Project Acronym: SecureIoT
Grant Agreement number: 779899 (H2020-IoT03-2017 - RIA)
Project Full Title: Predictive Security for IoT Platforms and Networks of Smart Objects

DELIVERABLE D2.5 - Architecture and Technical Specifications of SecureIoT Services_Final version

Deliverable Number	**D2.5**
Deliverable Name	**Architecture and Technical Specifications of SecureIoT Services_Final Version**
Dissemination level	Public
Type of Document	Report
Contractual date of delivery	30/06/2019
Deliverable Leader	AIT
Status & version	v1.50
WP / Task responsible	WP2 (FUJITSU) / T2.4 (AIT)
Keywords:	Security, Architecture, Probes, Data Collection, Security Monitoring
Abstract (few lines):	This deliverable presents the final version of the architecture of the SecureIoT security monitoring platform. The document presents the architecture following the 4+1 architectural view model. The outcome architecture is the basis for the setup and execution of the project's use cases.

Deliverable Leader:	Athens Information Technology (John Soldatos, Sofoklis Efremidis)
Contributors:	Nikos Kefalakis (INTRASOFT), Juergen Neises (FUJITSU)
Reviewers:	Daniel Calvo Alonso (ATOS), Giannis Ledakis (SiLO), Stylianos Georgoulas (INTRASOFT)
Approved by:	Stylianos Georgoulas (INTRASOFT)

Executive Summary

This deliverable presents the final version of the SecureIoT architecture, which has been defined to provide security support to IoT systems and applications. The architecture is presented as a set of modules, along with the structuring principles that drive their integration in an IoT security system that emphasizes security monitoring, security analytics and security automation. In order to define these modules and their structuring principles, SecureIoT has taken into account a number of reference architectures for IoT systems (like RAMI4.0 and the Reference Architecture of the Industrial Internet Consortium), as well as related security frameworks (such as the Industrial Internet Security Framework). The first version of the Deliverable has presented the Reference Architectures that influenced the definition of the SecureIoT architecture. The reference architectures have been exploited as means of understanding the structure of the IoT systems to be monitored and protected, while the security frameworks have been used as basis for defining building blocks for security data collection and analysis, as well as security monitoring and automation.

The SecureIoT architecture is a data-driven architecture, which is defined as an overlay layer that protects all assets of an IoT system or application. It collects security-related information for IoT devices through a number of probes, while at the same time it analyses this information to identify abnormal behaviours and instigate alerts or invoke security automation functions. The architecture specifies a powerful "templates" mechanism, which allows for the characterization and identification of abnormal or suspicious behaviours, based on the execution of advanced data analytics algorithms over the security-related data that are collected from the various probes. It also provides the means for customizing and contextualizing the various templates according to the status of the IoT system that is being protected. Furthermore, the SecureIoT architecture introduces an IoT security knowledge base component, which can be used to match identified abnormal or suspicious behaviours with known vulnerabilities or attacks.

The architecture provides also the means for collecting and analysing information from different IoT devices and platforms based on appropriate probes and a common modelling of the security-related information regardless of the platform for which it is collected. As such it can also support monitoring and protection of IoT systems and applications that span multiple IoT platforms, as part of security interoperability scenarios.

In this document the various components of the SecureIoT architecture are described along with the interactions and information flows between them. The architecture is presented following the 4+1 architecture views, in which the logical, development, process and physical views are presented as sets of models. The four views of the architecture are complemented with the modelling of the scenarios that will be used for its validation.

Based on the defined architecture, the project creates an IoT security platform that provides security services to IT systems and applications, namely Risk Assessment, Compliance Auditing and Developer support. These services will be used to support the project's use case scenarios and thus validate the SecureIoT architecture. This platform exposes a set of APIs to developers and deployers of IoT security services, which have been specified in the deliverables of the technical workpackages (WP3, WP4, WP5, and WP6) of the project.

The document is the final version of the SecureIoT architecture and a refinement of the previous deliverable D2.4. Besides the refinement of the components that support the three SECaaS services and the relationships among them, the present form of the architecture introduces the following:

1. A data bus for facilitating the communications between components.
2. Support for security policy interoperability among multiple platforms on which an IoT application may be deployed.
3. Metrics and utility calculations for assessing the trustworthiness of IoT and platform components.
4. Support for Service Level Agreements between IoT platforms and the SecureIoT platform.

Document History			
Version	**Date**	**Contributor(s)**	**Description**
0.01	20/5/2019	Sofoklis Efremidis, John Soldatos (AIT)	Document structure
0.02	27/5/2019	Sofoklis Efremidis, John Soldatos (AIT)	Initial text on overall architecture
0.10	3/6/2019	Sofoklis Efremidis, John Soldatos (AIT)	Additional text on overall architecture
0.20	10/6/2019	Sofoklis Efremidis, John Soldatos (AIT)	Updates to Logical View model and corresponding text
0.40	14/6/2019	Sofoklis Efremidis, John Soldatos (AIT)	Updates to Logical View model and corresponding text
0.50	17/6/2019	Sofoklis Efremidis, John Soldatos (AIT)	Logical View models completed
0.60	21/6/2019	Sofoklis Efremidis, John Soldatos (AIT)	Process view, Deployment view models
0.65	25/6/2019	Nikos Kefalakis (INTRA)	Data Models chapter and XSDs
0.70	5/7/2019	Sofoklis Efremidis, John Soldatos (AIT)	Text harmonization
0.75	8/7/2019	Nikos Kefalakis (INTRA)	Updated overall architecture figure
1.00	8/7/2018	Sofoklis Efremidis, John Soldatos (AIT)	Final text editing, submitted to internal review
1.10	10/7/2019	Sofoklis Efremidis, John Soldatos (AIT)	Interim version after first review comments
1.15	12/7/2019	Nikos Kefalakis (INTRA)	Updated the Data Models chapter and XSDs
1.20	15/7/2019	Sofoklis Efremidis, John Soldatos (AIT)	Interim version after ATOS comments
1.30	22/7/2019	Sofoklis Efremidis, John Soldatos (AIT)	Interim version after Instrasoft and SiLO comments
1.40	23/7/2019	Sofoklis Efremidis, John Soldatos (AIT)	Final version after Intrasoft comments
1.50	26/7/2019	Sofoklis Efremidis John Soldatos (AIT)	Final version after minor updates

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

Table of Contents

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

Table of Figures

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

List of Tables

Definitions, Acronyms and Abbreviations

Acronym	Title
AI	Artificial Intelligence
API	Application Programming Interface
CAPEC	Common Attack Pattern Enumeration and Classification (MITRE)
CAS	Compliance Auditing Service
CMDB	Configuration Management Database
CPE	Common Platform Enumeration (MITRE)
CTI	Cyber Threat Intelligence
CVE	Common Vulnerabilities and Exposures (MITRE)
CVSS	Common Vulnerability Scoring System (NIST)
CWE	Common Weakness Enumeration (MITRE)
CVSS	Common Vulnerability Scoring System
IISF	Industrial Internet Security Framework
IKSB	IoT Security Knowledge Base
JSON	Javascript Object Notation
ML	Machine Learning
NIST	National Institute of Standards and Technology (U.S.A.)
NVD	National Vulnerability Database
ORA	Open reference Architecture
OSI	Open Systems Interconnection
PDP	Policy Decision Point
PEP	Policy Enforcement Point
RAM	Risk Assessment and Mitigation
RAE	Risk Assessment Engine
RAMI 4.0	Reference Architecture Model Industrie 4.0
SKB	Security Knowledge Base
SLA	Service Level Agreement
SPEP	Security Policy Enforcement Point
STIX	Structured Threat Information Expression
TEE	Template Execution Engine
UML	Unified Modelling Language
XACML	eXtensible Access Control Markup Language
XML	eXtensible Markup Language

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

1 Introduction

1.1 Scope and Purpose

The main goal of SecureIoT is to introduce, validate, and promote a novel approach to the security of IoT applications, which emphasizes a timely, predictive and intelligent approach to the identification and mitigation of security threats and incidents. One of main characteristics of this approach is its ability to deal with smart objects, while at same time supporting security interoperability in supply chain scenarios that may involve multiple IoT systems and platforms with diverse security capabilities. The project will reflect its approach to an architectural concept, which will serve as a basis for implementing predictive and intelligent security systems. Based on this overarching architectural concept, the project will develop concrete security services that will be validated in the scope of the project's use cases.

In this context, the purpose of the present deliverable is to introduce the SecureIoT architecture, along with the architecture of the security platform that will be developed in the project. The latter architecture will be introduced and presented based on a number of interrelated modules and the interfaces between them. Moreover, the deliverable provides the detailed specification of each one of the modules of the architecture. The outcome of the deliverable is the specification of the SecureIoT architecture and the technical specification of the security platform and services that will be implemented based on this architecture. The SecureIoT architecture will serve as a reference for implementation of predictive security systems for IoT platforms and smart objects, through providing proper placeholders for the integration of data-driven security intelligence, beyond the range of algorithms that will be implemented in the project.

The deliverable is important in the scope of the project's workplan, as it is expected to drive the project's developments in other workpackages. In particular:

- The specification of the building blocks of the SecureIoT platform (and of related compliant systems) will give rise to their detailed design and implementation in other technical workpackages (notably WP3 and WP4).
- The specification of the interfaces between the building blocks of the architecture will drive the integration of the SecureIoT systems and services in WP5, as well as their use in the scope of the use cases in WP6.
- The architecture will provide the anatomy of the SecureIoT platform and will shed light on how it should be integrated and customized in order to support the needs of the use cases.
- Finally, the building blocks of the architecture and the SecureIoT platform will provide insights regarding the security modules and results that could become part of the project's marketplace / ecosystem platform in WP7.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

1.2 Background and Vision

The vision of the SecureIoT project is to provide new insights on security monitoring and data-driven security intelligence for IoT systems, through enabling the implementation of predictive security systems that can analyse data from IoT platforms and smart objects (i.e. objects with semi-autonomous behaviour). Hence, it is envisaged that the intelligence of the SecureIoT approach will stem from the implementation and integration of data analytics algorithms that will be able to proactively identify security threats and incidents towards more timely preparedness. Nevertheless, part of the SecureIoT intelligence will also stem from the modularity and the dynamic nature of the platform, which will provide the means for dynamically binding new IoT assets (e.g., devices, modules), while at the same time protecting them, based on new workflows and data driven algorithms.

The SecureIoT architecture takes into account the rich set of reference architectural models that have been introduced by standards development bodies and industry consortia regarding the structure of IoT systems. Most of these models (e.g., the Industrial Internet Security Framework (IISF) and the OpenFog Reference Architecture (ORA)) specify security monitoring and data analysis as one of their core security mechanisms. SecureIoT aims at substantiating these mechanisms based on concrete algorithms and security systems middleware that guarantees modularity, extensibility, configurability and scalability.

The architecture will serve as a basis for implementing the security systems of the project's use cases. Hence, three instances of the SecureIoT architecture will be devised and deployed for the three main contexts and application areas of the use cases, namely industrial automation, socially assisted robots and connected cars. Therefore, the project's use cases will be used for the validation of the architecture, that is, the SecureIoT architecture will be validated against its ability to meet the main security requirements of the use cases. Furthermore, our architecture work develops a vision regarding its application in other use cases as well. In particular, the SecureIoT architecture will serve as a blueprint for the implementation of IoT security monitoring platforms, including platforms that comprise advanced data analytics. Moreover, this blueprint will be supported by a range of middleware components that will underpin the SecureIoT platform in-line with the architecture introduced in this deliverable.

The goal of this deliverable is to provide the structuring principles for building standards-based data-driven security monitoring systems for IoT infrastructures and applications, and in particular, systems that can effectively deal with multiple diverse IoT platforms and smart objects.

1.3 Methodology

The methodology for specifying the SecureIoT architecture includes the following activities and steps (Figure 1):

- **Analysis/Review of the state of the art**, in terms of relevant IoT projects and initiatives, notably projects that have dealt with the implementation of data-driven intelligence and data driven security policies. The aim of this analysis is to take into account insights from these projects regarding the implementation of relevant IoT security mechanisms. In

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

addition to reviewing various state of the art initiatives, this methodological step included also a review of reference architectures that have been introduced by industrial consortia and standards development organizations, such as the OpenFog Reference Architecture, the Industrial Internet Security Framework and RAMI4.0 [Schweichhart16]. The review of these reference architectures provided insights about how to structure the architecture both logically and physically, based on the consideration of aspects such as the collection and analysis of data from IoT devices, as well as the specification and implementation of Policy Decision Points (PDP) and Policy Enforcement Points (PEP).

- **Synthesis and Specification of a Security Architecture for IoT systems**, based on the combination of concepts from the analysis of the previous steps, but also based on the specification of solutions that successfully address the main functional and technical requirements of the SecureIoT platform, including the requirements listed in earlier deliverables [SecureIoTD2.1] and [SecureIoTD2.2]. The specification of the SecureIoT Architecture is provided in terms of a description of its main modules and of the interfaces between them. Emphasis is paid on providing both logical and physical views of the architecture, while at the same time documenting both the static and dynamic behaviour of a SecureIoT system that adheres to the architecture. To this end, multiple views of the SecureIoT architecture are provided.
- **Validation of the SecureIoT architecture against the project's scenarios and use cases**, notably the use cases that are specified and implemented in WP6 of the project. The validation aims at ensuring the appropriateness of the architecture for the project's use cases, while at the same time validating that the functional characteristics of systems that follow the principles of the architecture are sufficient to address IoT security for diverse IoT platforms and smart objects.

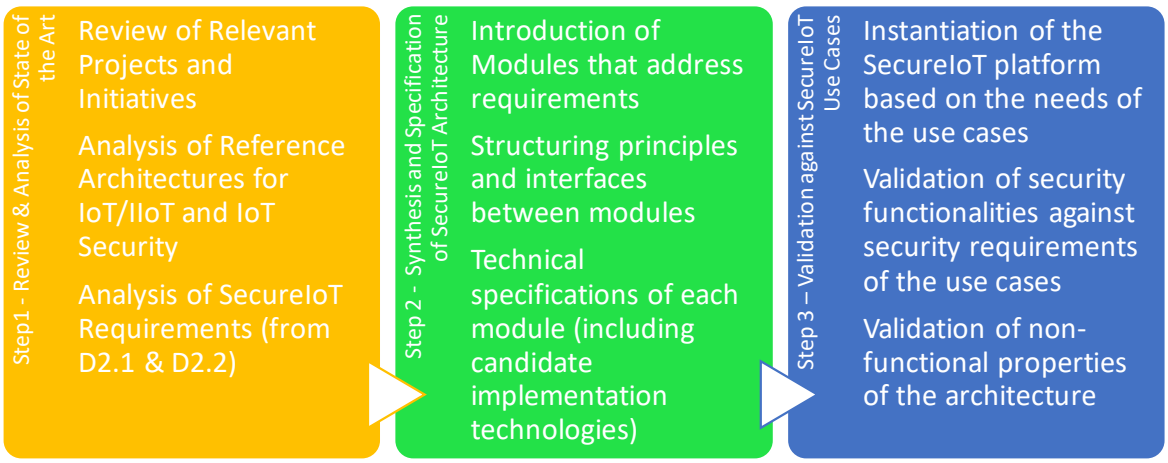
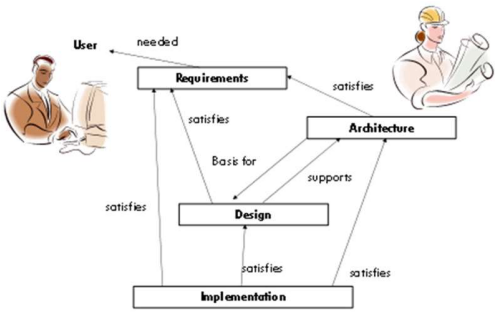
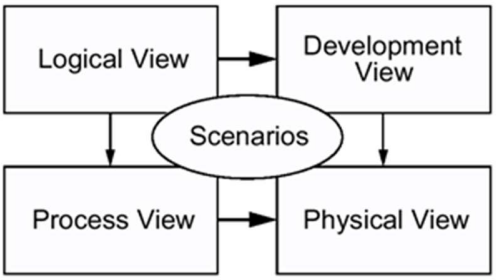


Figure 1: Main methodological steps for the SecureIoT Architecture Specification and Validation.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

In order to serve the objectives of the above-listed methodological steps, the SecureIoT architecture is specified based on multiple views, according to the "4+1" views methodology [Kruchten95]. The latter specifies that an architecture that can be described based on four complementary views, namely, a logical view, a process view, a development view and a physical view. These views are complemented by a set of specified scenarios and use cases, which are used to validate the architecture (side figure). As shown in the side figure, in the 4+1 approach special emphasis is paid in the logical view, which drives the specification of the development, and of the process view. The logical view depicts the high level view of the architecture, including its main components and the way they are structured together. Following the specification of the logical view, a development view can be derived and elaborated in order to provide insight on the implementation task of the architecture, while a process view can be also elaborated in order to show the dynamic behavior of the system, including interactions and information flows between the various components. Finally, the physical view provides insights on the physical implementation and deployment of a system that is based on the specified architecture.





The present deliverable has taken into account WP2 developments (in particular the requirements and reference use cases deliverables i.e. [SecureIoTD2.1] and [SecureIoTD2.2]) for the development of the SecureIoT architecture. The three project scenarios complement the 4 viewpoints of the architecture as shown in the side figure and their specifications are presented in Chapter 5. A typical interaction between the requirements, the architecture, the detailed design and the implementation of the SecureIoT platform has been considered for WP2 and its interaction with the technical work packages, as shown in the side figure. In particular, the architecture of the project satisfies the functional and non-functional requirements of the project, as the latter are specified in [SecureIoTD2.2].

The SecureIoT architecture has been developed based on the requirements expressed as part of the project's representative use cases. The development of the SecureIoT architecture has progressed in two versions; D2.4 presented the first version of the architecture. The present document gives the final and refined version of the architecture based on feedback received from the technical workpackages of the project and distilment of its components and the functions they support. Different views of the architecture have been expressed as result of the feedback received from the technical workpackages of the project. These views have been aligned in a unified form and the respective models have been created.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

1.4 Document Structure

The structure of the deliverable is as follows:

- Section 2 presents the architecture of the SecureIoT platform. It first gives a high level and unified view of the architecture, along with the security services it supports. In the sequel, Section 2 presents different viewpoints of the architecture expressed as a set of UML models following the 4+1 architectural approach.
- Section 3 gives the specification of the security services the platform offers, in particular the Risk Assessment, Compliance Auditing and Developer Support services. For each of the security services further details are given following their modelling contained in Section 2.
- Section 4 gives the data models used by the architecture, which serves the basis for the interfaces between its components.
- Section 5 presents the +1 of the 4+1 architectural views of the SecureIoT architecture, namely, the architecture models of the three project use cases. The +1 view of section 5 follows the views presented in Section 2.
- Finally, Section 6 concludes the document.

2 SecureIoT Platform Architecture

This chapter presents the SecureIoT platform Architecture. The architecture presented in this document is an update of the one described in [SecreuIoTD2.4]. The present final form of the architecture is the result of interactions with the other technical WPs of the project. The requirements for the functionality of the components of the architecture and their interactions have been refined and the result of this refinement is the finalized SecureIoT architecture that is presented in this document. The chapter gives an overview of the architecture and presents its unified form, in which the SECaaS security services are represented. In the sequel, it gives the 4+1 views through the corresponding UML diagrams.

2.1 Overview

The SecureIoT architecture has been defined for a security platform that delivers SECaaS services to various IoT systems/platforms owners or operators, in-line with the high-level model illustrated in Figure 2. According to the model, different IoT systems provide data to the SecureIoT services provider i.e. the entity that is deploying and operating the SecureIoT platform. Accordingly, the SecureIoT services providers provides to IoT systems owners or operators services such as Risk Assessments, Compliance Auditing and Developers' Support, along with a range of security automation (e.g., alerts) and visualization services (e.g., display of information in dashboards), which are typically provided in-line with a security policy and are flexibly accessible and configurable by platform operators and owners.

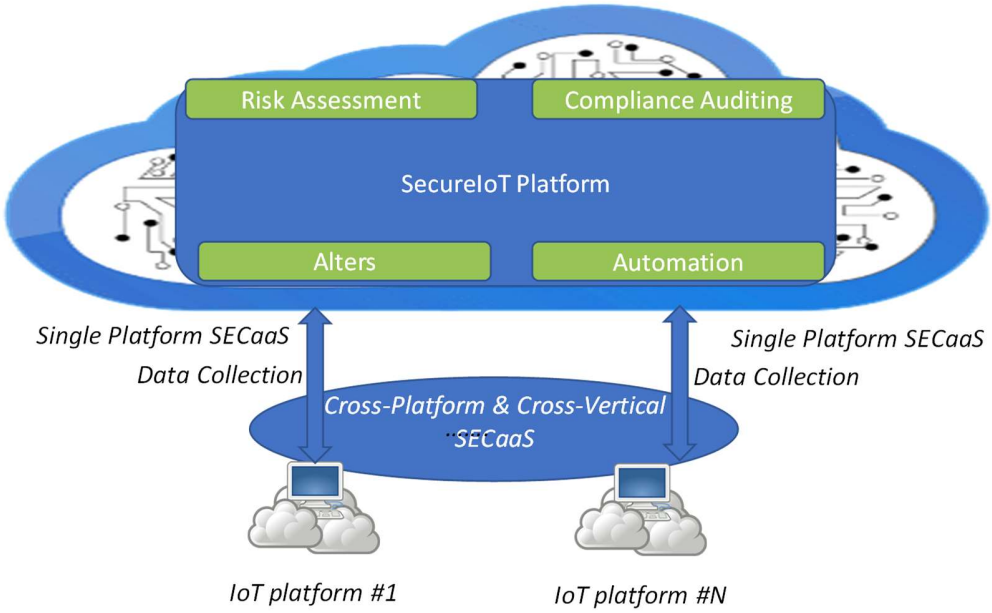


Figure 2: SECaaS Model Overview.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

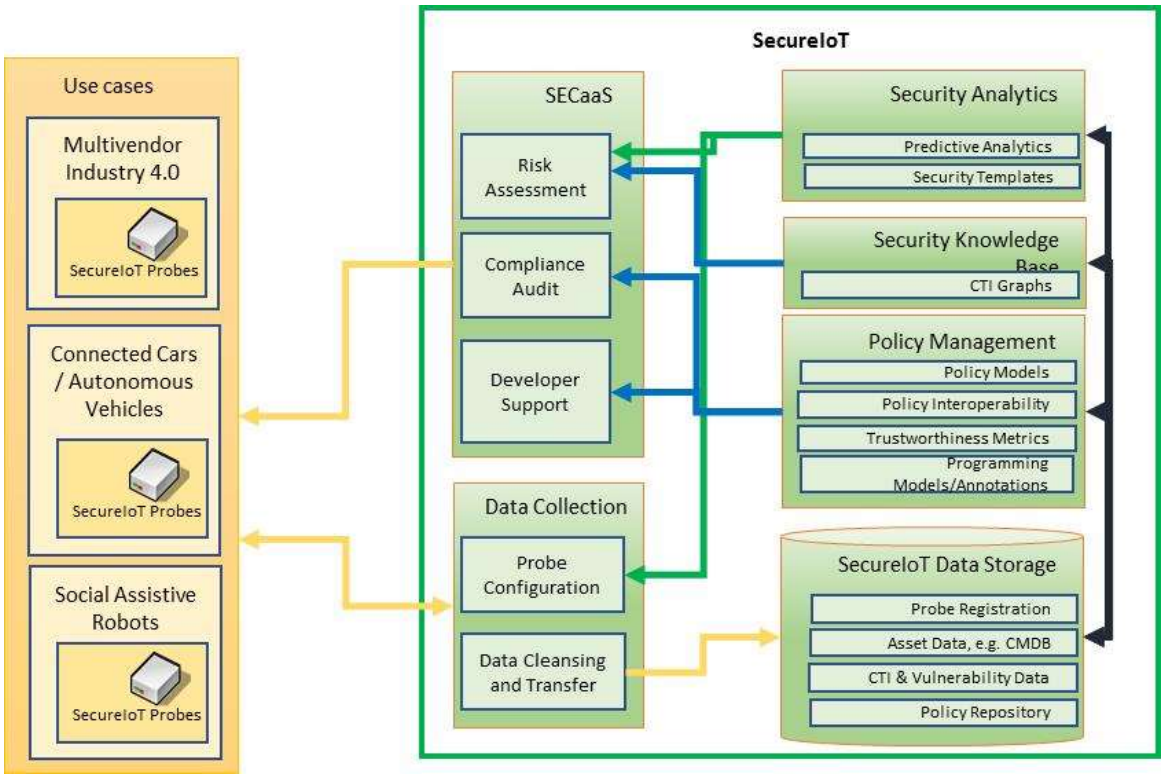


Figure 3: SecureIoT Services Overview.

