3.1 Use Case: Monitoring train conditions for safe operations

Railway technologies is rapidly advancing towards autonomously or cooperatively driven trains. Systems like trackside traffic management system (TMS), supply the train/driver with information about the rail network, while systems such as ETCS monitors the train’s movement to ensure it adheres to the local speed limit and its own permitted top speed. Systems such as ATO, use line data, schedule data and real-time information from the infrastructure to constantly drive at an optimized speed profile, thus making additional energy savings.

To implement proactive safety control requires situation-aware integration and transmission of safety information obtained from IoT sensors. Thus, integration and transmission of safety information should be performed with IoT sensors providing the safety information proper for faced situation.

Predictive maintenance is a good example of the types of benefits afforded to rail systems by IoT: On-board analytics can monitor acoustic vibration data for bearing problems, temperature sensors for brake issues, and even track RFID and photo data to associate potential issues to particular rail cars or components. Control and safety subsystems can also utilize IoT sensors for fire suppression, digital video surveillance, and air conditioning.

In this scenario, CHARIOT acts as an additional safety layer to existing onboard and track-side rail safety systems. In addition, CHARIOT acts as a parallel model to existing safety infrastructure on the train, thus in principle lessening the risks of being afflicted by the same factors such as faults and malicious activity, and acting as an independent backup security system.

CHARIOT for example can compare information received from the sensors about the current train speed and accelerating/decelerating braking patterns against historical data provided by Cloud IoT Analytics Service. CHARIOT employs information from available sensors connected to the CHARIOT IoT Gateway to obtain train location, time information and environment (e.g. temperature characteristics).
If Safety Engine detects that the driving characteristics of the train, significantly differ from historical values, then though an HMI presents an advisory diagnostic to the driver, explaining the discrepancy between current driving pattern and historical driving patterns.

Of course, a future CHARIOT system that is compliant with the relevant rail safety standards could also be connected to the train driving systems such as the braking system.

The above use case scenario is illustrated in the following sequence diagram.

Figure 11.3: Sequence diagram for Rail Safety Use Case
12 Conclusions

CHARIOT is a technically ambitious and complex system. It combines advances in IoT, cybersecurity, blockchain, Fog cognitive and Cloud computing, and IoT analytics. Conceptually, it serves as an additional layer of ‘safety/security/privacy’ protection for IoT, in addition to any existing safety and security systems already deployed. In this manner, it can be considered as part of the new generation of Intrusion Monitoring Detection and Prevention Systems. However, while such systems only address (cyber)security, CHARIOT also addresses safety and privacy. But CHARIOT also addresses the design aspects of IoT safety, security and privacy, by proposing the use of static source and binary code analysis of IoT devices to ensure that any accidental or deliberate design and coding flaws are detected early, before the IoT device is deployed.

CHARIOT also advocates the importance of installing and running only the latest QA tested firmware on IoT devices. Many IoT devices currently run outdated firmware exposing the IoT Network and every other system and network that connects to it to vulnerabilities due to lack of quality assured firmware.

The added benefits that CHARIOT brings are due to its augmented capacity to process environment (i.e. sensor) data and to evaluate it from a combined safety, security and privacy perspective. It has broader functionality and scope than for example, current cybersecurity tools and a more extended situation awareness ability than current safety related systems, because it can detect a wider range of IoT malfunctioning—not only security breaches. This allows CHARIOT to be deployed in active critical infrastructure monitoring, prevention and detection tasks.

CHARIOT can coexist with safety and security systems already deployed and augment their capabilities, or act as a parallel / backup system to increase the overall resilience level. CHARIOT implementations need of course to consider the requirements of the industrial system where they are deployed and to meet their specific performance, scalability, resilience, security and functional requirements.

The CHARIOT architecture approach has already demonstrated its versatility by been configured for three diverse application domains (building management, airport security and rail safety) comprising many heterogeneous IoT technologies and ICT systems. The architecture described in this report is going to be verified in ongoing trials in the Project’s three Living Labs and a revised version of it reported in Month 25.
13 References


[UKGOV] The Internet of Things: making the most of the Second Digital Revolution A report by the UK Government Chief Scientific Adviser


ISO/IEC 27001:2013 Information technology Security techniques Information security management systems Requirements


Lazarte M. Are we safe in the Internet of Things? ISO. 5 September 2016


The Industrial Internet of Things Volume G1: Reference Architecture Copyright © 2017, Industrial Internet Consortium

Xiaohui Research on Safety Certification and Control Technology in Internet of Things. Fourth International Conference on Computational and Information Sciences (ICCIS), 2012.
Annex I: Architecture Model of the Rail Living Lab

Figure A1.1 Implementation architecture of the CHARIOT Rail Lab.