Figure 3 depicts a high-level view of the SecureIoT services and its interfaces to applications that make use of them. In the course of the projects three representative use cases will validate the SecureIoT services: (a) multivendor Industry 4.0 application, (b) connected cars and autonomous vehicles, and (c) social assistive robots. For the three use cases as well as any applications that may make use of the SecureIoT services, there are two visible points of interaction with the SecureIoT platform.

The first is the set of security services that are provided by it, as follows:

- Risk Assessment: a set of services for the quantification of risks to which an IoT deployment or application may be exposed to, based on the probability and impact each one may have. The NIST CVSS (Common vulnerability Scoring System) is used as the base for this service. [SecureIoTD5.1] contains the specifications of the service.
- Compliance Auditing: a set of services that support auditing of IoT deployments and services against existing sets of security and privacy controls. Auditing services depend on data collected from the IoT deployment as well as output of analytics from their processing. Auditing requirements may include policies that need to be complied with and may also be expressed in XACML. The auditing service outputs sets of

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

recommendations about issues in which compliance may be at risk and require further attention. [SecureIoTD5.4] contains the specifications of the service.

- Developer Support: a set of services for secure IoT programming based on programming annotations that extend and complement existing programming constructs. The services will be expressive enough to allow for enforcement of policies at various levels of the IoT target deployment, i.e., device, smart object, edge, cloud. The developer support services are automatically mapped into controls of the target IoT deployment or application and are used during its operation. [SecureIoTD5.7] contains the specifications of the service.

The second point of interaction of an IoT deployment or application with the SecureIoT platform is through the mechanisms used for supporting those services. Monitoring of the target IoT deployment is the basis for the provided security services and as a consequence the collection of security data from its various levels and their transfer to the SecureIoT platform for processing is required. In particular, security data collection and processing are necessary for supporting the risk assessment and compliance auditing services.

The SecureIoT SECaaS services are supported by a number of internal services and components, as shown on the right part of Figure 3. The main ones include

- Security Data Collection and Monitoring: the service collects security related data from components of the deployed IoT system or application and stores them for further processing. The collected data are used to drive the analytics components that detect patterns of abnormal behavior.
- Security Analytics: the service operates on collected security data and tries to find patterns in them that may be deemed suspicious resulting to the raise of alerts. Security contextual data in the form of template may also be used for tuning the analytics process to the context to which it is applied.
- Security Knowledge Base: it contains security related data, in particular those related to cyber threats. The corresponding CTI graphs are used by the security analytics that must maintain an up-to-date knowledge of cyber threads.
- Policy Management: it is a repository for the policy models and provides functionalities for their management and for policy interoperability. Moreover, it supports the definition of trustworthiness metrics and provides programming support for expressing security requirements.
- SecureIoT Data Store, probe registration: it is a repository of the deployed probes.
- SecureIoT Data Store, asset data: it is a repository of the assets of the target IoT deployment.
- SecureIoT Policy Repository: it is repository of the policies for the target IoT deployment.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

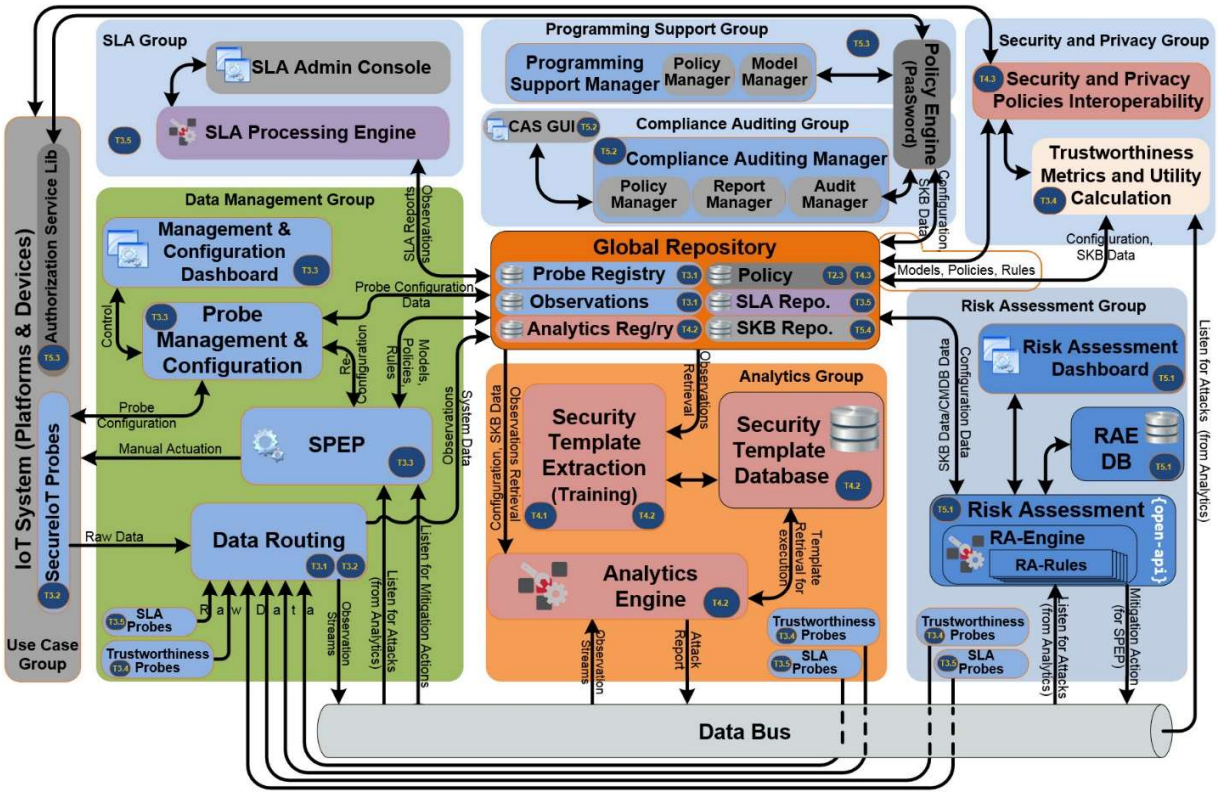


Figure 4: Overall SecureIoT Architecture and mappings to project tasks.

Figure 4 depicts the overall SecureIoT architecture based on the model of Figure 3 and the mappings of its components to the project tasks. The main parts of the architecture are described below.

- Data bus. It is a communications channel through which al real time data are routed. Platform componenents may subscribe to the data bus to receive data of specific interest to them.
- Data management. It is responsible for collecting data from the parts of the target IoT system or application and streaming them to the Observations repository. Its main subcomponents are
  - Deployed probes: they collect data from the target IoT system or application and stream them to the IoT platform through the data routing component.
  - Probe Management and Configuration: the component is responsible for managing and configuring the deployed probes. It interacts with SPEP from which it receives automatic probe configuration commands and correspondingly configures the managed probes. Manual probe configuration commands may also be received by the dashboard. The Management and Configuration

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

dashboard provides a user interface to the Probe Management and Configuration component.

- Probe Registry: it maintains a record of the deployed probes. Probe deployment data, as well as state and configuration data are maintained by the registry. The registry provides probe creation, reconfiguration and search capabilities.
- SPEP (Security Policy Enforcement Point): It is responsible for effecting actuations to the target IoT deployment or application. Two types of actuations are foreseen. First, is the automatic reconfiguration of probes as the result of the output of the Analytics Engine. The probe configuration is done through the Probe Management and Configuration. Second, is the manual of probes and components as the result of the Risk Assessment service. When a risk is identified after vulnerability on an asset is detected, the Risk Assessment dashboard alerts its user for a mitigation action to be taken. The decision is mapped into the appropriate actuation which is conveyed to the IoT element through SPEP.

- Analytics engine. Provides security analytics functionalities based on the monitored security data that are collected by the probes as well as training data and templates. The Analytics Engine trains itself using historic data that are stored in the Observations part of the Global Repository. The outcome of the training the a set of templates that are stored in the Security Template Database, which is used by the Engine for the real time identification of security issues by monitoring real time data that are streamed from the deployed probes through the data bus.
- Risk Assessment service: it checks the target IoT deployment against a set of known vulnerabilities databases, like NIST's NVD or CVSS. The service operates on the collected security data for the target IoT deployment as well as stored CTI data. The Risk Assessment service tries to match patterns of vulnerabilities against the collected data and in case it discovers that the IoT deployment is exposed to a vulnerability it raises an alert. The Risk Assessment service comprises the following components:
  - Risk Assessment Engine that implements the service itself. It makes use of a Risk Assessment database that contains sets of known vulnerabilities and interacts with its user through the Risk Assessment Dashboard.
  - The Risk Assessment Dashboard that provides a user interface to the Risk Assessment Engine. Mitigation actions that are proposed by the Risk Assessment Engine are presented to the Dashboard allowing the user to decide for their enforcement. These actions are eventually carried out through the SPEP component.
- Compliance Auditing service: it checks and verifies the compliance of the target IoT deployment to regulatory related requirements, e.g., GDPR, NIS, etc. The service operates on collected security data as well as regulatory requirements that are expressed as policies. The engine that implements the service verifies that the collected data conform to the regulatory requirements as expressed through the respective

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

policies. If no verification can be secured, it raises alerts to the CAS console. The component comprises the following.

- Compliance Auditing GUI that provides a user interface to the Compliance Auditing Manager.
- The Compliance Auditing Manager that supports the auditing of the target IoT application, the specification of policies and the generation of the respective reports.

- Programming Support service: it provides programming level support for security controls through annotations and a set of policies for backing their handling at runtime. The Policy Engine is responsible for applying the defined policies and is common to the Compliance Auditing and the Programming Support services. The Programming Support Manager comprises the Policy Manager that allows the specification of policies and the Model Manager that allows the definition of the policy models.
- Security and Privacy Policies Interoperability: the group comprises two modules: Trustworthiness Metrics and Utility Calculation and Policy Interoperability. To identify the trustworthiness of each participant (provider of IoT platform), the Trustworthiness module provides a metric related to the IIC model and utility calculation tools that facilitate the assessment of a participant's trustworthiness based on defined policies and configuration data. This way the two modules are closely linked together. It is assumed that the Policy Interoperability module can demonstrate support for XACML-based policies to facilitate the assessment of attributes and rules in a common abstraction, e.g. by enabling a common assessment.
- Global Repository: it comprises a number of repositories for the persistent storage of data that are used by the other components of the architecture. The particular repositories involved reflect the Security and Privacy Policy Models that have been developed in [SecureIoTD2.3]. Thoses models are represented in the content of CMDB, SKB and policy repository that are presented below. The constituent repositories are as follows.
  - Probe registry: it contains a record of the deployed probes along with their attributes, including state information. It facilitates the automatic deployment of probes and their dynamic discovery.
  - Observations: it contains historic security data that have been collected by the deployed probes. These data are used by the Analytics Engine to train itself and produce a set of security templates that will be used subsequently for identifying security issues on the target IoT system.
  - Analytics Registry: it contains information about the analytics algorithms that are employed by the SecureIoT platform and data to configuring their lifecycle, i.e., processor manifest, definition, orchestrator. The Analytics Registry serves similar purpose like the Probe Registry.
  - Policy Repository: it stores the defined policies that prescribe the handling of security issues that are detected in the target IoT system.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

- SLA repository: it contains SLA reports along with configuration data about SLAs.
- Security Knowledge Base repository: it includes publicly available data on known vulnerabilities and risks, and corresponding CTI graphs. In particular the contents of the SKB are
  - CAPEC: MITRE Common Attack Pattern Enumeration and Classification collection
  - CVE: MITRE Common Vulnerabilities and Exposures collection
  - CWE: MITRE Common Weakness Enumeration collection
  - CPE: MITRE Common Platform Enumeration collection
  - CVSS: NIST Common Vulnerability Scoring System collection
- CMDB: it contains information about all assets of the target SecureIoT system along with their attributes and configuration parameters.

- SLA support components: The SecureIoT platform supports IoT applications that are deployed over a number of IoT platforms. A typical example is a supply-chain application which may collect data from a number of sensors that may be supported by different IoT platforms. The architecture that was defined in [SecureIoTD2.4] has been extended with components that support multi-platform IoT applications as reported in [SecureIoTD3.9]. In such scenarios SLAs have to be put in place between IoT platform providers and the SecureIoT platform provider that will govern the obligations of the different parties regarding the collection of security data and the generation of security alerts. The main enhancements include:
  - SLA probes in different parts/layers of the SecureIoT architecture, including probes in the Data Management, Analytics and Risk Assessment Groups. These probes enable the acquisition of (raw) data that are send to the data routing component.
  - Enhancements to the Data Routing component for transforming raw data to properly structured Observations and storing them to the Global Repository (Observations Repo).
  - SLA Processing Engine that processes raw data and generates SLA reports. The latter are stored in a dedicated part of the Global Repository destined to support SLA Management, which is conveniently called (SLA Repo).
  - The SLA Repo that hosts SLA reports along with configuration data about each SLA.
  - The SLA Admin Console, which provides the means for configuring and managing SLAs, based on the processing of configuration data of the SLA Repo.

2.2 Logical View

The high-level logical view of the SecureIoT platform is shown in the model of Figure 5. The model shows: (a) the basic mechanisms used by the platform for providing its security services, i.e., Data Management for collecting data from the target IoT system and applying actions or reconfigurations to it, (b) the three SECaaS services, i.e., Risk Assessment, Compliance Auditing

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

and Developer Support, and (c) supporting services for extending the security services to IoT applications that span multiple IoT platforms that may belong to different administrative domains, i.e., SLA management and Policy Interoperability. The constituent parts of the model are detailed in the sequel.

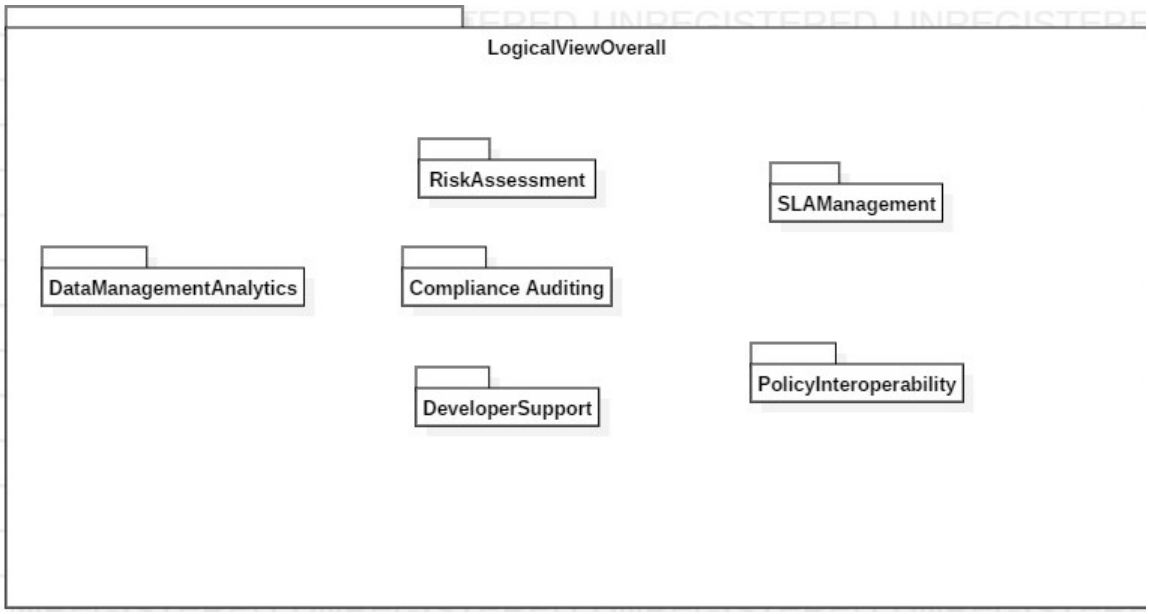


Figure 5: High level logical view of SecureIoT platform.

Figure 6 shows the Logical View model of the Data Management and Analytics part of the SecureIoT architecture, which provides the basic mechanisms of the SecureIoT platform upon which the SECaaS services are built. The SECaaS services depend on data collected through probes from the target IoT system or application to which they are provided. The central notion in the model is that of an Observation, which represents data that are collected from the deployed probes, and they may be possibly normalized. The collected data are fed into the Analytics Engine, as well as the components that implement the Risk Assessment, the Compliance Auditing and the Developer Support Services of the platform.

Observations are generated by the deployed probes and are routed through the data bus. They are subsequently stored in the Observations part of the Global Repository to be retrieved by the analytics engine and the componenets that implement the SECaaS services.

Probes are registered to the Probes Registry and they many be configured through the Management and Configuration component. They receive raw data from a number of sources and they generate observations. Specializations of the Source class of Figure 6 are shown in the model of Figure 7.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

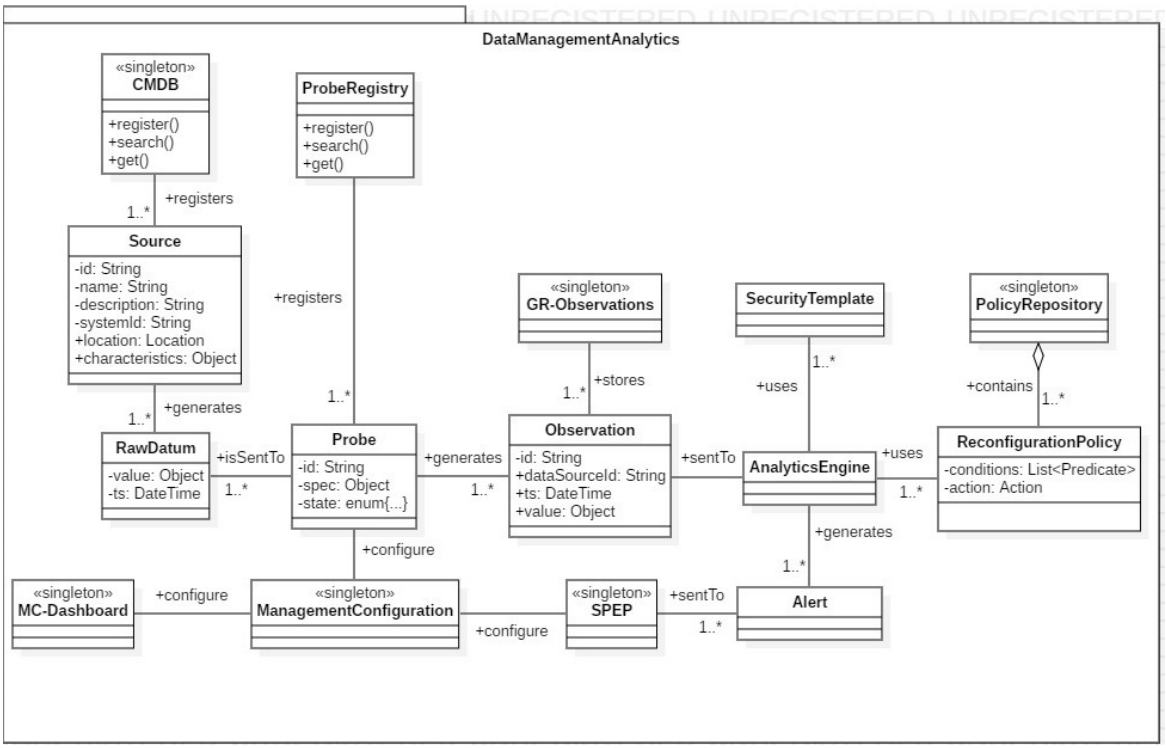


Figure 6: Logical View of the Data Management and Analytics parts of the SecureIoT architecture.

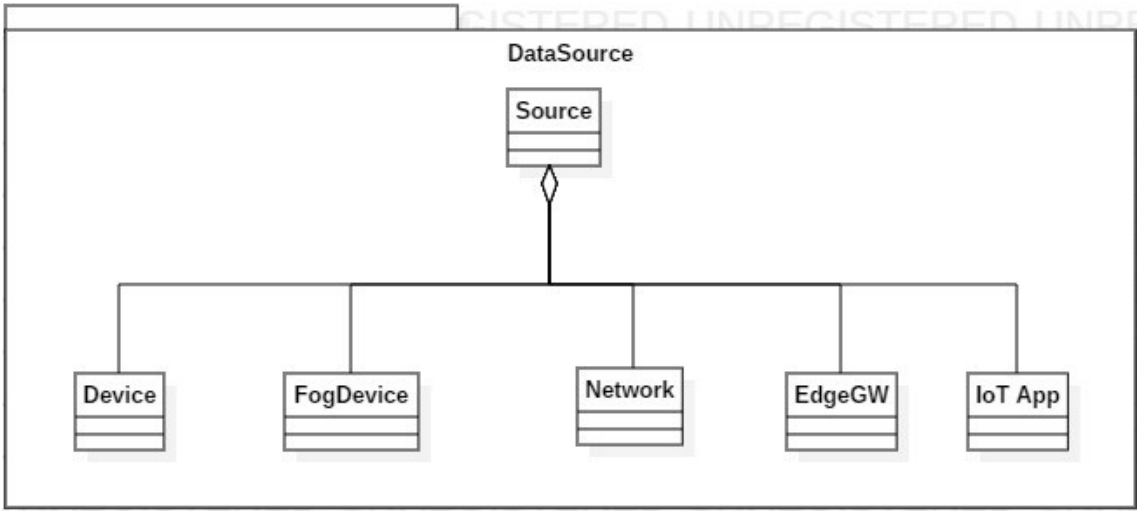


Figure 7: Logical view of data source specialization.

The analytics engine uses historic security data for its training and produces security templates that contain the training models, which are stored in a Security Templates Data Base. The security templates are subsequently used during monitoring of the target IoT system to generate alerts in case security issues are detected. The templates express contextual information as well as the conditions that have to be met for alerts to be generated and actions to be taken. Moreover, the analytics engine makes use of reconfiguration policies, which reside

at a Policy Repository and specify conditions under which reconfiguration of the deployed probes must be enforced.

The Analytics Engine may raise alerts for security issues that may be detected while monitoring the target IoT system. The generated alerts may be intercepted either the SPEP component or by the Risk Assessment Engine. Depending on the defined reconfiguration policies they may carry reconfiguration commands that are to be applied to the deployed probes. Alerts that are intercepted by SPEP result into the reconfiguration of probes, which is effeced through the Management and Configuration componenet. Additionally, probe reconfigurations may be effected manually from the Management and Configuration dashboard.

The model of the Analytics Engine is shown in Figure 8. The engine provides lifecycle support for the analytics algorithms it may run. At any given time the engine may run several algorithms, each one trained with its own training sets and configuration. The training part of the analytics algorithm results into the generation of the security templates that are stored into the Security Templates database and are subsequently used by the monitoring part of the analytics algorithm, which, based on the received observations, generates alerts as discussed above.

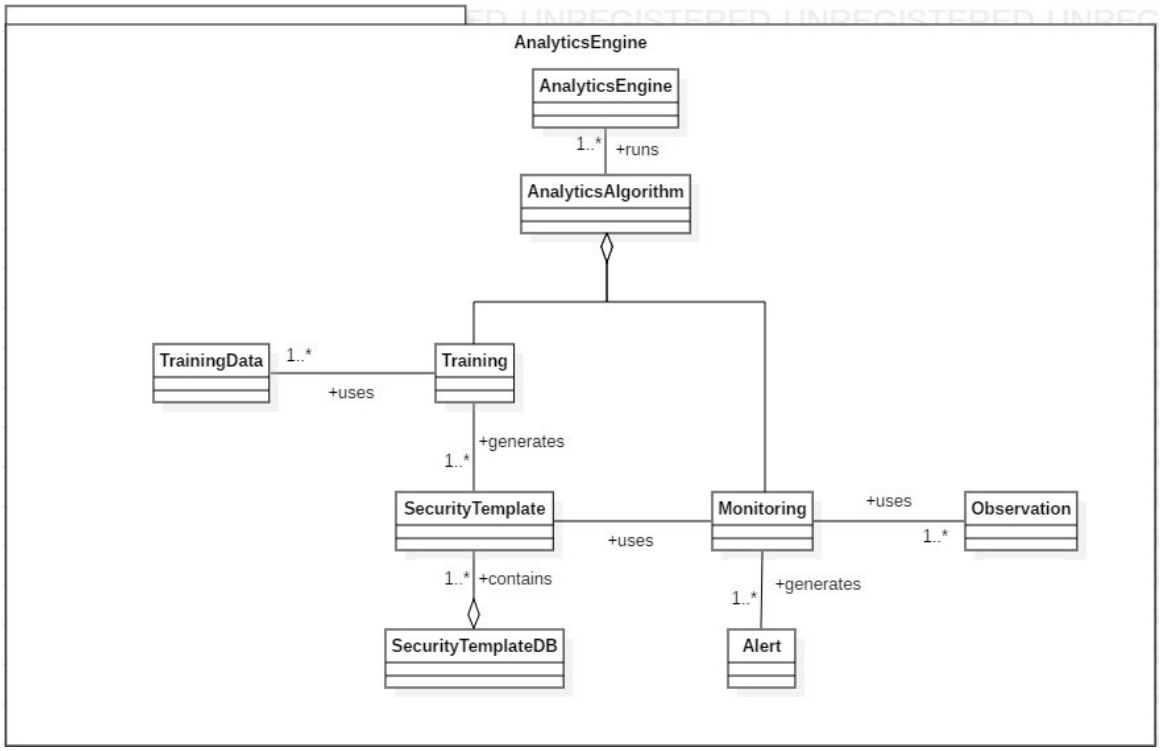


Figure 8: Logical View of the Analytics Engine.

For accommodating multiple analytics algorithms, the Strategy Design Pattern is used by the Analytics Engine as shown in Figure 9.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

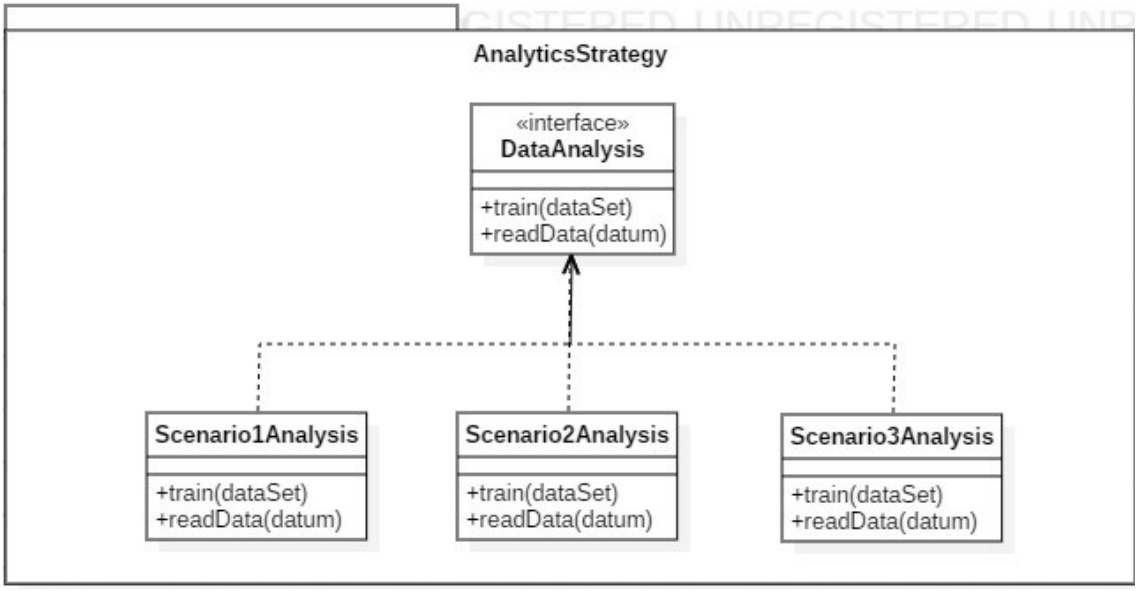


Figure 9: Strategy pattern for data analysis.

The Risk Assessment SECaaS service is implemented by the Risk Assessment Engine. The Risk Assessment Engine intercepts the alerts that are produced by the Analytics Engine and proposes mitigation actions based on the contents of the Security Knowledge Base, which, as noted above, contains

- CAPEC: MITRE Common Attack Pattern Enumeration and Classification collection
- CVE: MITRE Common Vulnerabilities and Exposures collection
- CWE: MITRE Common Weakness Enumeration collection
- CPE: MITRE Common Platform Enumeration collection
- CVSS: NIST Common Vulnerability Scoring System collection

The proposed mitigation actions are communicated to the SecureIoT platform operator, and, if accepted, result into actions to be applied to the target IoT system. The corresponding directives are sent to the SPEP component, which is responsible of mapping the high level action directives to the low level reconfiguration commands for the deployed probes. Probe reconfiguration is eventually done through the Probe Management and Configuration Component. Figure 10 depicts the model for the Logical View of the Risk Assessment service.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

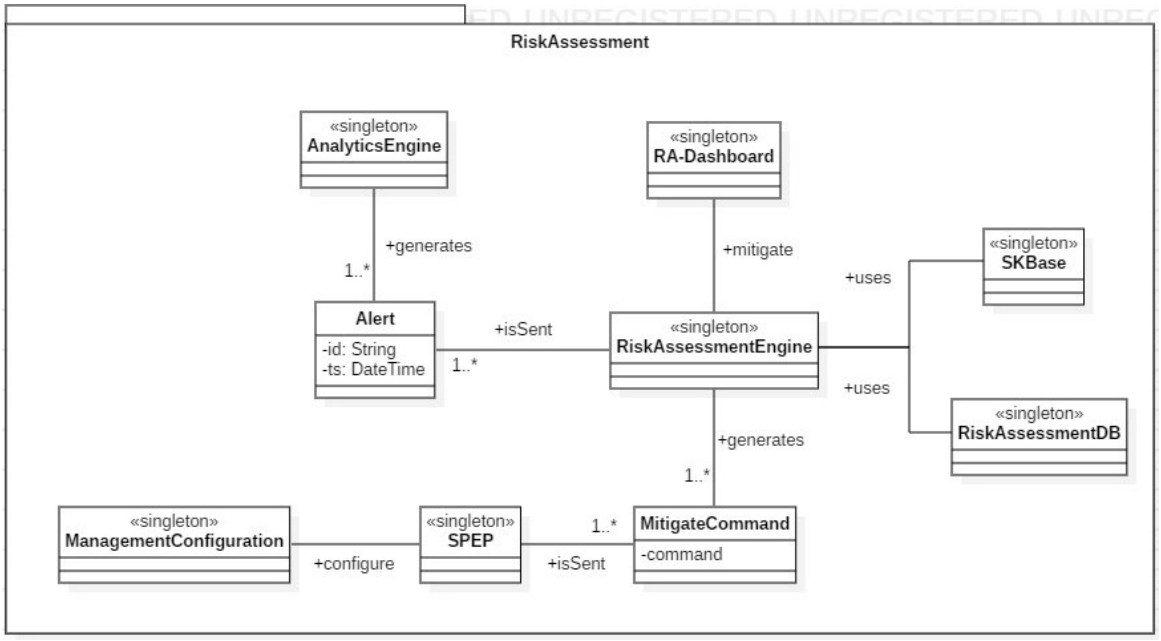


Figure 10: Logical view of the Risk Assessment service.

The Compliance Auditing Service makes use of Compliance Policies that reside in the Compliance Policy Repository. It monitors the generated observations and produces alerts when compliance violations are identified based on the applied policies. The alerts are communicated to the SecureIoT platform operator. Figure 11 depicts a model for the Logical View of the Compliance Auditing service.

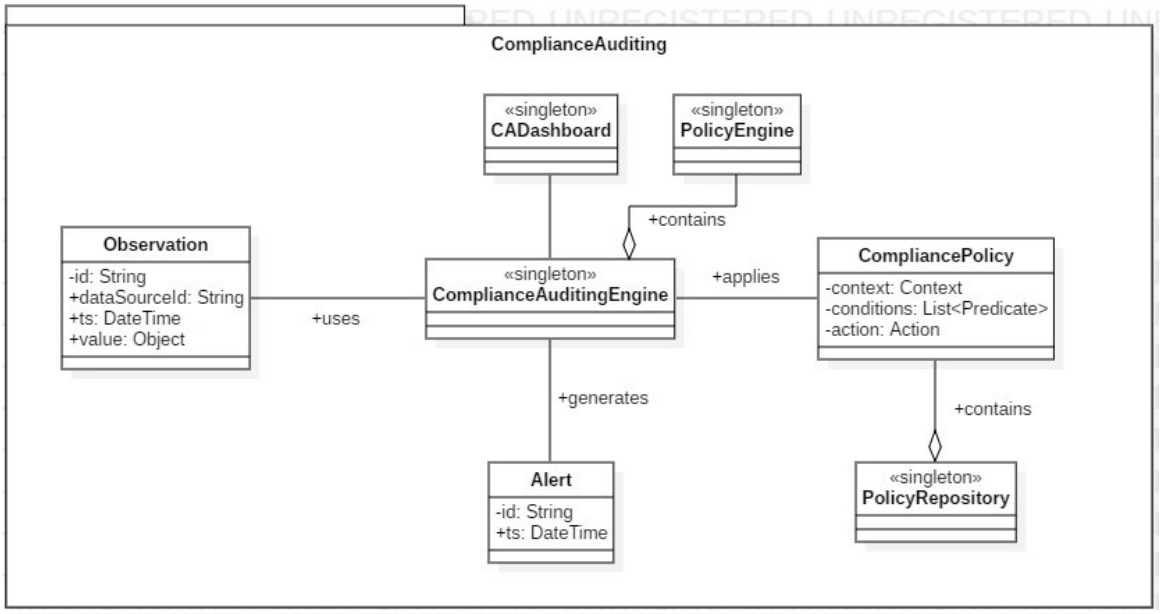


Figure 11: Logical View of the Compliance Auditing service.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

The Developer Support service provides development time support for security services through a set of annotations. The annotations are mapped into policies by the annotation processor that is supported at the programming language level. Even though in the course of the project mostly authorization policies are used, the model shown in Figure 12 is generic enough to allow any other type of policy.

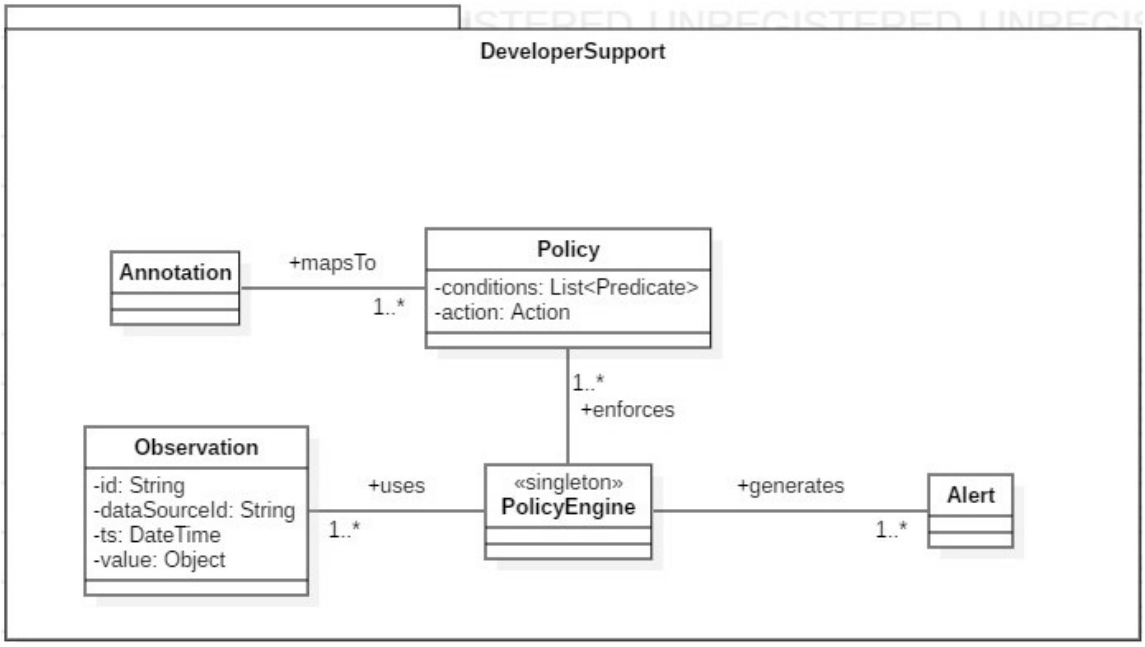


Figure 12: Logical View of the Developer Support service.

A unique characteristic of the SecureIoT platform is that it offers its security services to IoT systems and applications that can span multiple IoT platforms. SLAs (Service Level Agreements) govern the relations between IoT platform providers and the SecureIoT platform. Collected security data from probes deployed on several IoT plaforms and generated alerts are monitored for guaranteeing the compliance to the SLAs. [SecureIoTD3.9] gives the details of the mechanisms used for monitoring the SLAs between platform providers. Figure 13 depitcs the Logical View model for SLA management.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

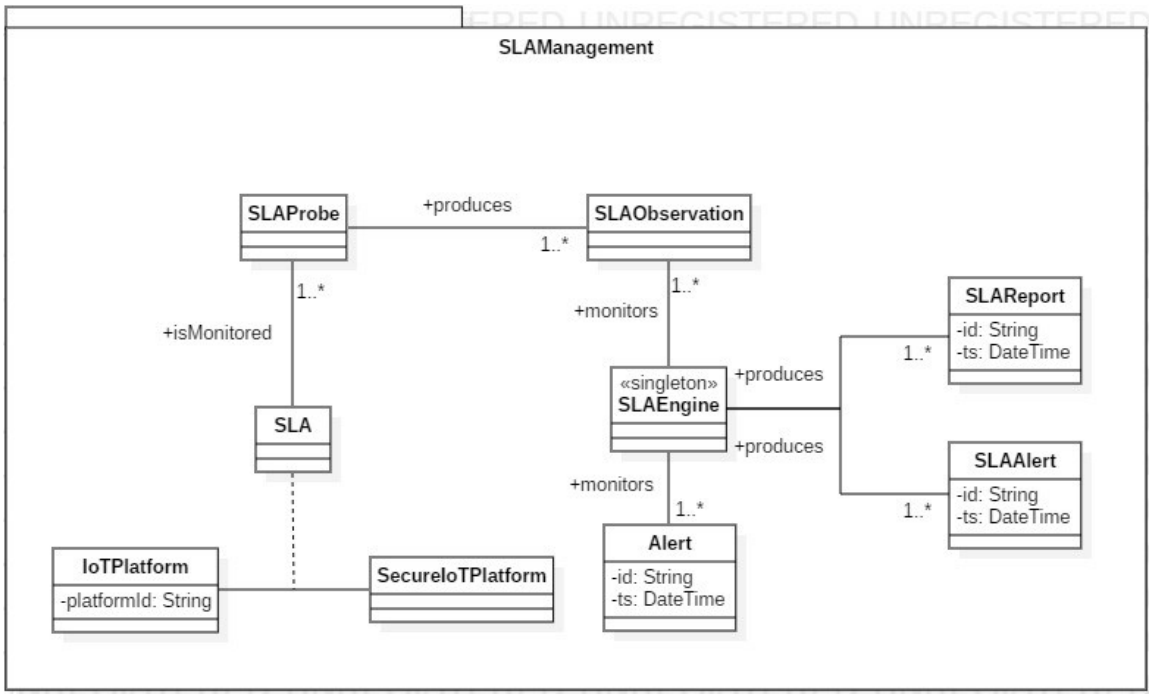


Figure 13: Logical View model of SLA management.

Different IoT platforms on which an IoT application runs may define different policies, for eample for acces, authorization, and so on. To guarantee interoperability between policies, they should be expressed with a common formalism and fixed semantics. Either the same formalism should be used for expressing different policies or, if different formalisms are used they shold be mapped to a common underlying one. Policy interoperability guarantees that policies that have been defined for different platforms can be mapped to a formalism with common and agreed semantics.

2.3 Development View

The development view of the Intelligent Data Collection Architecture focuses on its design level details and reflects the perspective of the developers and implementers. The development view provides for the modelling of components of the software system along with their interfaces. Figure 14 shows a UML component diagram of the SecureIoT architecture. The diagram shows the components of the architecture that support the functionalities presented in the logical view along with their interfaces and dependencies among them. The components shown in the model can be categorized as follows:

- Low level components that implement the tasks of data collection, transport, transform, and storage. These components are implemented by the probes, the streaming infrastructure, components that perform transformations on the collected data for the purpose of normalizing them and finally databases for their storage.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

- Intermediate level components for probe management and configuration as well as the policy enforcement point. These components abstract away from the low level probes and provide a management and configuration layer for them.
- High level components that implement the SECaaS services of Compliance Auditing and Risk Assessment. These components make use of repositories in which security related data are stored.
- Registries like CMDB that contains the assets of the target IoT deployment and a number of repositories that contain security related information, policies and collected data.
- Management GUI that is used for management and configuration purposes and for presenting alerts that may be generated by other components of the architecture.

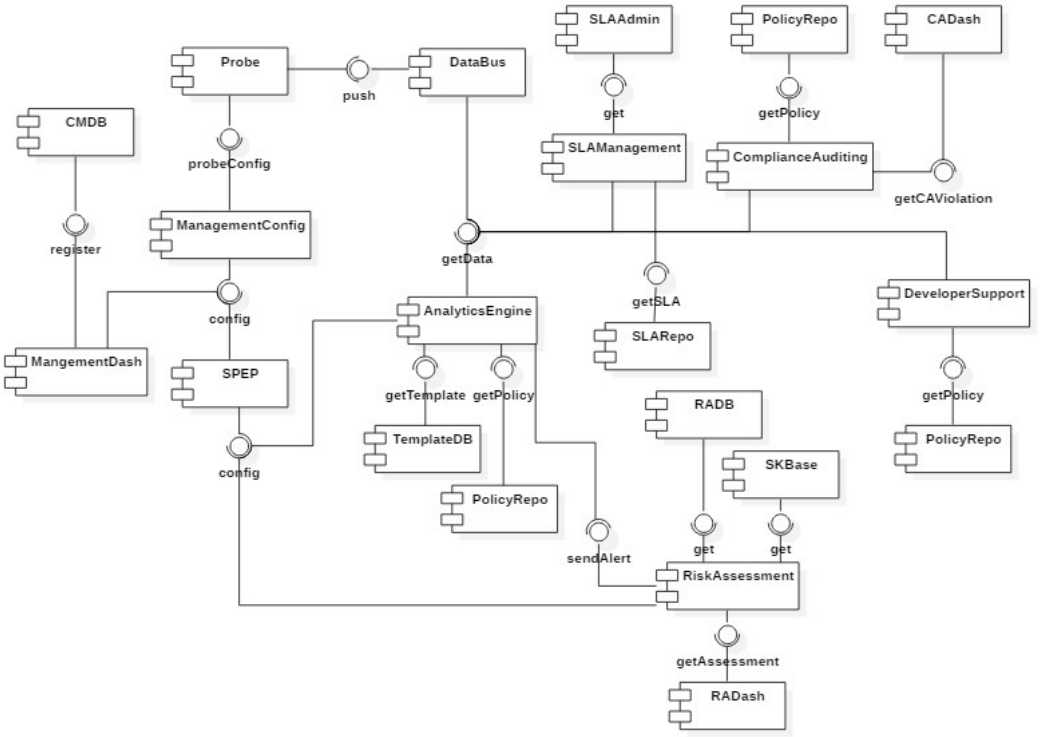


Figure 14: Development View model of the SecureIoT architecture.

Figure 15 shows the same Development View model of the architecture in a more abstract but structured way. It presents the structure of the main SecureIoT modules to be developed in a layered fashion, clustering together modules from the field (i.e. probes), edge (i.e. data routing, probes registry, policy enforcement points), cloud (i.e. template extraction, template execution, contextualization, global storage) and service (i.e. compliance, risk assessment, developers support) tiers. Moreover, cross-cutting functionalities (like management, configuration and visualization) are also depicted. The diagram explains also the dependencies between some modules through identifying modules that use other modules. This "usage" dependency is

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

outlined across layers and tiers (e.g., each tier/layer uses components and services from its underlying layer), but also with a layer (e.g., in the edge tier/layer the data routing uses the IoT probes registry).

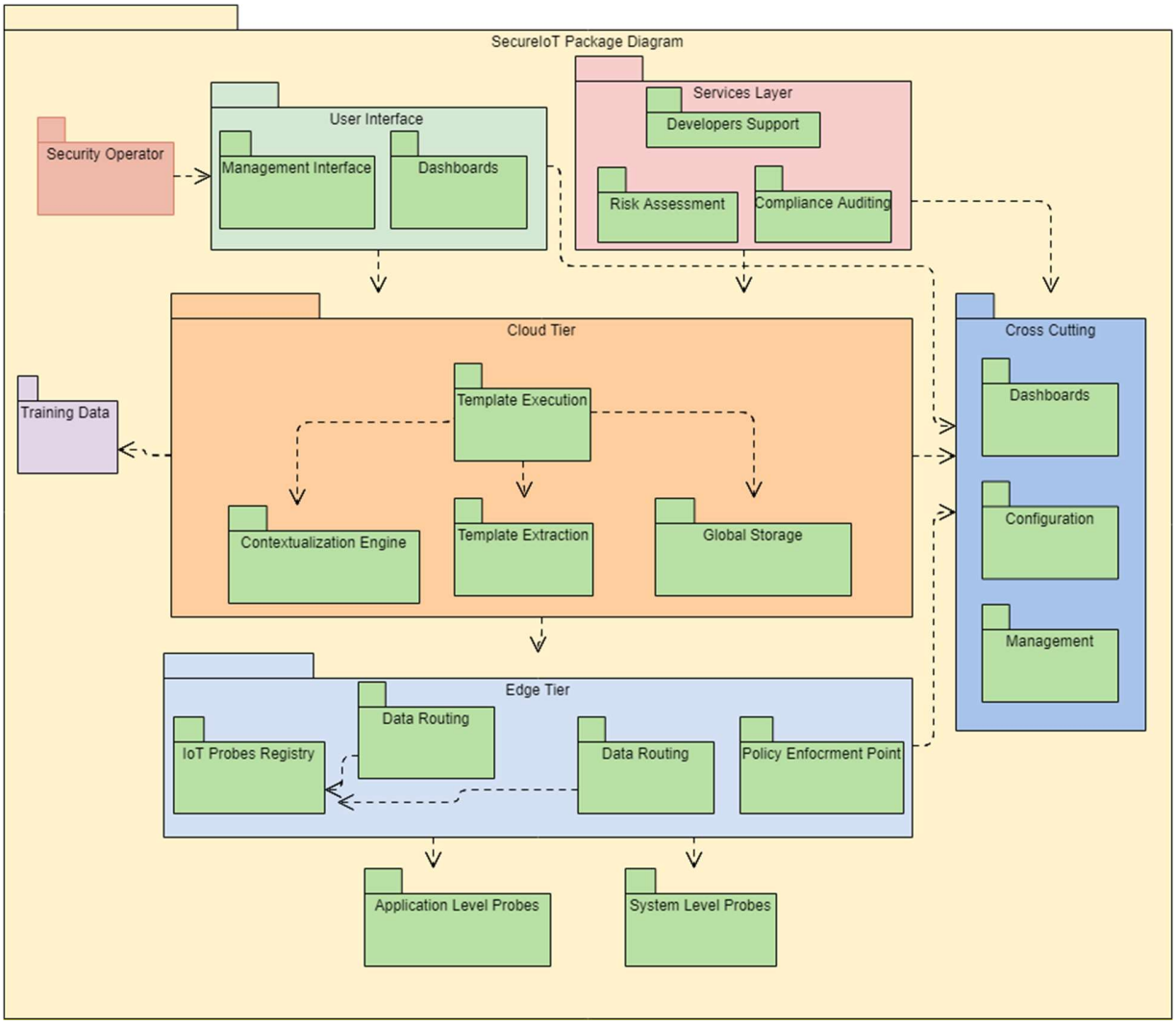


Figure 15: High-Level Development View of SecureIoT architecture.