Figure A1.1 shows an implementation of the CHARIOT architecture for the purposes of the Rail Safety Living Lab, as it stands in May 2018. Key components of the architecture are shown in the diagram, together with reference implementations of IoT sensors and devices, and the network technologies that connect them.

In this implementation, train data (measuring critical parameters of train components such as vibrations, temperature, etc.), are collected via wired and wireless sensors and uploaded to Panthora which is the reference Southbound IoT Gateway for CHARIOT.

In turn, Panthora uploads the sensor data to both the CHARIOT System (‘CHARIOT Platform’) and to the Cloud IoT Analytics service.

The architectural implementation of Figure A1.1 is currently undergoing trials and results will be reported by Work Package 4 tasks, from Month 20 onwards.
Annex II: Reference Prototype of the CHARIOT System

Figure A2.1: Realisation of a CHARIOT Network using a cluster of Raspberry Pi single board computers.

Figure A2.1 shows a proof of concept realisation of the CHARIOT system using a cluster of Raspberry Pi single board computers. As shown in Figure A2.1, each cluster node undertakes a role as a component of the CHARIOT system, i.e. as a gateway (both northbound and southbound), dispatcher and so on.

As shown in the figure, to the southbound gateway several sensors are attached that measure, sound, light intensity, air pressure, temperature etc.

To the northbound gateway, an actuation system is attached, allowing the gateway to connect to electro-mechanical automation systems that correspond to the industrial control system, for example to HVAC systems.

The cluster nodes communicate with each other using the MQTT protocol, over a wireless LAN. MQTT is used as a client server protocol where the various CHARIOT components and services act as client and receive IoT data packets from a dispatcher that acts as an MQTT server. Figure A2.2 shows logs of communications between clients and server. Figure A2.3 shows the results of performance tests executed to measure the system throughput, i.e. how fast the system can process streams of IoT data packets.

This is important as many industrial systems impose several performance constraints, including required response times, and CHARIOT must be capable of meeting such performance requirements.
Figure A2.2: Use of the MQTT protocol in the CHARIOT network
Figure A2.3: Measuring the performance of the CHARIOT Network
Annex III: CHARIOT Reference Gateway

Panthora, as shown in Figure A3.1 is the reference IoT Gateway of CHARIOT. It is capable of connecting to several types of IoT devices on one end, and to client systems on the other end and upload to them sensor data.

The maximum number of sensors handled by Panthora are:

- 32 MOBUS devices
- 1 RS-232 device
- 6 “Dumb” analogue sensors
- 16 Digital On / Off sensors
- Many BLE devices (depends on advertising settings)
- 16 Digital Outputs for actions

The Gateway was an in-house design of the TCS CHARIOT partner who have also developed its firmware.

As shown in Figure A3.3, Panthora supports many network and communication protocols, both wired and wireless. The wired sensors are connected in BUS topology (MODBUS & I2C sensors) or in point to point (RS-232 sensors, analogue sensors). The wireless sensors (Bluetooth Low Energy- BLE) are connected in Central / Peripheral mode or in Broadcaster / Observer mode.

Panthora can work with sensors that have proprietary firmware (‘COTS‘), provided that an SDK, driver or other connectivity interface is provided by the manufacturer. This includes for example MODBUS sensors. TCA has also developed inhouse firmware for BLE and Wi-Fi sensors.

The sensors are connected to the Gateway wireless or wired. The Gateway can connect to the Internet wired by using 10/100 LAN interface or wireless with 2G/3G/4G Modem). The Northbound Internet connection employs standard UDP SNMP protocol (Figure A3.3).

Panthora is capable of updating the firmware of attached sensors through a remote interface.

Currently, Panthora has been integrated with CHARIOT systems providing the Blockchain service, while further tests and integrations are ongoing.
Figure A3.1: Gateway / Sensors interconnection
Figure A3.2: Gateway / Internet communication
Figure A3.1: Gateway Hardware structure