2.4 Process View

The process view focuses on the dynamic aspects and behaviour of the software system and reflects the perspective of the system integrators. Figure 16 shows a process model for the integration and deployment of the SecureIoT platform.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

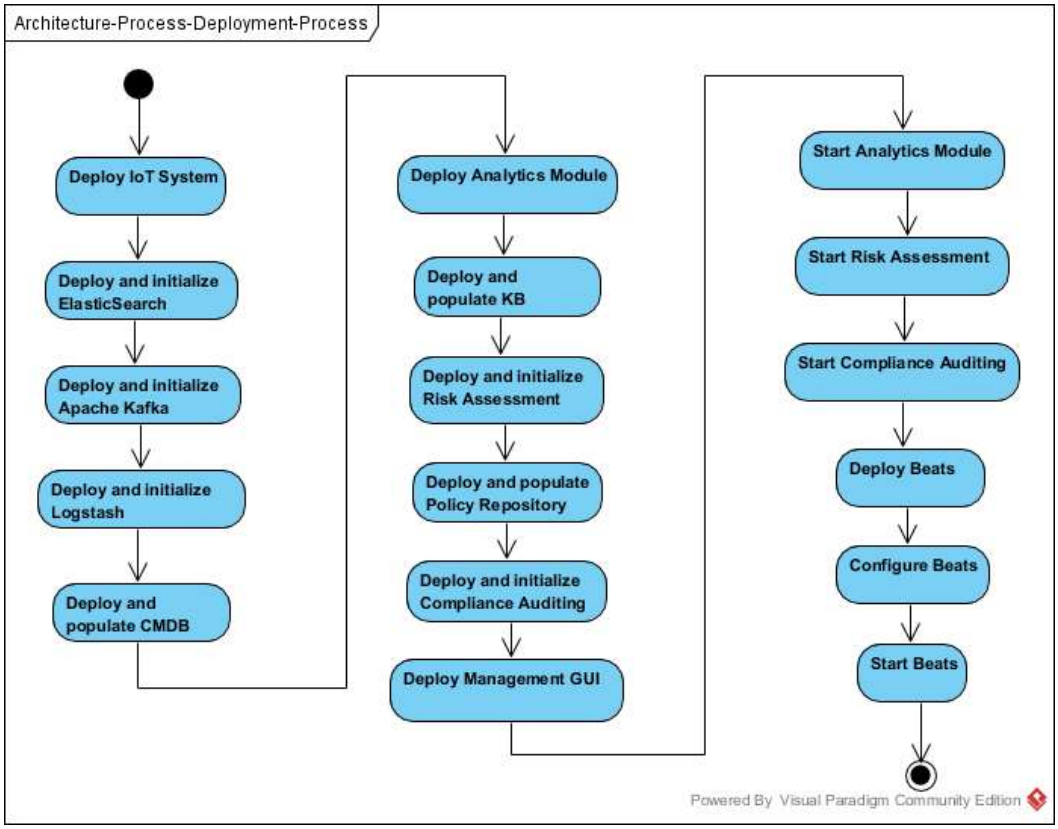


Figure 16: Deployment Process Model.

Figure 17 is a model of the interactions that take place for the creation of a probe its registration and its attachment to an IoT item. The same diagram models the start of the probe as well as the collection, transfer and storage of data.

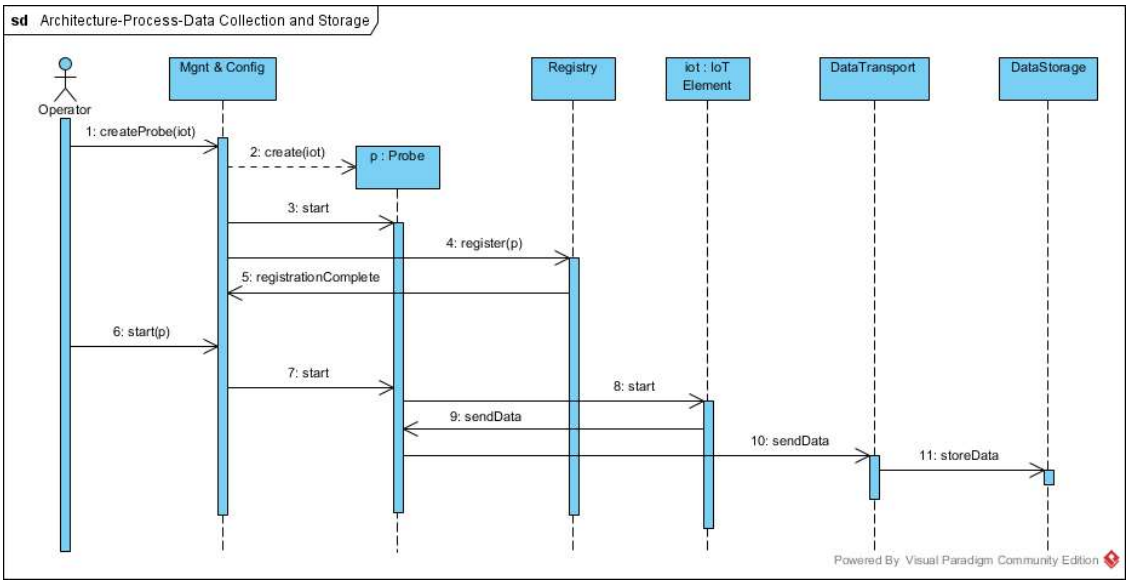


Figure 17: Probe creation, registration, start, and data collection and storage.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

Figure 18 is a model of the interactions between components in the scenario the Analytics module has detected a security breach. In this scenario the Analytics module will send an alert message to SPEP to initiate reconfiguration of one or more probes in an attempt to mitigate the effects of the breach.

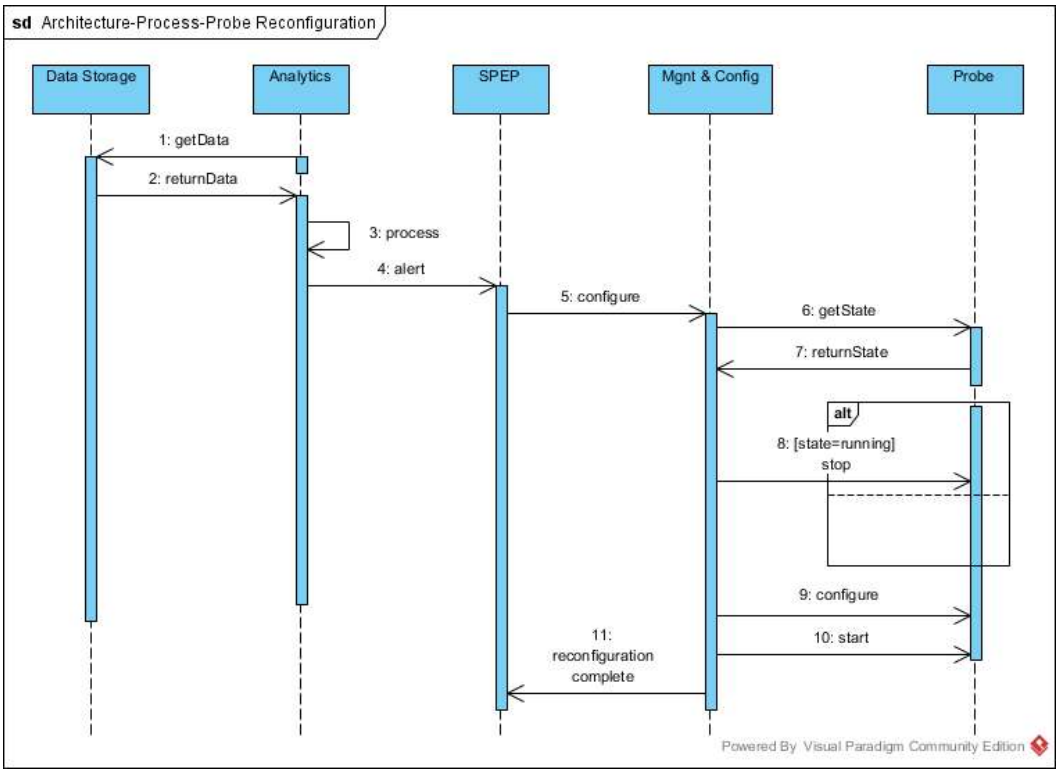


Figure 18: Probe reconfiguration.

The Process View model that is shown in Figure 19 shows the interactions between components of the architecture for the provision of the Risk Assessment service. The corresponding component is initialized based on the vulnerability metrics that it extracts from the Security Knowledge Base. An abstract view of what appears as the Knowledge Base in the model is assumed as the KB may be also linked to external sources of vulnerability data. After initialization the Risk Assessment module keeps monitoring the collected data from the deployed probes that are stored in the Data Store and processes them. If a risk is identified based on the set of vulnerabilities that are checked an alarm is raised and transmitted to the Management GUI for further actions.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

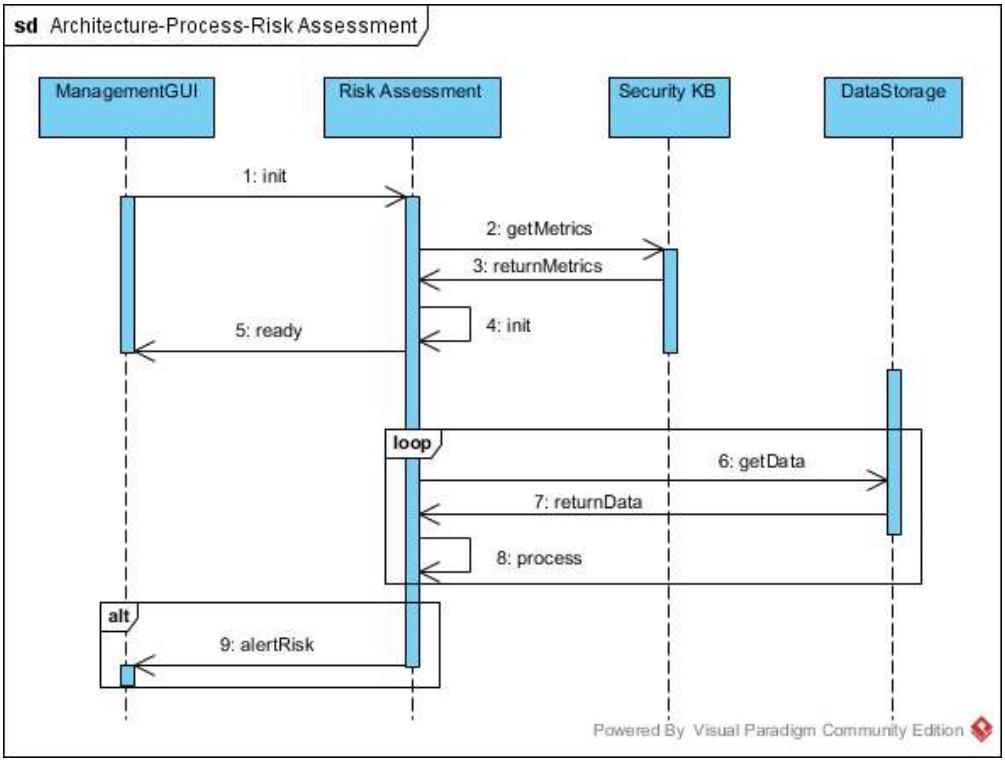


Figure 19: Risk Assessment service provision.

Similarly, the Process View model of Figure 20 shows the interactions between components of the architecture for the provision of the Compliance Auditing service. After the initialization of the component that implements the service, data collected from the probes are retrieved from the Data Storage, they are processed and if a compliance issue is detected, a corresponding alarm is raised and sent to the Management GUI.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

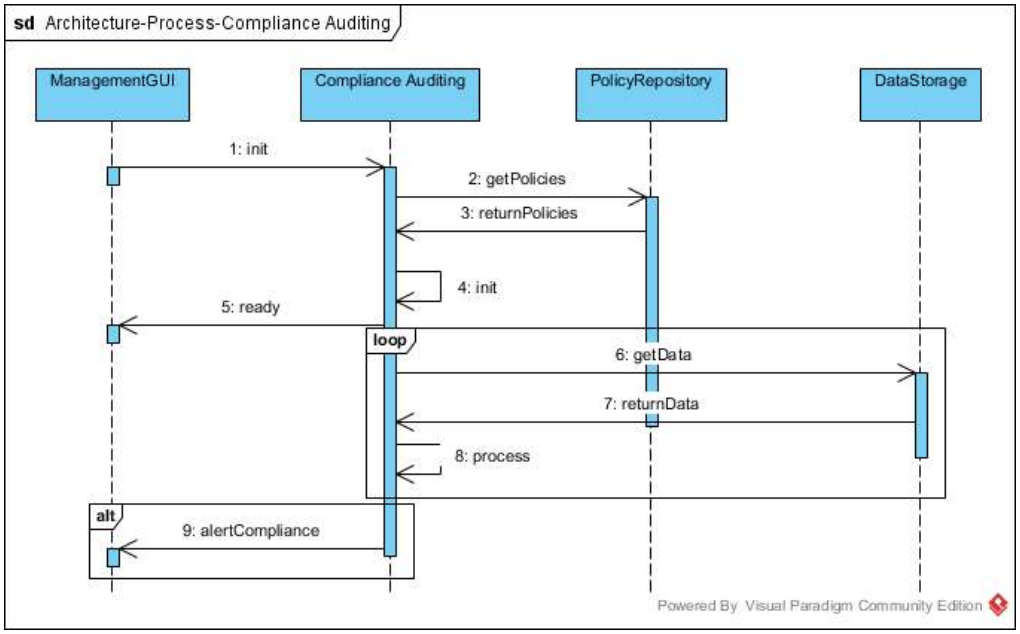


Figure 20: Compliance Auditing service provision.

The sequence diagrams of Figure 21 and of Figure 22 depict the dynamic behaviours of the data collection and of the template execution processes presented as part of the logical view of the architecture. In particular, the security data collection sequence involves the dynamic registration of platforms and probes within an IoT probes registry (called Common Interoperability Registry in this context) as a means of enabling the routing of information at the edge tier and its subsequent storage both a local/edge level and at global/cloud levels.

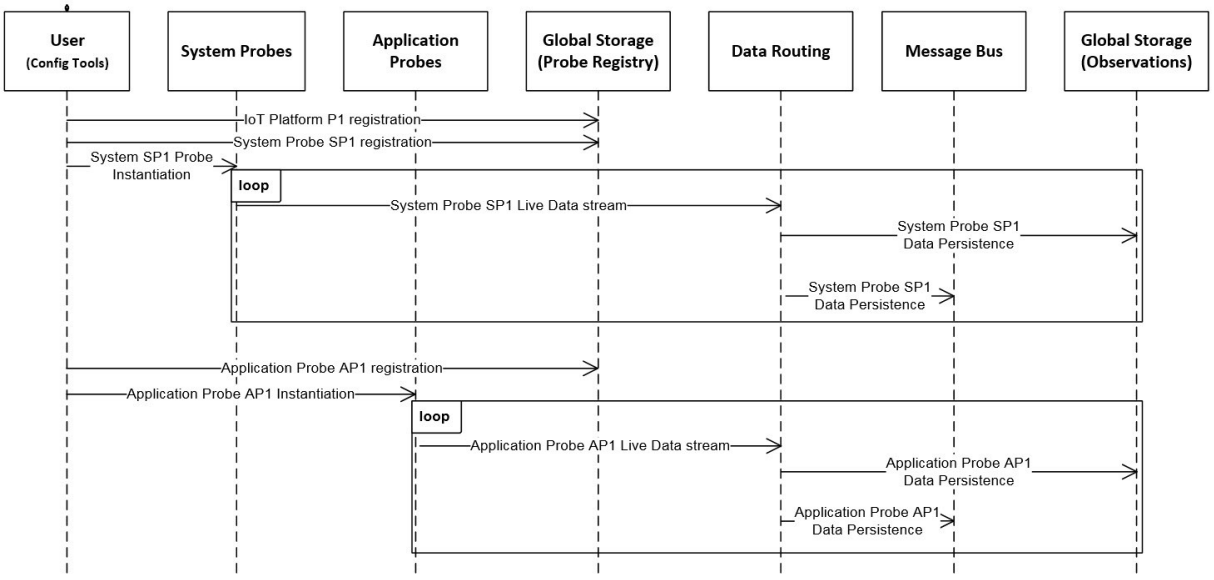


Figure 21: Platform and Probes Registration and Data Collection.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

Likewise, the template execution process is based on templates available within a database of templates, following the process of identifying patterns and extracting templates. The execution leverage data from the global storage. A last step in the execution process involves the matching of identified threats/incidents to knowledge available in the SKB component.

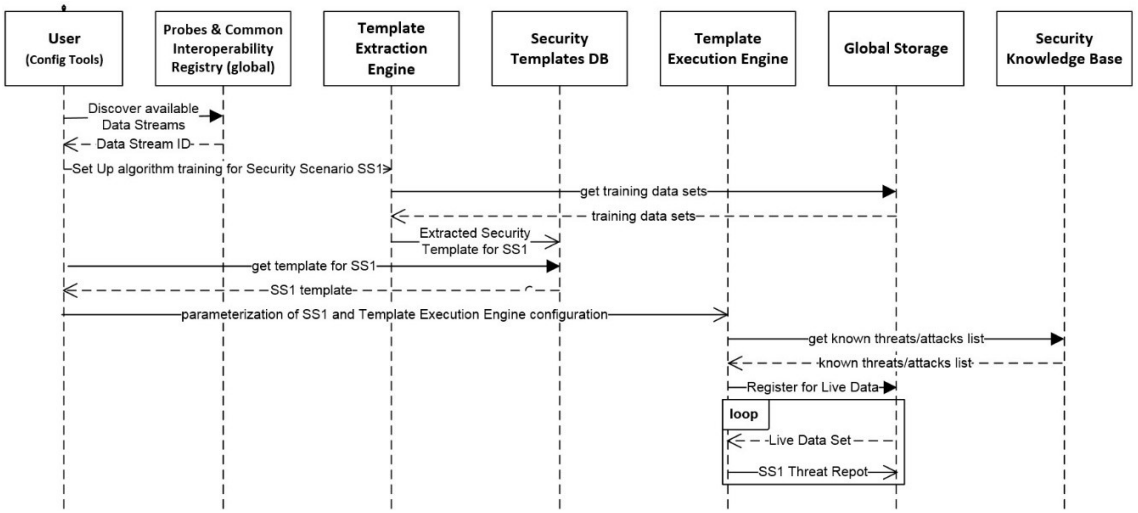


Figure 22: Template Extraction and Execution Process.

2.5 Physical View

The physical view focuses on the topology of components and their connections and reflects the system engineer's perspective. Figure 23 is the deployment diagram for the components of the SecureIoT architecture. The components correspond to the ones of the Development View model of Figure 14.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

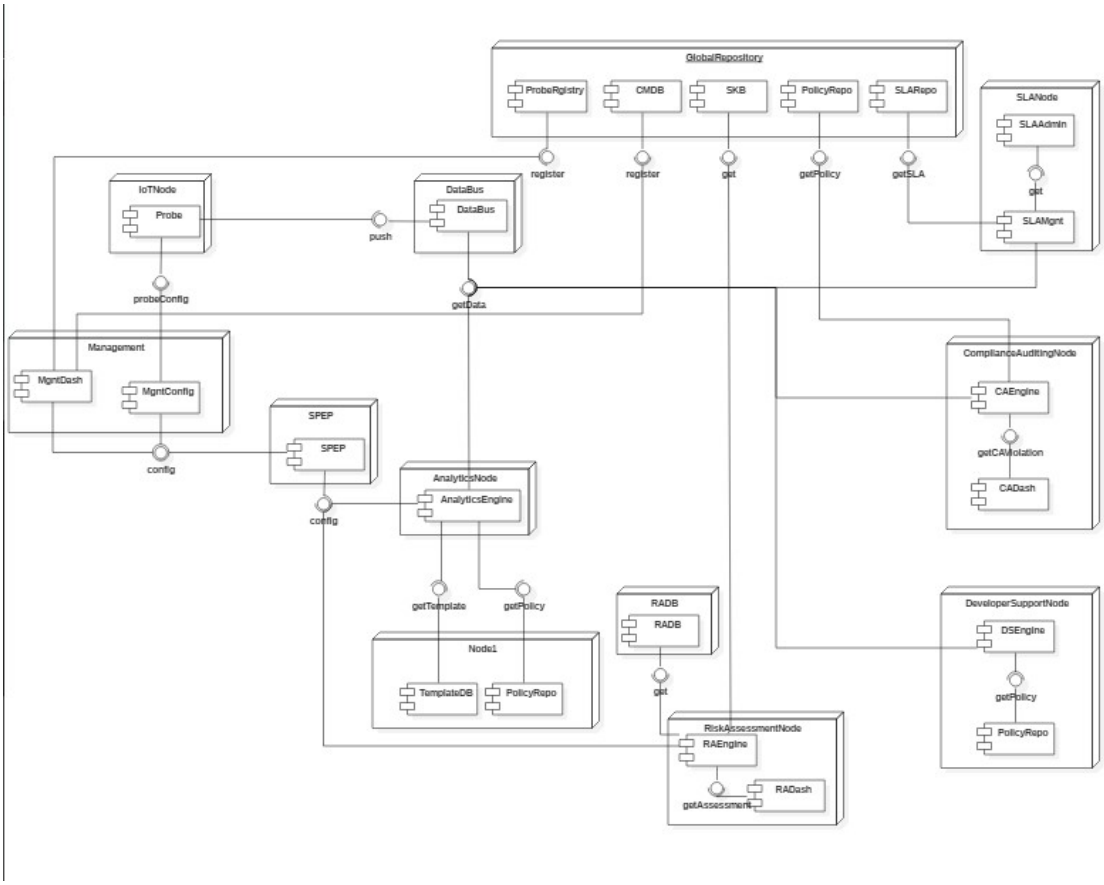


Figure 23: Physical View model of the SecureIoT architecture.

A high-level physical view of a security system that is compliant to the SecureIoT architecture is illustrated in Figure 24. This figure is a deployment diagram, which clarifies aspects of the physical configuration of the SecureIoT system. Multiple probes from different platforms and devices are deployed and used in order to collect information from probes of different systems and devices, notably in cases where these systems are characterized by geographical or administrative distribution. Information is routed from the edge gateways to the cloud, where it is stored in a global datastore i.e. datastore comprising and combining information from multiple edge gateways. The figure illustrates also some of the main components that are deployed in the cloud tier, including the security knowledge base.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

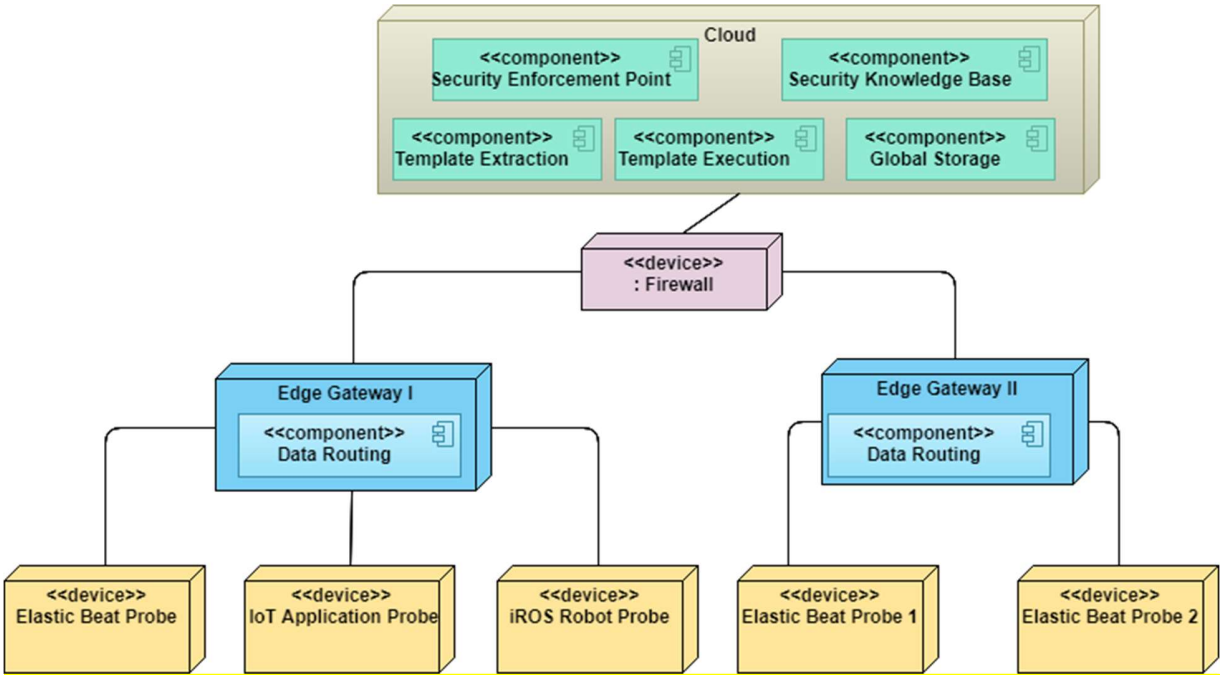


Figure 24: High-Level SecureIoT Deployment Diagram.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

3 SecureIoT Security Services Specifications

SecureIoT Security Services (SECaaS) leverage the information gathered by the *Intelligent Data Collection* and the advanced models and results provided by the *Monitoring & Analytics component.* Thus, the specification of the SECaaS services place requirements for the different elements and technologies developed within WP3 and WP4, which are being implemented to enable the realization of advanced IoT security functionalities that will be delivered *as a service* to IoT developers, managers, security officers and final users. The following paragraphs provide a description of the SECaaS services of the project, given the SecureIoT architecture as presented in the previous section. The present section only gives the description of the SECaaS services, whereas detailed specifications of the components that implement them are presented in the deliverables of the technical workpackages of the project.

3.1 Risk Assessment and Mitigation

The Risk Assessment & Mitigation Services layer shown in Figure 25 sits on top of the SecureIoT architecture and interacts with all the other SecureIoT layers. The core component of the RA&M service is the Risk Assessment engine which is responsible for collecting processed data from various sources of the SecureIoT underlying infrastructure to successfully assess potential risks and propose mitigation actions. As shown in Figure 25, the main volume of data for risk assessment is driven by information produced from the Data Monitoring and Analytics components. The data are retrieved either by subscribing to the data streams or by accessing to the SecureIoT data storage where the Analytics results are persisted. Moreover, information will be retrieved from the Security Knowledge base where the NIST's Common Vulnerability Scoring System will drive the computation of a vulnerabilities' "likelihood" factor.

The results produced from the RA engine will be presented to the end-user accompanied with possible risk alleviation activities. These activities will include enforcement of proper security policies based on the capabilities specified by the monitored platform and the SecureIoT Policy Enforcement component. As shown in Figure 25, as soon as a mitigation action is specified the RA engine should be capable of sending the command to the exposed interface from the Policy Enforcement component at the Data Management group. Then the SecureIoT Policy Enforcement component will be responsible to execute the chosen action based on the exposed capabilities of the monitored platform.

The specification of the Risk Assessment and Mitigation service as well as the APIs of the components that implement it are presented in [SecureIoTD5.1]

Data collection

The Risk Assessment and Mitigation service (RAM) collects data from a number of sources of the SecureIoT underlying infrastructure to assess potential risks and propose mitigation actions. Figure 25 shows that the main volume of data that drive the risk assessment comes from the Monitoring and Analytics components. The data are retrieved either by subscribing to the data

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

streams of the streaming service or through the SecureIoT data storage where the Monitoring and Analytics results are stored. Moreover, the Risk Assessment service uses information from the Security Knowledge base where the NIST's Common Vulnerability Scoring System is stored that drives the computation of vulnerabilities' "likelihood" factor.

Data Analytics

As shown in Figure 25 the RAM services make use of information from the Monitoring and Analytics component. The Risk Assessment (RA) service monitors the collected data and makes use of the information stored in the Security Knowledge Base to produce intelligent risk assessment output. The Risk Assessment service calculates different configurations and risk criticality values to output its risk assessment.

Security Policy Enforcement Points

The results produced from the RA component are presented to service operator's management console and accompanied with risk alleviation tasks like enforcement of proper security policies based on the capabilities specified by the monitored platform and the SecureIoT PEP. As shown in Figure 25, as soon as a mitigation action is specified the RA engine sends the reconfiguration commands to the PEP at the Data Management group, which enforces the action to the components of the monitoring platform.

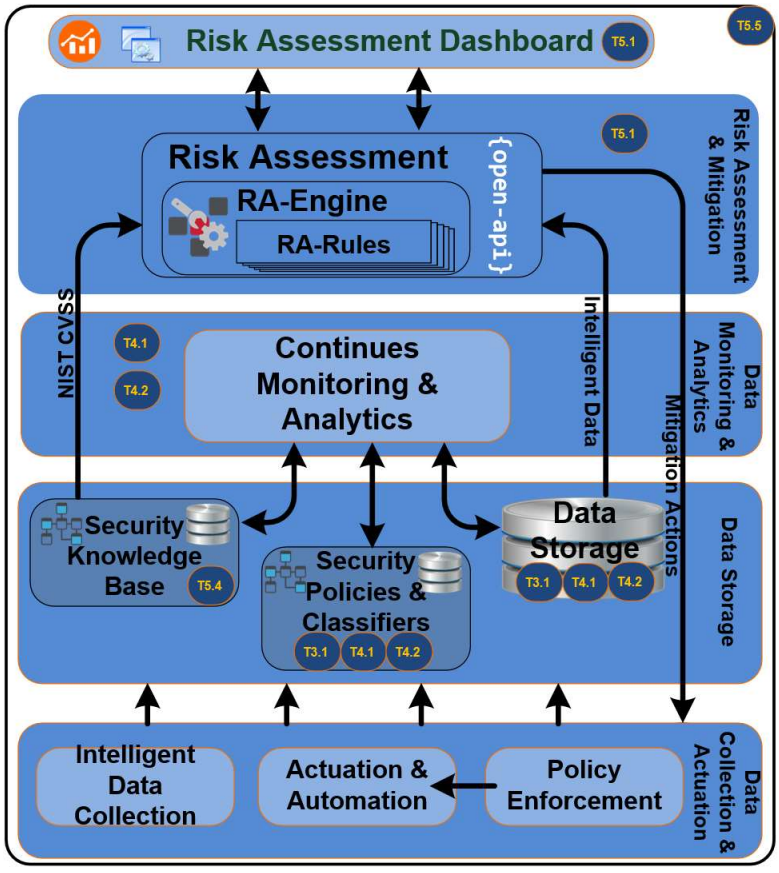


Figure 25: Risk Assessment & Mitigation Interactions.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

Cross Platform Data Exchange

A unique characteristic of SecureIoT is the support of security services to IoT applications that may be deployed to multiple IoT platforms. The RAM service supports such applications by processing security data that are collected from probes that may be deployed into a number of IoT platforms and outputting risk assessment metrics.

3.2 Compliance Auditing

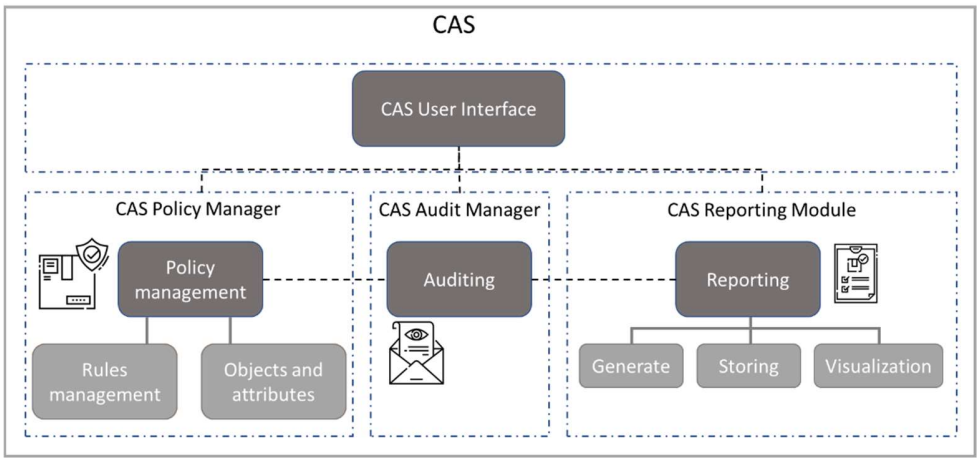


Figure 26: CAS Components.

The CAS is integrated into the SecureIoT architecture by means of different components that interact for storing data, reporting, accessing data of the target system via interfaces, etc. Figure 26 shows the initial composition of the CAS and the integration/interaction with other components of the architecture. The CAS is composed of, from a logical point of view (which also is described in the following sections), four different components, each of them with a specific objective. Following we describe each of the components and their relationship with the elements of the architecture.

- User Interface: This component is a front-end to be used by the end-users for auditing the compliance of their systems and obtaining a report of the audit. The architecture of SecureIoT didn't require a specific way to provide the functionality of the CAS but we wanted to make it available in as many ways as possible. Therefore, we are planning to provide it either with an interface or an API for being integrated with existing or new systems.
- Policy Management: One feature of the user interface is providing the policy modelling tool for authoring, test and deployment of policies, rules and objects.
- Auditing and Compliance: The Auditing and Compliance module is provided by the user interface in the first iteration. This element is the main functionality to be satisfied in the CAS. The main objective of this component is to provide an auditing and compliance module that can be used to satisfy the needs of the end-users for their IoT devices. Therefore, this module provides the end-points of the functionality, which is then

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

provided to the end-users via the user interface as it is described in the previous component.

- Reporting: The reporting module provides the audit information to the service user via the user interface. The report shall include information on the rule, object and auditing result. It will be stored database. The XML file may be transformed by the user according to its needs and specifications.
- Data Storage and Data Collection: According to the needs of the policies for assurance and compliance, the system must be able to assess data of the target system and decide on its compliance to policies and the related rules, which are defined in the user interface. This is a critical component of the CAS and, therefore, it will be integrated with the CMDB of the SecureIoT pilots. In the subsequent implementations the integration with the components of data collection, in order to obtain further data and use it for the analysis of the compliance auditing, will be assessed as the improved maturity of those modules will facilitate the integration.
- Moreover, the CAS requires to store and use different data elements: the policies for assurance and compliance to be used in the target system, the information obtained from the analysis and the report to be generated. Therefore, the CAS shall utilize an integrated database in the first implementation. In a later iteration of the CAS the integration with the future databases for policy storage, aka Policy Repository Point (PRP), will be assessed. The implementation of a PRP will facilitate the management of all the information of the CAS centralized in the architecture of SecureIoT and will facilitate the exchange of data with other components. Additionally, the reports may be provided either directly to the end-users via API or as raw data or in a component for showing the requested analysis and, if possible, historical data of past ones.

The specification of the Compliance Auditing service as well as the APIs of the components that implement it are presented in [SecureIoTD5.4]

Data collection

Compliance Auditing services also rely on collected security data. The service assesses compliance of an IoT application to standards and directives according to the set requirements. The data required for assessing compliance of the IoT system to be audited for adherence to specific requirements are gathered using an asset list, e.g. a CMDB, and corresponding controls catalogue. Moreover, data collected by this service enable the monitoring module for auditing the IoT system.

Data Analytics

Data Analytics may support a continuous auditing model. In this case, which WP5 will assess, the data of the IoT system shall be analysed continuously by the Data Analytics module facilitating detection of a breach on conformity.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

Security Policy Enforcement Points Specification

Within this module the relevant compliance policies of the IoT system reside. The policies facilitate evaluation of conformity the IoT system must adhere to. Moreover, this module ensures conformity based on data coming from monitoring by the Data Collection module and insights gained using the Data Analytics engine.

3.3 Developer Support Services

The Developer Support service supports programming of secure IoT applications that may enforce policies according to programming language level directives. The directives are expressed as programming annotations and trigger security functions during program execution. These services also rely on collected security data and tasks that are to be enforced according to specified policies. The specification of the Developer Support service as well as the APIs of the components that implement it are presented in [SecureIoTD5.7]

Data collection

Data Collection is used as a context collector to add further conditions for access control policies such as type of device. During data collection, the PIP retrieves the necessary attributes for the policy evaluation from several security probes and communicates with the PDP to trigger a decision.

Data Analytics

Data Analytics is used to further analyse the collected context and add combined conditions for access control policies such as type of device, location and environment information. During data analytics, the PDP collects all the necessary information, yields the decision and passes it to the PEP at the edge/cloud level.

The Security Intelligence layer makes use of the Observations component of the Global Repository that contains historic data and uses them for its training. The templates that result from the training activity are stored in the Security Template Database and are used by the Analytics Engine for monitoring the stream of security data collected by the deployed probes that are communicated through the data bus. The alerts that may be generated are used for enforcing security policies to the deployed probes.

Security Policy Enforcement Points

The Security Policy Enforcement Points receives application specific access requests and translates them to XACML[1] access control requests that result into denial or allowance of resource accesses. Data collection and analytics functionalities may be used as an additional context for access control purposes. Developer support services provide parametric contextual information (e.g., location, device type, time of interaction) of an HTTP request that is used for policy enforcement. Based on the collected data and the decision taken on the PDP, the security PEP enforces the necessary policies.

[1] https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

The Security Intelligence Layer makes use of a Security Templates Database to store the data of security templates of the XACML policies and the TEE to execute polices. The SPEP of access control is defined for both Data Collection and Actuation Layer and Security Intelligence Layer to enforce the security policies.

Cross Layer Data Exchange

Context-aware access control is extensible and is based on the context independently of any IoT layer. Open APIs between the layers are used for cross-layer data exchange. Management and Configuration Tools are used for managing these elements whereas Visualization (Dashboards) tools are used for presenting the status of the data collection and actuation layer including cross-layer data exchanges.

Cross Platform Data Exchange

Developer support services can be extended to be applied to multiple IoT platforms. The same directives and mechanisms that collect security data and apply policies as directed by the programming annotations can be used to collect data from multiple IoT platforms and apply security policies to components that are supported by them.

3.4 IoT Security Knowledge Base

The SECaaS services of SecureIoT rely on an IoT Security Knowledge Base (SKB) that stores structured information on threats including, but not limited to CPE, CWE, CVE and CAPEC specifications. The CVE (Common Vulnerability and Exposure) data source contains publicly known cybersecurity vulnerabilities. The CWE (Common Weakness Enumeration) contains a list of common software vulnerabilities and CAPEC (Common Attack Pattern Enumeration and Classification) contains a dictionary and classification taxonomy of known attacks. These information sources are known collectively as Cyber Threat Intelligence (CTI).

An overview of the components of the SKB is shown in Figure 27. The CTI collector is responsible on collecting and storing external CTI data sources and provides an API endpoint to store data provided by other components of SECaaS in the CTI database. The correlator is responsible on correlating the available information to generate knowledge graphs where vertices are assets, vulnerabilities, weakness, or attack patterns and edges their relations. The graph computing base is responsible on providing a query layer for the available graphs to extract from them security properties of available assets.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

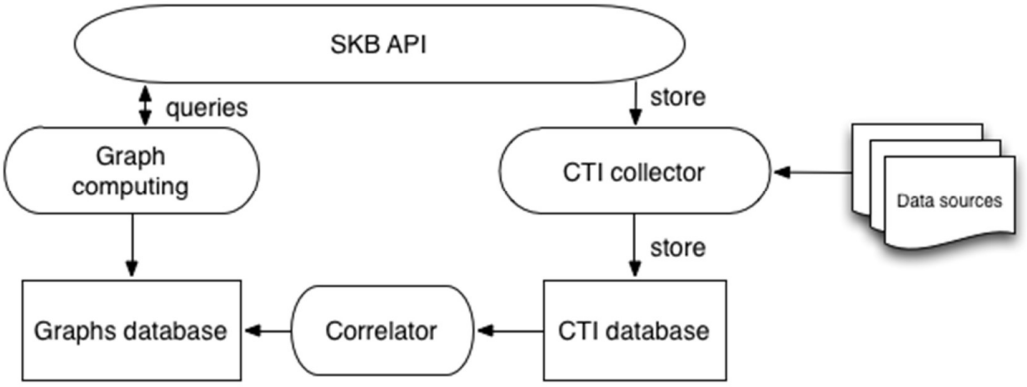


Figure 27: The components of the IoT Security knowledge base.

In the following the interaction of this security knowledge base according to the different components and layers of the SECaaS architecture is specified.

The classes of CTI database are depicted in Figure 28. The CVE class contains the attributes of a CVE document including its label, for example CVE-2018-13074. The CVSS attribute denotes the cvss score of the cve. The summary is a brief summary of the cve. ExploitDBid is the id of the exploit-db module if it exists. Nessus is list of the id of the different nessus module. Access contains information about the exploit of a vulnerability: (a) authentication that indicates if it needs authentication to exploit this vulnerability, (b) complexity that indicates the level of complexity of the exploit of this vulnerability, (c) vector that indicates the vector of the attack for this vulnerability. Each CVE is linked to one or many CWE and CAPEC and it is related to one or many CPE which identifies the vulnerable configuration.

The CWE class contains a label which is the id of the cwe, a name, a description and the list of the main consequences after the attack. A CWE is linked to one or many CVE and one or many CAPEC. It is also linked to one or many CWE.

The CAPEC class contains a label which is the id of the capec, a name, a description, the list of attack prerequisites and the required resources. A CAPEC is related to one or many CVE, and one or many CWE. It is linked to one or many CAPEC.

The CPE class contains only a label which is the identifier of the cpe. It is only related to one or many CVE.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

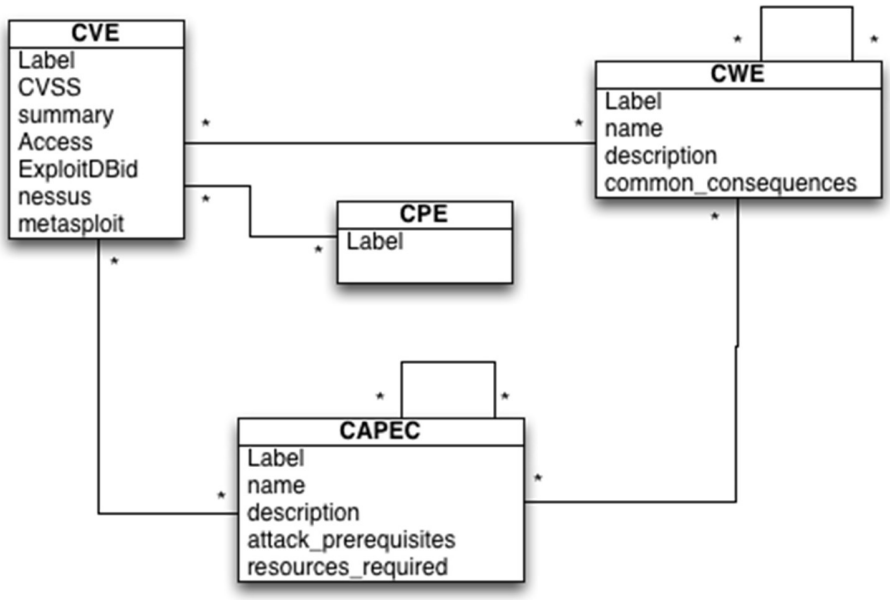


Figure 28: The classes of the CTI database and their relationships.

3.5 SECaaS Services Management and Configuration

The Continuous Integration (CI) and Configuration of SECaaS services provides an approach, an environment, tools and automated mechanisms to warranty an agile, reliable and robust development of the SECaaS, following modern agile and DevOps practices. Thus, it is a horizontal task that supports all the development across the technical workpackages of the project. The requirements established to the different components of SecureIoT are the following:

- Use a Source Code Management (SCM) tool during the whole development lifecycle, e.g., GitLab or GitHub.
  - Use issues to manage the development of new features and the fix of bugs.
  - Rely on the SCM for version control and code repository.
  - Develop the new code in separate branches. Use *Pull Request (PR) or Merge Request (MR)* mechanisms to integrate them in the master or development branches. Commits to master or development branches are not allowed.
  - Perform small commits and small pull or merge requests.
- Continuous Integration and Testing
  - Include automated building using Docker.
  - Include unit tests. Use of external component's APIs should be mocked.
  - Include integration tests using Docker and docker-compose.
  - Integrate automation tools to execute CI tests each time a commit or PR/MR is done.
  - Integrate a code coverage tool. Percentage should be higher than 70-80%.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

- Integrate a tool that detects style errors.

- Continuous Deployment
  - Docker images must be published.
  - Docker-compose must be used to illustrate the configuration, deployment and interconnection of SecureIoT components.
- Documentation
  - Include a README file in the repository with the following minimum content:
    - Description
    - How to build and install
    - API overview and examples
    - API reference documentation using Swagger or a similar tool.
    - How to run tests.
- High-availability
  - Documentation, docker images and docker-compose files must consider how to deploy the component in a high-availability environment, addressing specifically aspects like scalability or elasticity.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

4 SecureIoT Data Models

This section provides a high-level specification of the SecureIoT Data Models. It does not go into in-depth descriptions since these are provided in detail in the WP3, WP4 and WP5 deliverables they are related to. The SecureIoT Data Models form five main logical groups (Figure 29), which are:

- **Data Management**: focuses on the modelling of the data collection and data routing (Figure 31).
- **Analytics**: focuses on the Analytics configuration and management at the Security Intelligence layer (Figure 32).
- **Knowledge Base**: focuses on the Security Knowledge coming from third party standardized sources (Figure 33).
- **Risk Assessment**: focuses on the configuration and management of the Risks Assessment and Mitigations SECaaS service (Figure 34).
- **Configuration and Management DB (CMDB)**: focuses on configuration aspects of the SecureIoT infrastructure mainly modelling information related with the monitored IoT systems and the Attack Scenarios (Figure 35).
- **SLA**: which deals with the data models for configuring and reporting the SLA agreements (Figure 55)

Some of the Logical Groups are focusing on different aspects of the SecureIoT infrastructure and some are shared along the different layers. All together they form an integrated solution, which is used in the SecureIoT Infrastructure.

Figure 29 shows the SecureIoT Root Entity of the data model with the different Logical Groups.

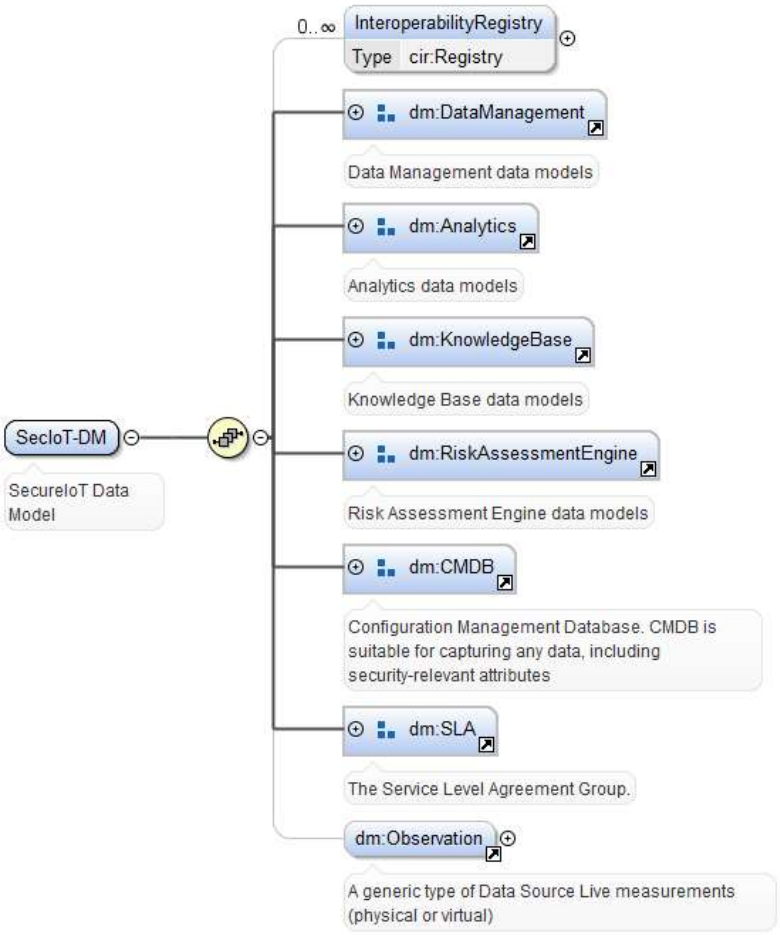


Figure 29: SecureIoT Data Model Logical Groups.

Among the different Logical groups of Figure 29 is the Observation (detailed in Figure 30), which is used as a common format to present data reports within the SecureIoT infrastructure (i.e., at the Data Collection layer or the Security Intelligence layer). Moreover, the Interoperability Registry is modelled, which is capable for enabling the SecureIoT observed data from the field to be enriched with additional Use Case semantics coming from the monitored IoT System third-party databases.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

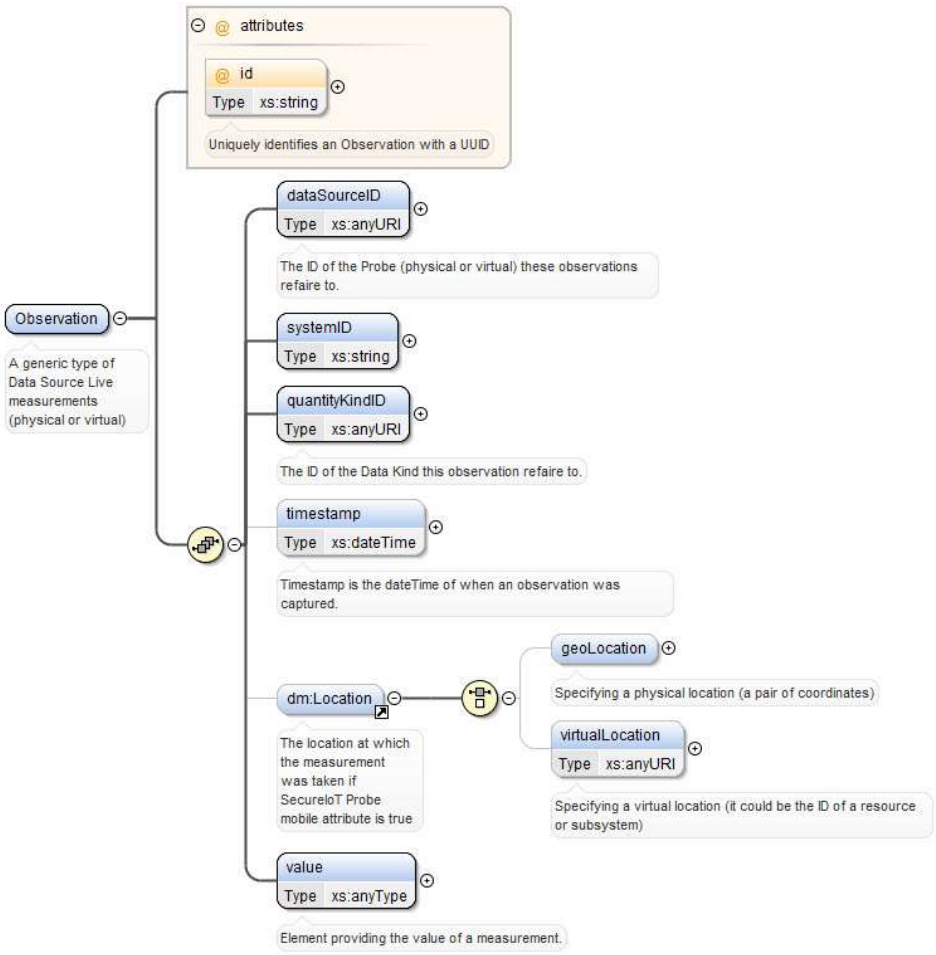


Figure 30: SecureIoT Observation Entity.

Figure 31 shows the Data Management Logical Group with the entities it comprises.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

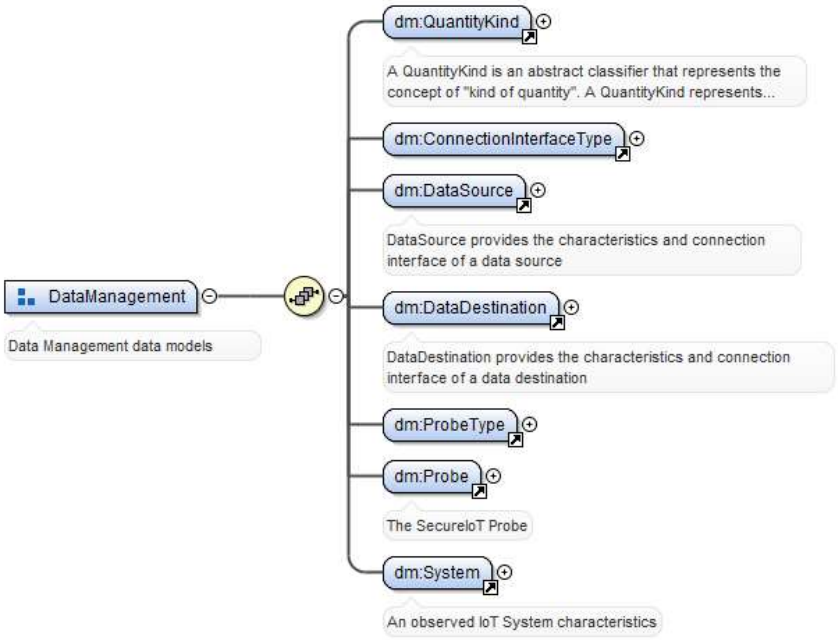


Figure 31: SecureIoT Data Management Logical Group.

Figure 32 shows the Analytics Logical Group with the entities it comprises.

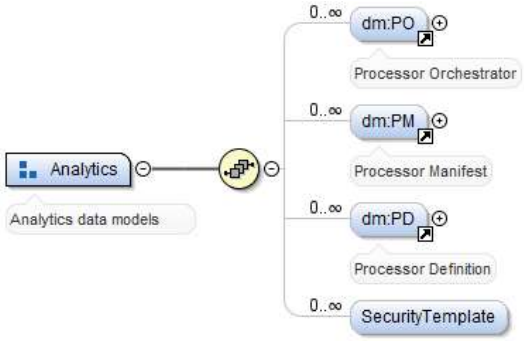


Figure 32: SecureIoT Analytics Logical Group.

Figure 33 shows the Knowledge Base Logical Group with the entities it comprises.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

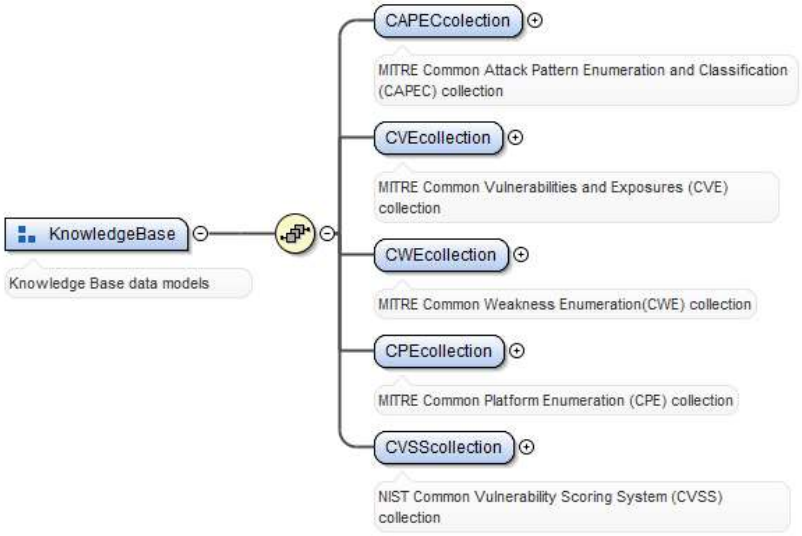


Figure 33: SecureIoT Knowledge Base Logical Group.

Figure 34 shows the Risk Assessment Logical Group with the entities it comprises.

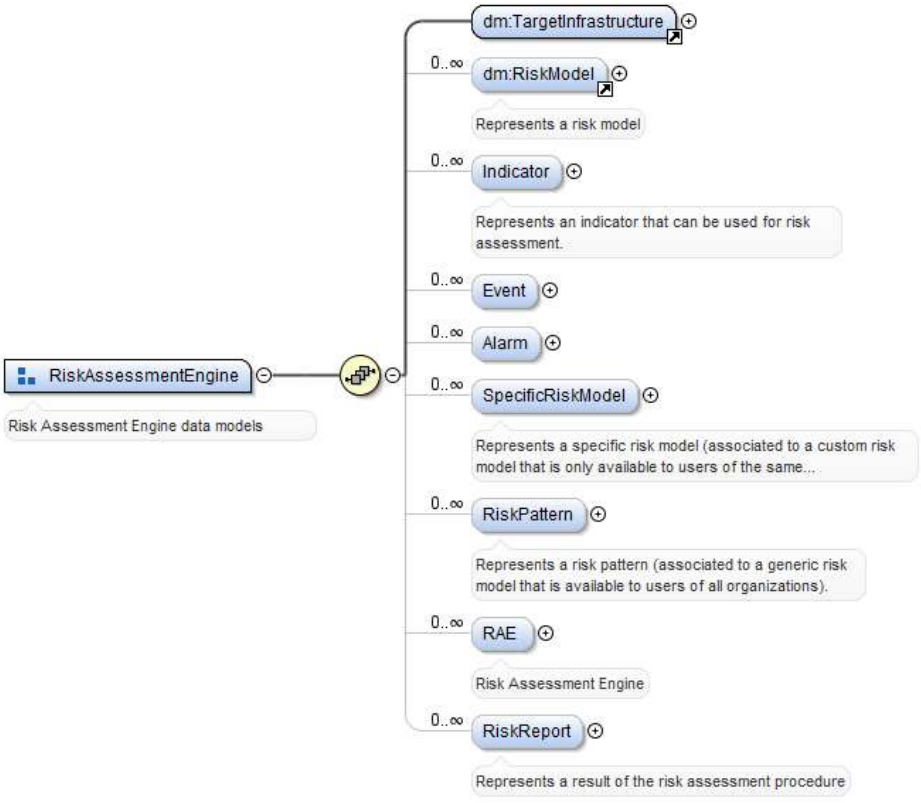


Figure 34: SecureIoT Risk Assessment Engine Logical Group.

Figure 35 shows the Configuration and Management Logical Group with the entities it comprises.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

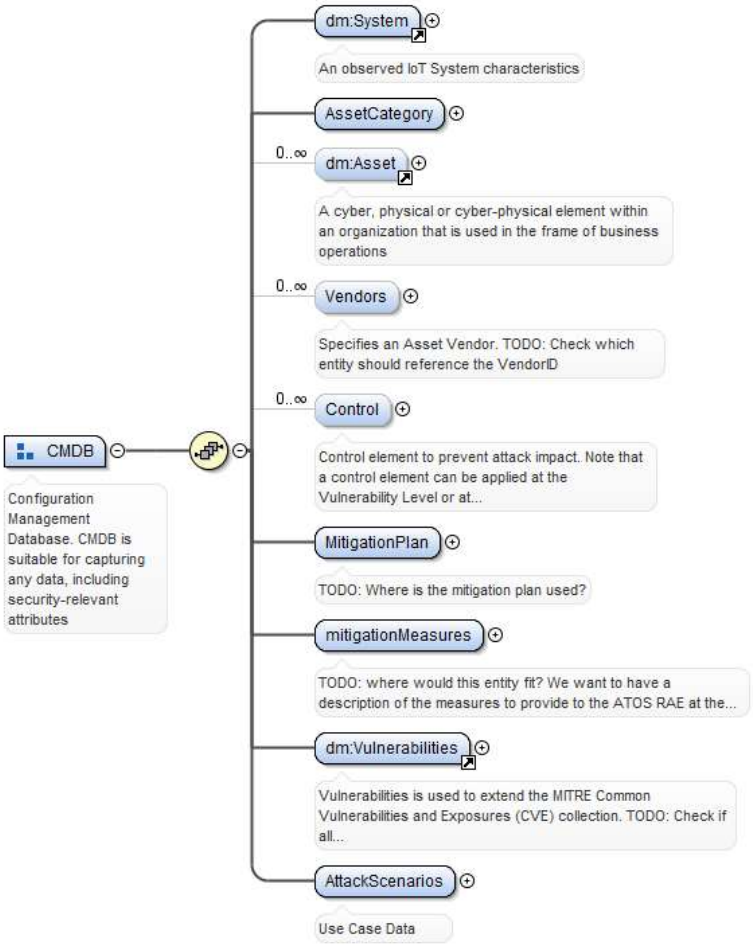


Figure 35: SecureIoT CMDB Logical Group.

The complete SecureIoT Data Model schema is available in the form of an XSD which is provided in Table 2 of Appendix A.

4.1 Data Management

Figure 36 depicts the data model that pertains to data collection. The main entities are the probe, which is responsible for collecting data and streaming it to the processing components of the platform and the Observation, which represents the collected datum. Each Observation concerns a metric that is measured, and this is represented as the QuantityKind in the model. A given probe collects data of a given kind and if different metrics are to be measured then different probes must be deployed. Moreover, each probe is associated to a Data Source from which data originate and a Data Destination to which data are fed.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

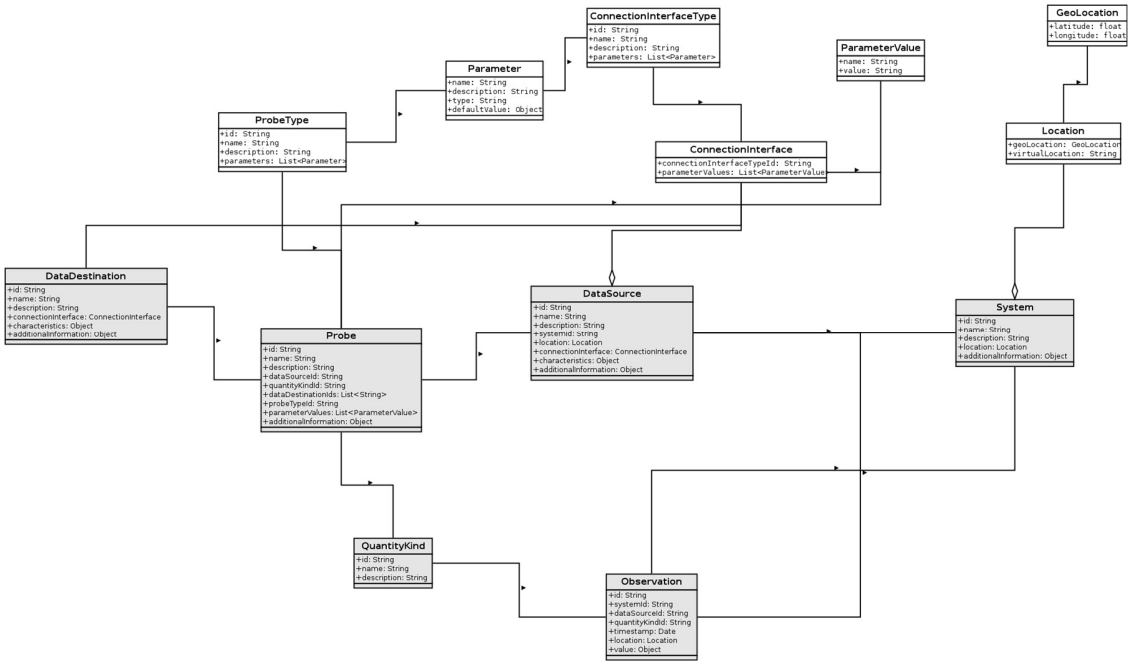


Figure 36: Data Collection Data Model.

The data management group focuses on the data collection and data routing of the SecureIoT platform and contains the following core entities.

Quantity Kind: QuantityKind is an abstract classifier that represents the concept of "kind of quantity" (Figure 37). A QuantityKind represents the essence of a quantity without any numerical value or unit (e.g., if sensor s1 measures temperatures then s1 has quantityKind temperature).

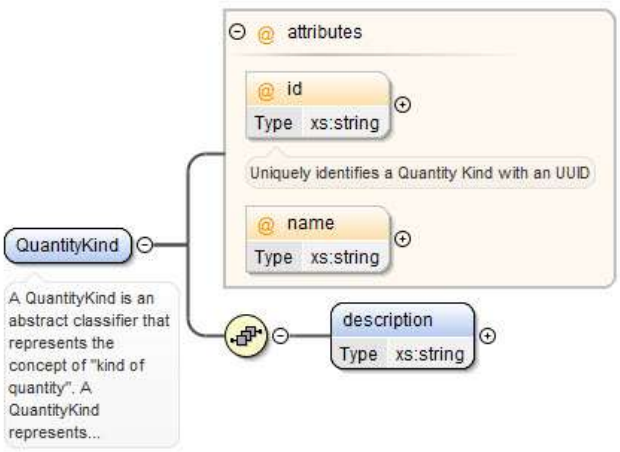


Figure 37: SecureIoT Quantity Kind entity.

Connection Interface Type: models the type of connection interface which is used at the Data Source and Data Destination entities (Figure 38).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

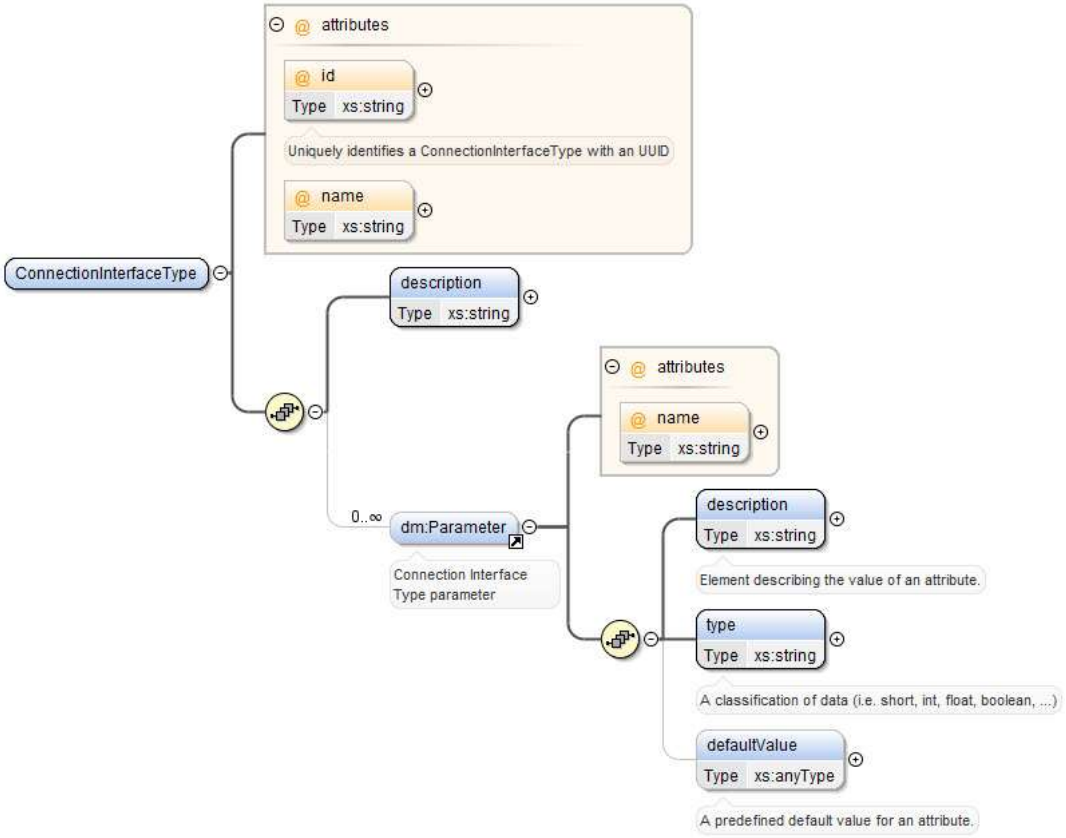


Figure 38: SecureIoT Connection Interface Type entity.

Data Source: DataSource provides the characteristics and connection interface of a data source (Figure 39).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

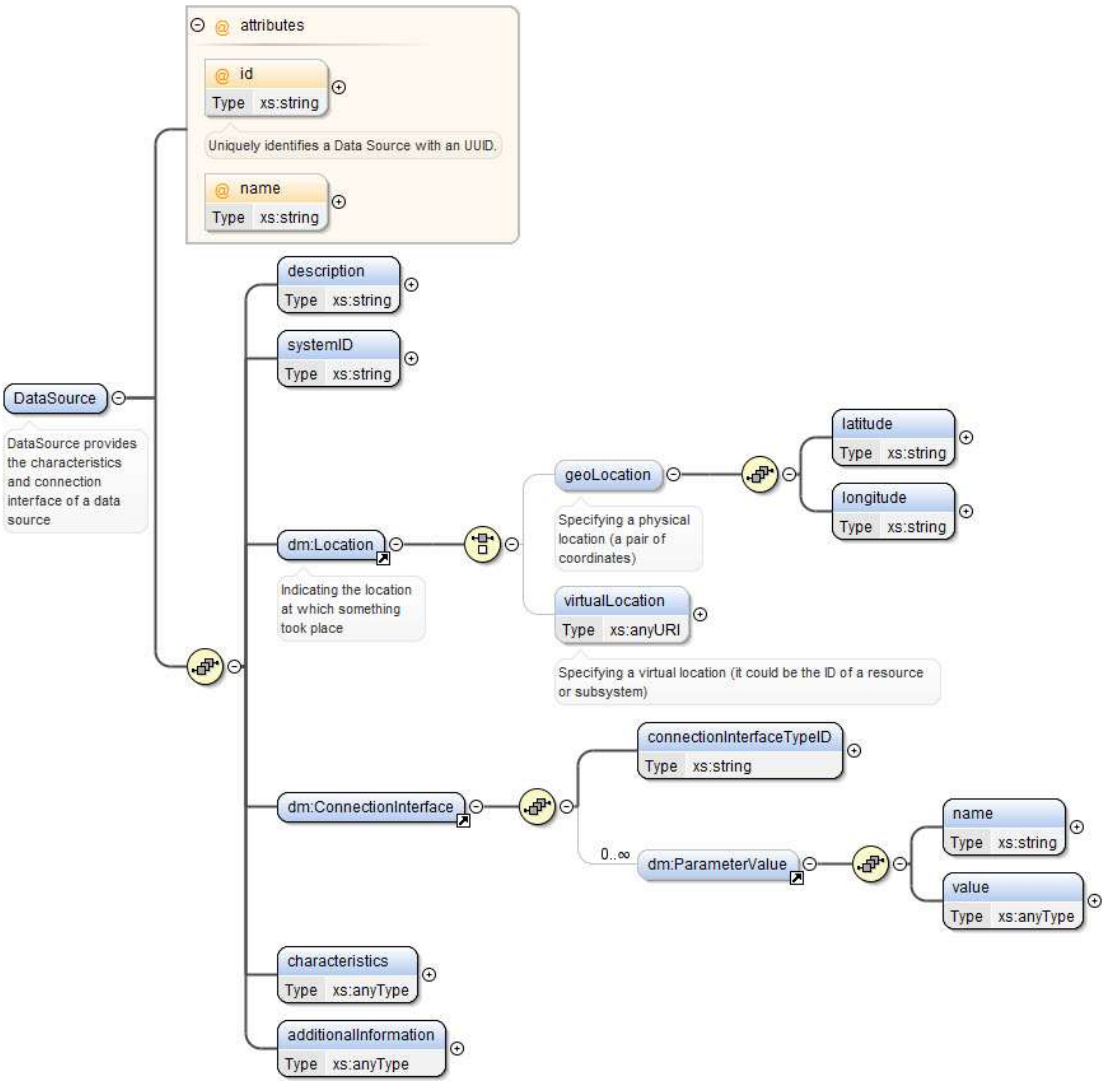


Figure 39: SecureIoT Data Source entity.

Data Destination: Data Destination provides the characteristics and connection interface of a data destination (Figure 40).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

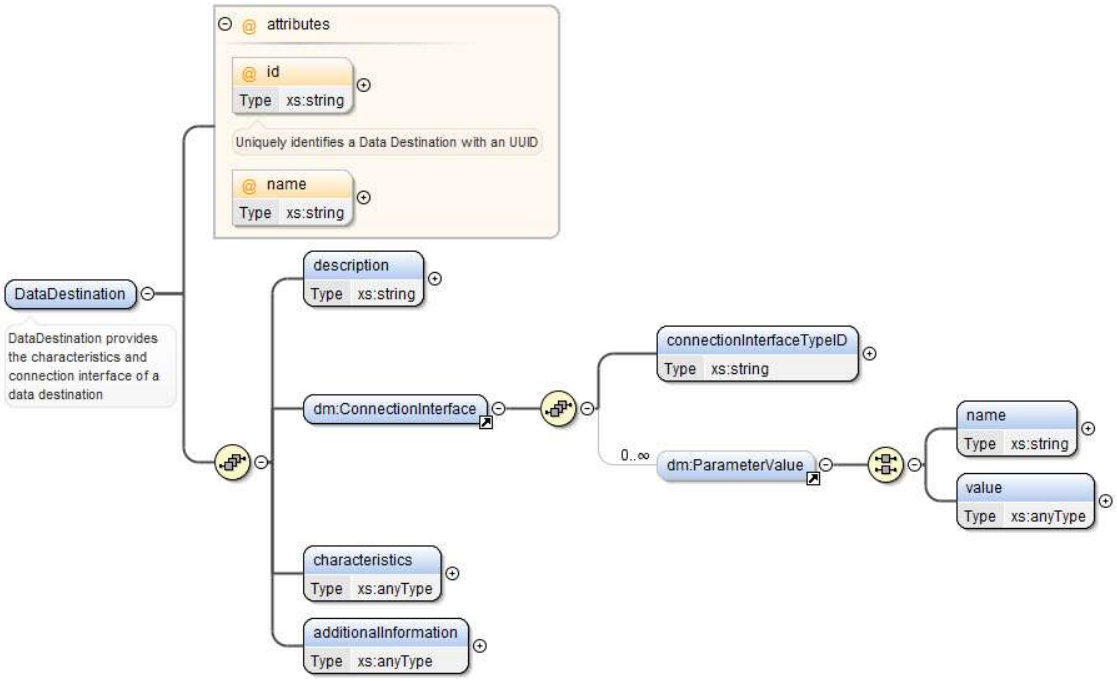


Figure 40: SecureIoT Data Destination entity.

Probe Type: models the type of probe which is used to categorize a Probe instance (Figure 41).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

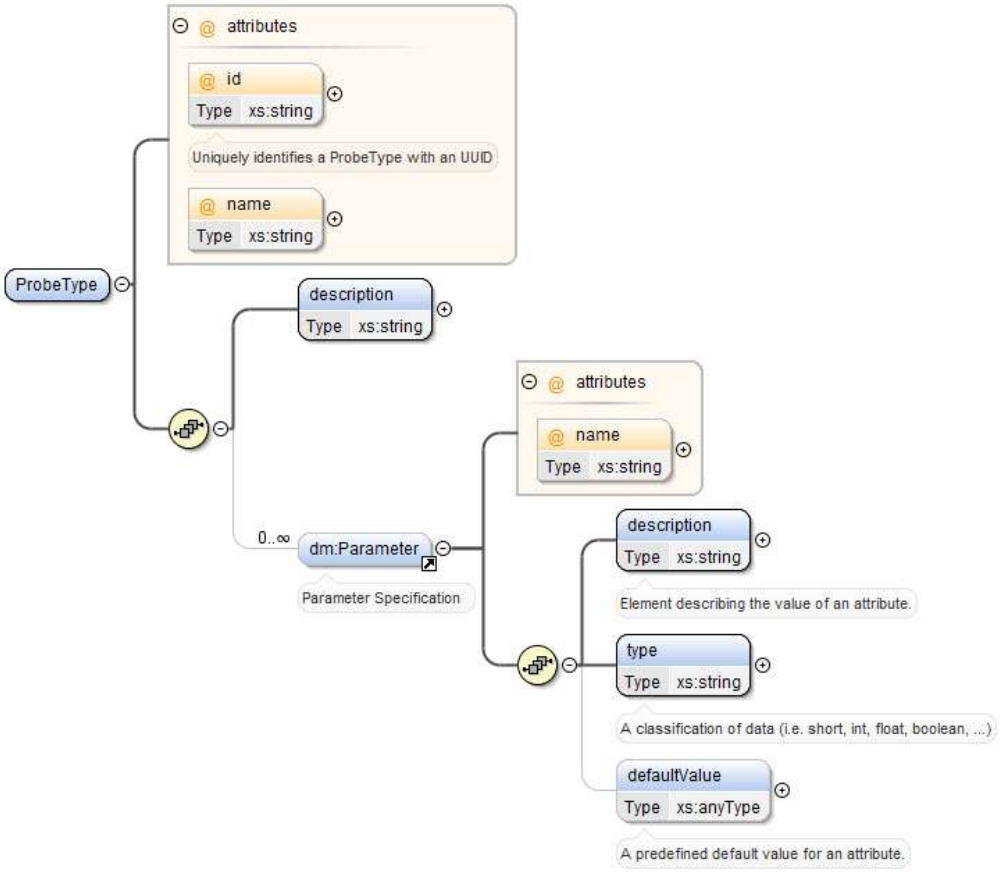


Figure 41: SecureIoT Probe Type entity.

Probe: Models the probe instances. The Probe entity contains the identity and configuration information of the probes deployed to the monitored IoT Systems (Figure 42).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

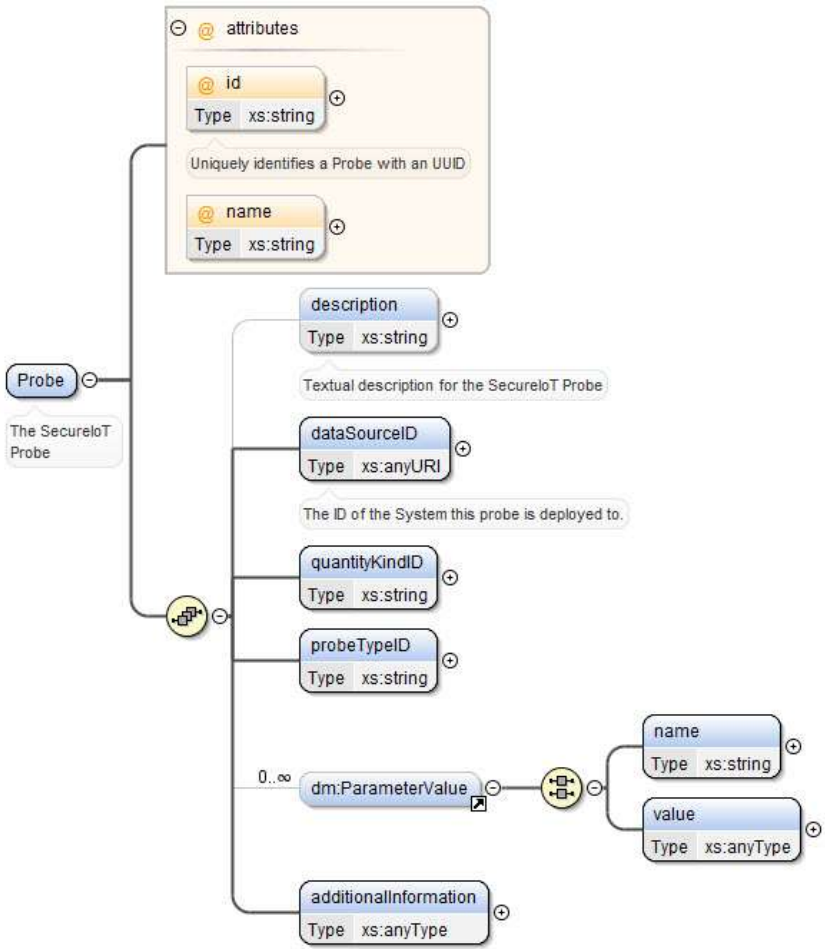


Figure 42: SecureIoT Probe entity.

The complete Data Management Model schema is available in the form of an XSD which is provided in Table 2 of Appendix A.

4.2 Analytics

The Analytics group focuses on the Analytics configuration and management at the Security Intelligence layer (Figure 32) and consists of the following entities.

Processor Definition: is used to describe an Analytics Algorithm and its parameters. As shown in Figure 43 it lists the metadata of the processor and its required parameters.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

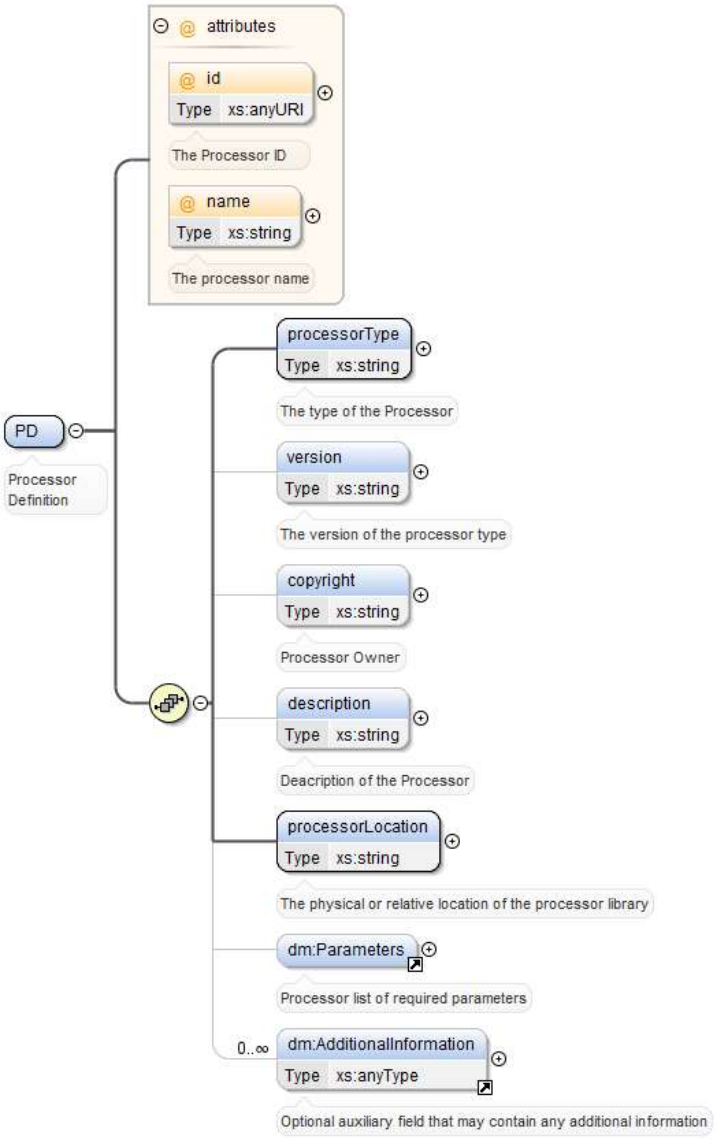


Figure 43: SecureIoT Processor Definition entity.

Processor Manifest: is an instantiation of a Processor Definition. As shown in Figure 44 it specifies the data sources that are consumed form the processor and also the data sources that it produces. Finally it provides the required parameters for the Analytics instantiation.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

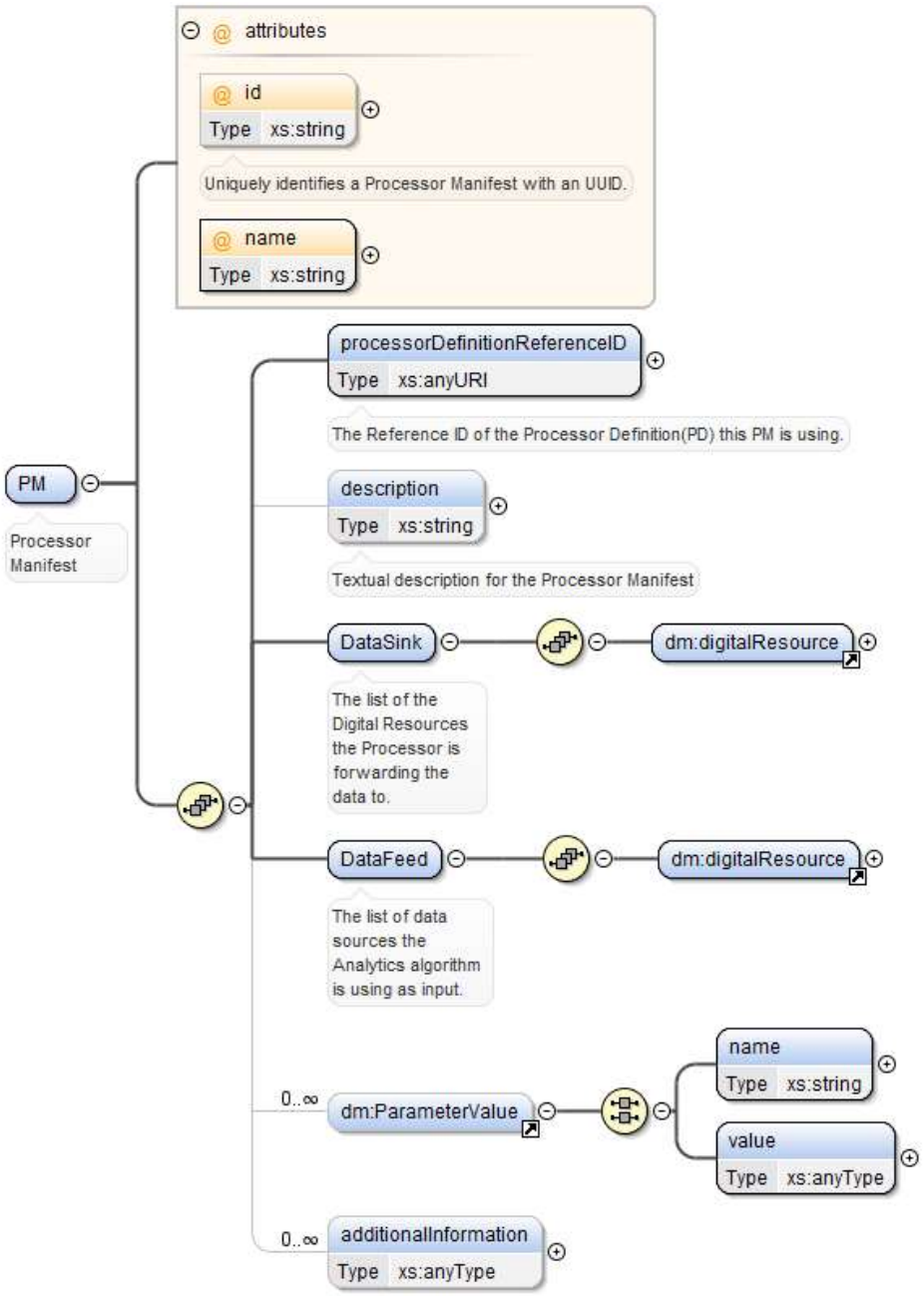


Figure 44: SecureIoT Processor Manifest entity.

Processor Orchestrator: it is capable of grouping a number of Processor instancies (Figure 45) to create a complex analytics data flow that is capable of providing adittional inteligence by utilizing multiple algorithms.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

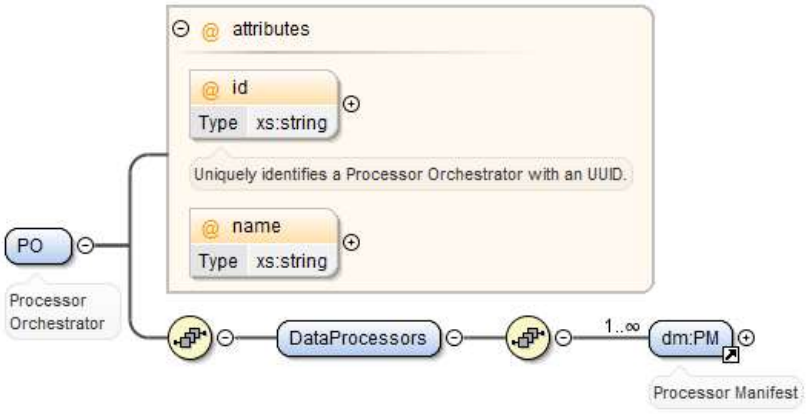


Figure 45: SecureIoT Processor Orchestrator entity.

The Analytics output is utilizing the Observation format (Figure 30) thus it indicates the data source, the system, the timestamp and the location respectively through the dataSourceID, systemID, timestamp and Location elements. The quantityKind of the Observation is a STIX (Structured Threat Information Expression) [OASIS] element. Its value includes one or more STIX indicators determined by the algorithm. Each one of these STIX indicators contains at least:

- An attackID
- An assetID and may contain
- An attackContext.
- Possible CVEs may be added (provided with join analysis) (from probes metadata if existing or coming from a 0-day CVE).

The complete SecureIoT Analytics schema is available in the form of an XSD, which is provided in Table 2 of Appendix A.

4.3 Knowledge Base

The Knowledge Base focuses on the Security Knowledge coming from third party standardized sources (Figure 33) which consists of:

- **CAPEC**: MITRE Common Attack Pattern Enumeration and Classification collection
- **CVE**: MITRE Common Vulnerabilities and Exposures collection
- **CWE**: MITRE Common Weakness Enumeration collection
- **CPE**: MITRE Common Platform Enumeration collection
- **CVSS**: NIST Common Vulnerability Scoring System collection

All these entities are using the standardized schemata provided from each organization which are imported to the SecureIoT Data Models. These entities are utilized from the other Logical Groups.

The complete Knowledge Base Data Model schema is available in the form of an XSD which is provided in Table 2 of Appendix A.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

4.4 Risk Assessment

The Risk Assessment Logical Group, focuses on the configuration and management of the Risks Assessment and Mitigations SECaaS (Figure 34). The RA Logical Group is heavily using the CMDB and Knowledge base entities for the configuration and management of the Engine. The main entities are:

Risk Model: represents a risk models based on specific Use Cases (Figure 46)

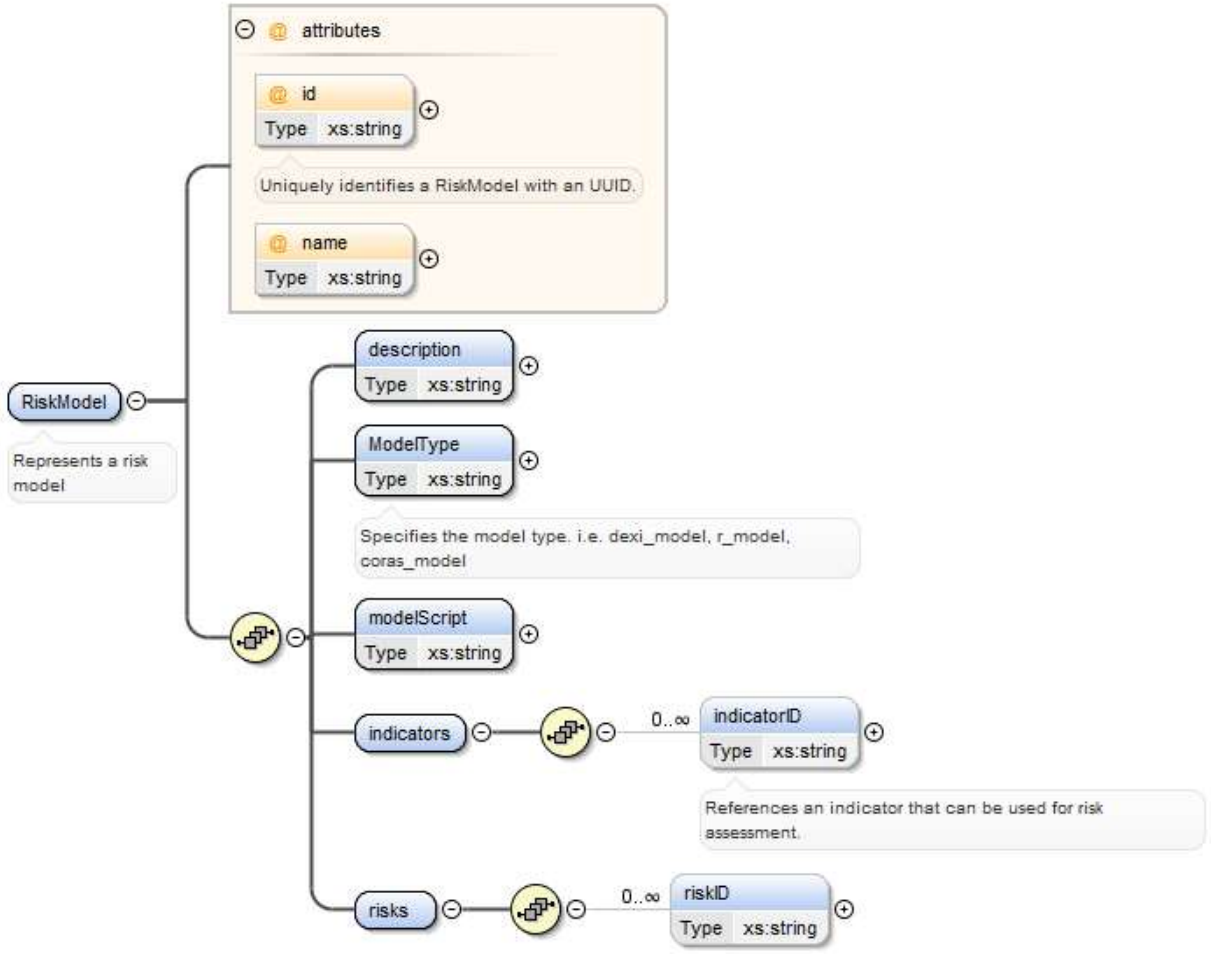


Figure 46: SecureIoT RAE Risk Model.

Indicator: represents an indicator that can be used for risk assessment.

Alarm: is used for reporting identified attacks from the Analytics layer

Specific Risk Model: represents a specific risk model (associated to a custom risk model that is only available to users of the same organization).

Risk Plan: represents a risk pattern (associated to a generic risk model that is available to users of all organizations).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

Risk Assessment Engine: provides the Risk Assessment Engine runtime configuration parameters.

Risk Report: represents a result of the risk assessment procedure that is delivered to the end user.

The complete Risk Assessment Engine Data Model schema is available in the form of an XSD which is provided in Table 2 of Appendix A.

4.5 Configuration and Management DB (CMDB)

The CMDB logical group focuses on configuration aspects of the SecureIoT infrastructure mainly modelling information related with the monitored IoT systems and the Attack Scenarios (Figure 35). It consists of the following core entities.

System: provides the observed IoT System characteristics.

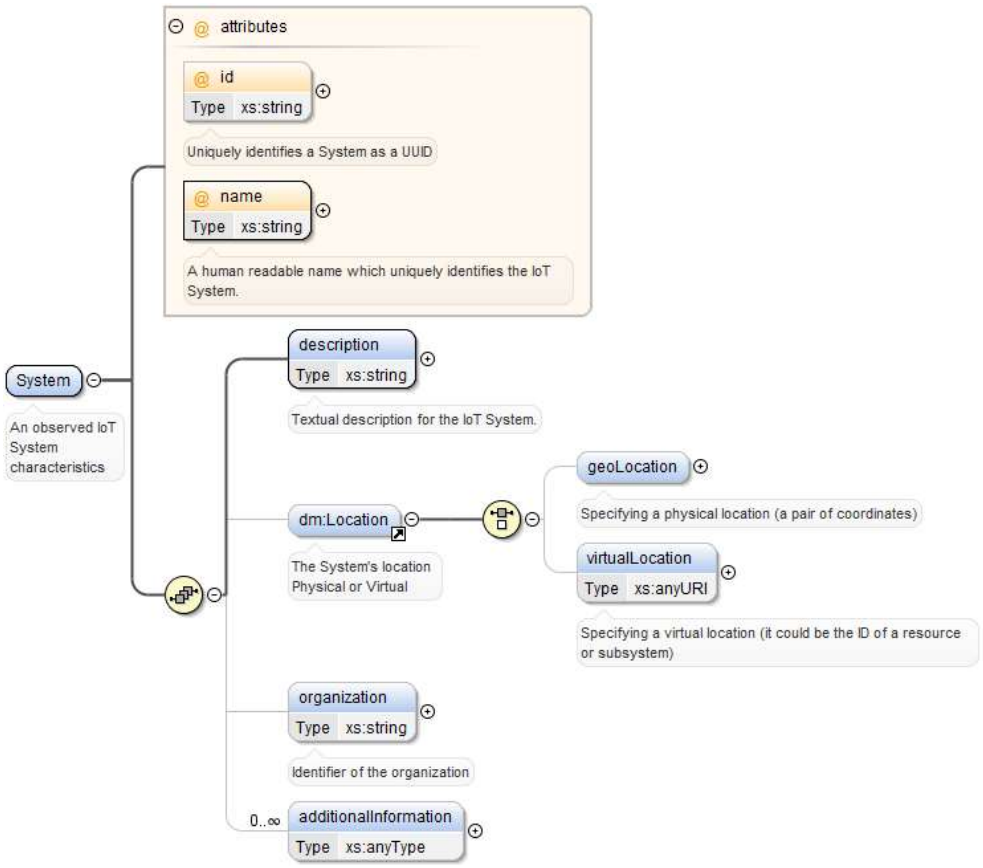


Figure 47: SecureIoT CMDB System entity.

Asset: models a cyber, physical or cyber-physical element within an organization that is used in the frame of business operations (Figure 48).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

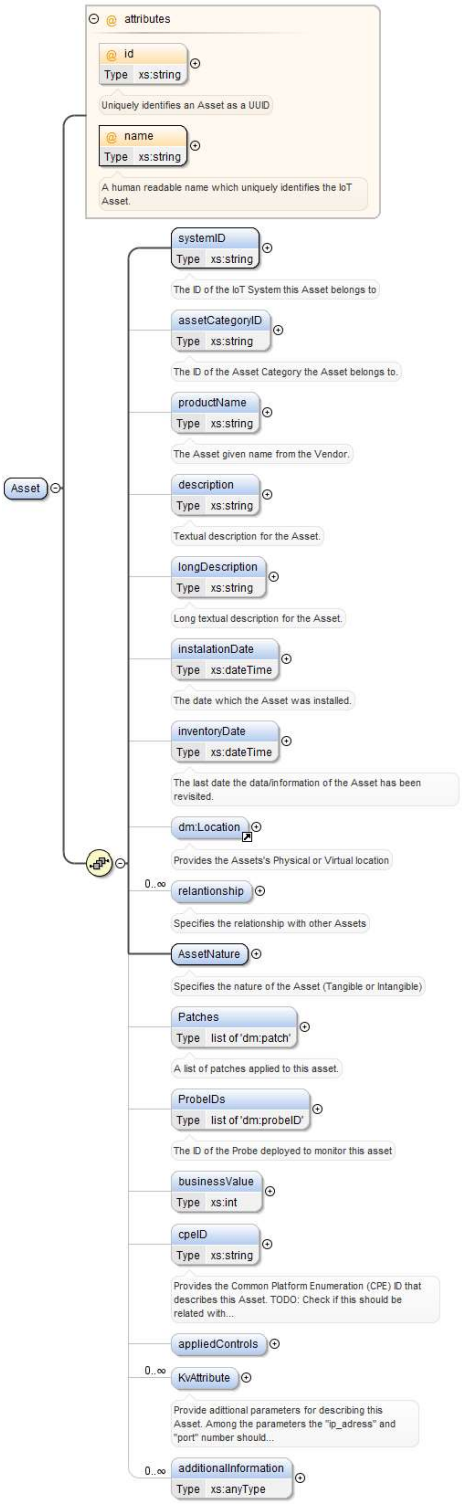


Figure 48: SecureIoT CMDB Asset entity.

Vendors: specifies an Asset Vendor (Figure 49).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

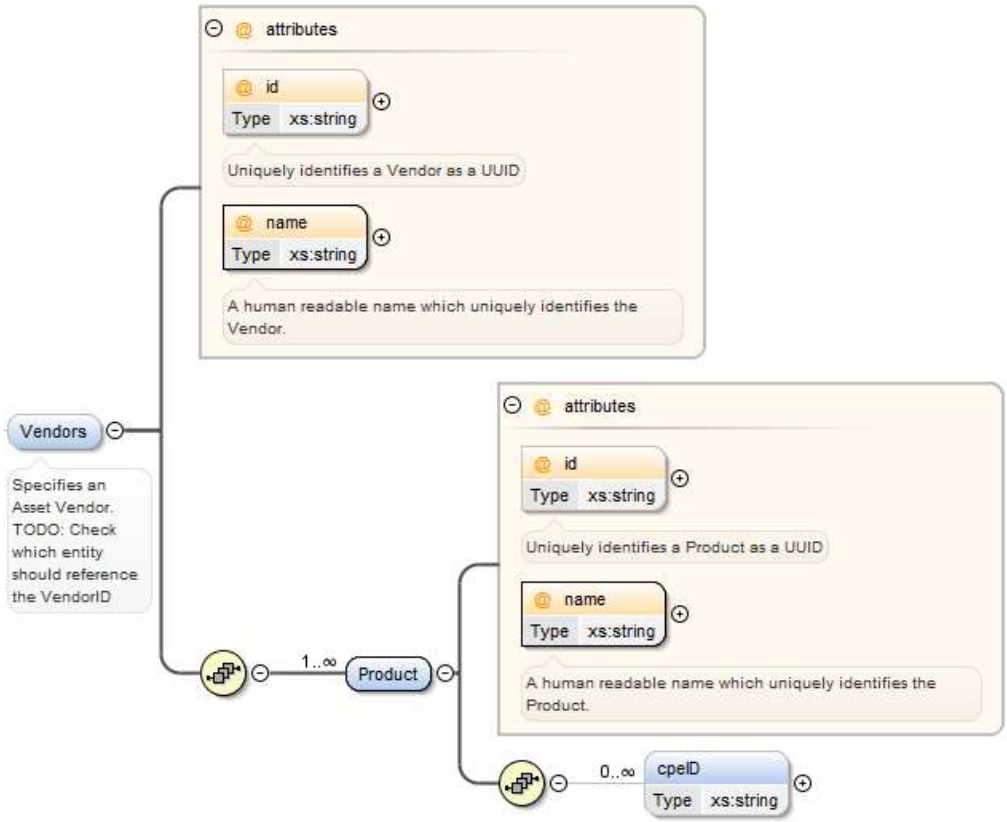


Figure 49: SecureIoT CMDB Vendors entity.

Control: models a control element capable to prevent attack impact (Figure 50). A control element can be applied at the Vulnerability Level or at the Threat Level (never to both of them).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

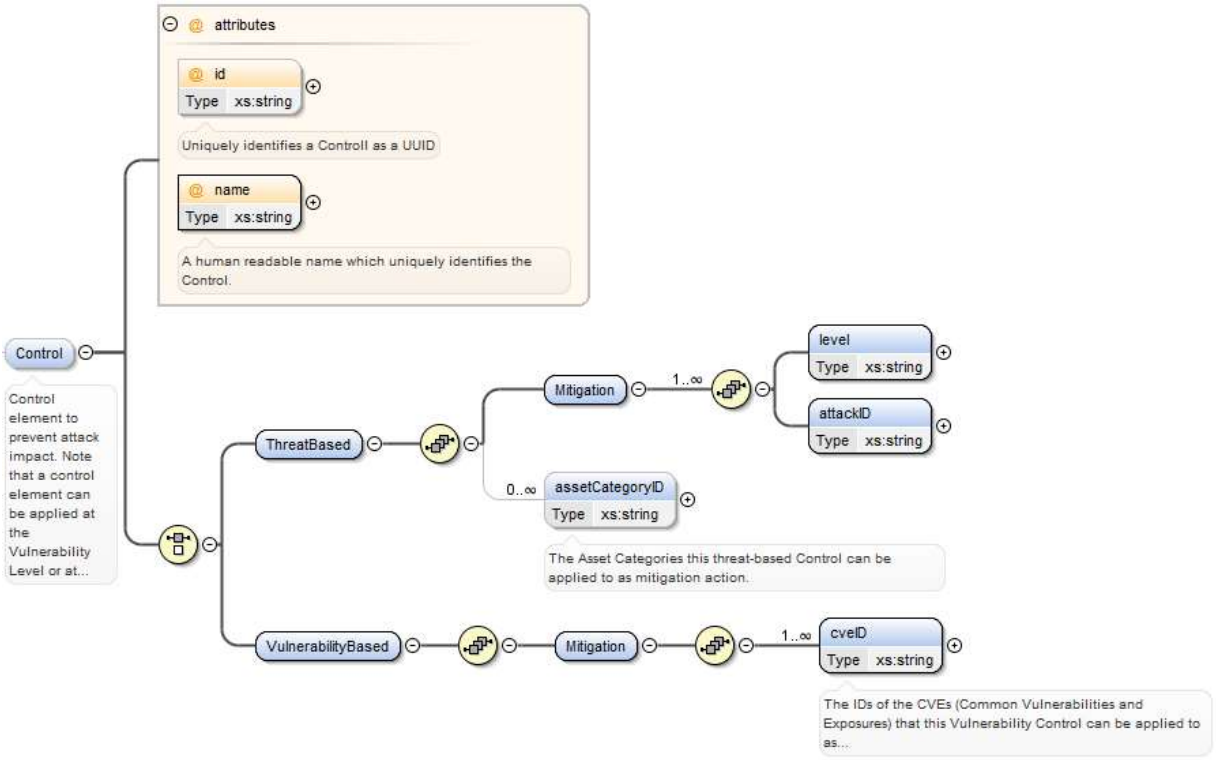


Figure 50: SecureIoT CMDB Control entity.

Mitigation Plan: provides a mitigation plan based on specified asset (Figure 51).

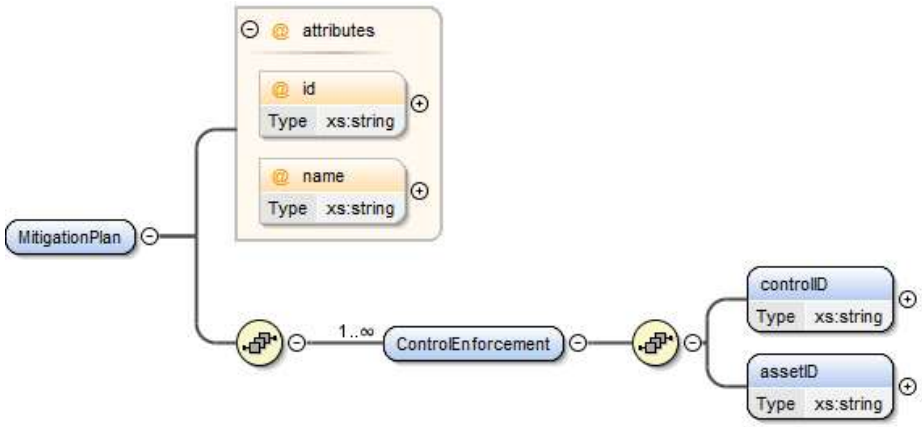


Figure 51: SecureIoT CMDB Mitigation Plan entity.

Mitigation Measures: provides a list of available Mitigation Measures to be applied for a given Abuse Case (Figure 52).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

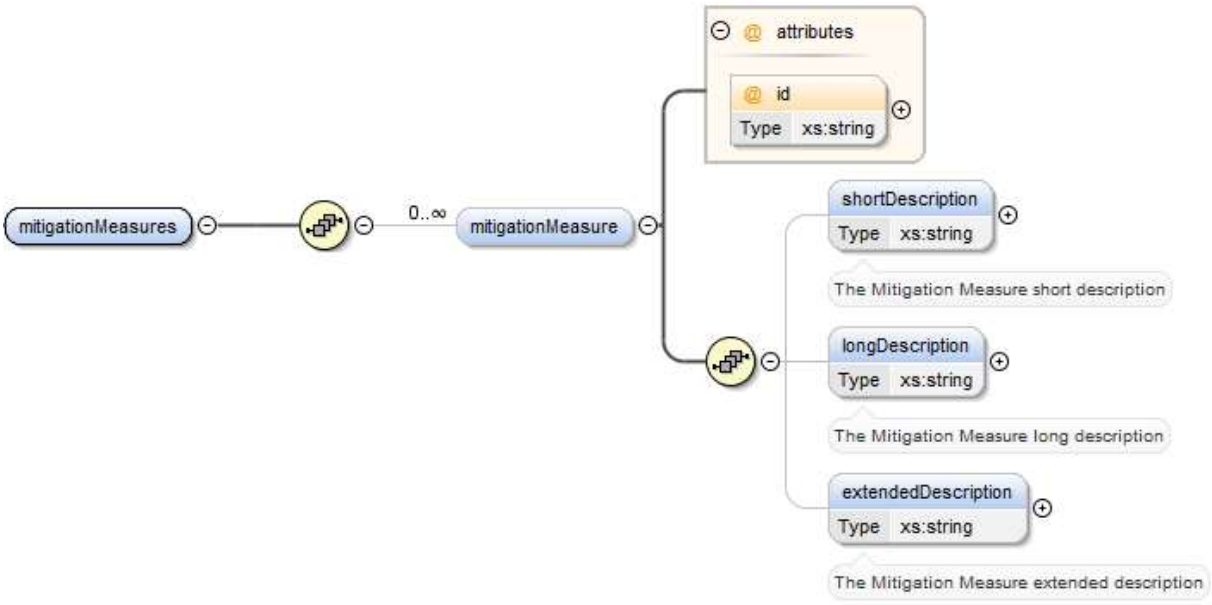


Figure 52: SecureIoT CMDB Mitigation Measures entity.

Vulnerabilities: is used to extend the MITRE Common Vulnerabilities and Exposures (CVE) collection (Figure 53).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

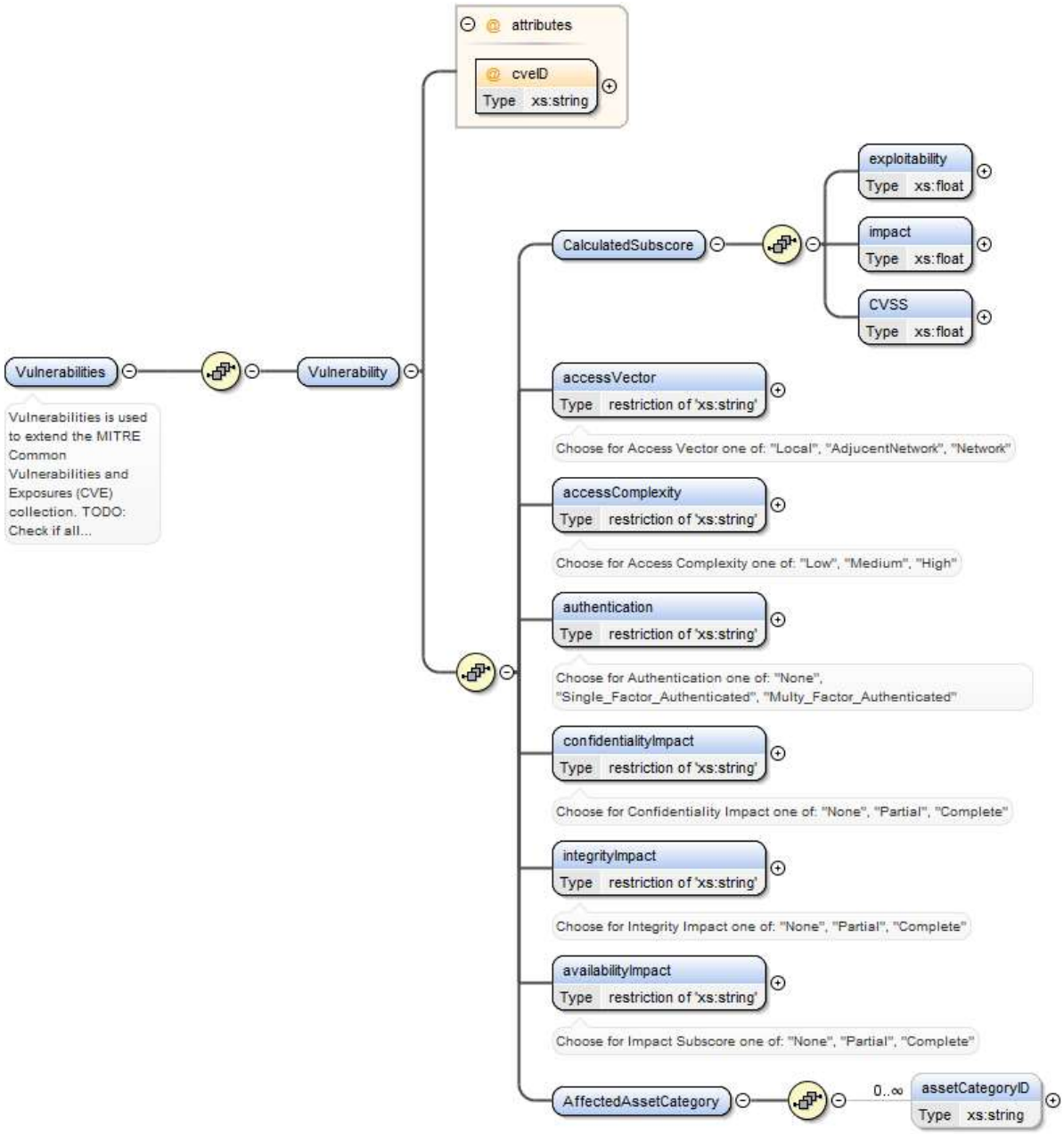


Figure 53: SecureIoT CMDB Vulnerabilities entity.

Attack Scenarios: models a set of Abuse Cases based on the monitored asset and specified scenarios (Figure 54).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

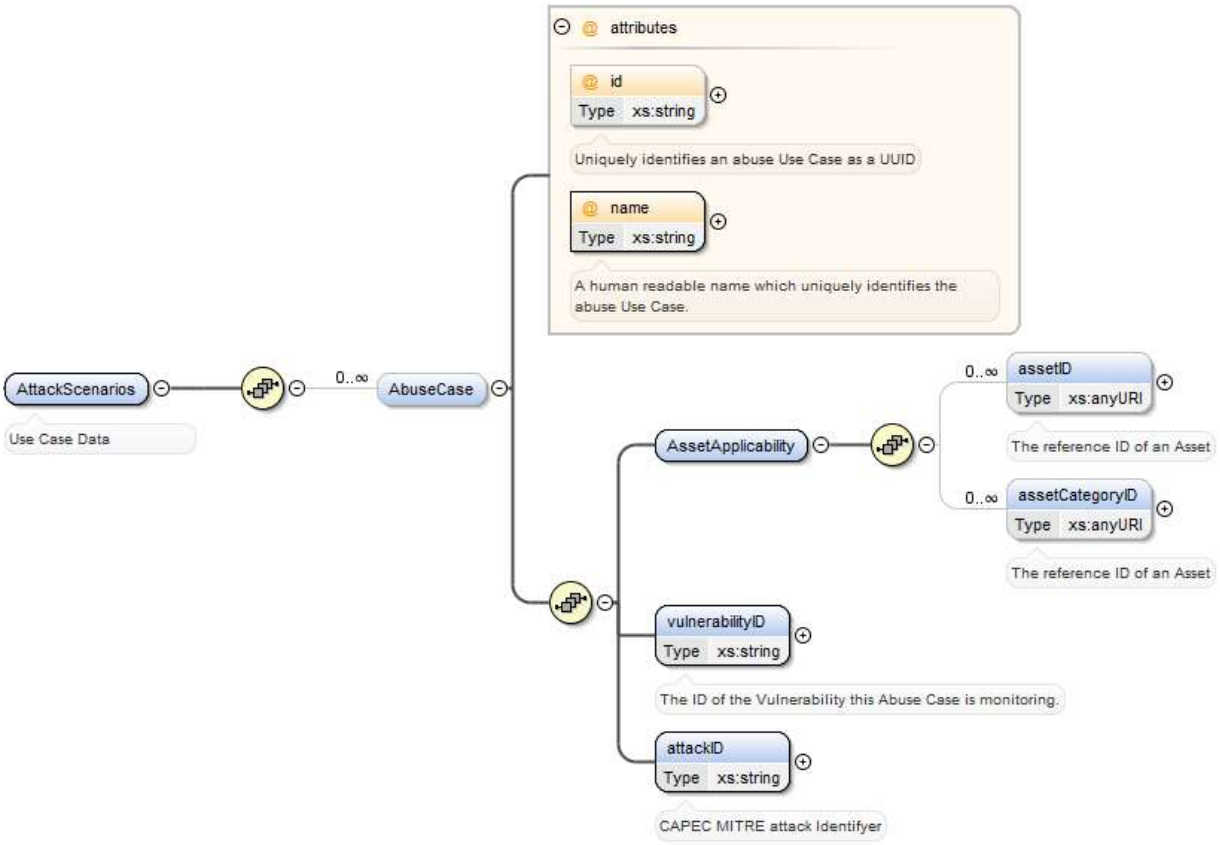


Figure 54: SecureIoT CMDB Attack Scenarios entity.

The complete CMDB Data Model schema is available in the form of an XSD which is provided in Table 2 of Appendix A.

4.6 Service Level Agreement (SLA)

The SLA Group, as mentioned above, focuses on the data models for configuring and reporting the SLA agreements.

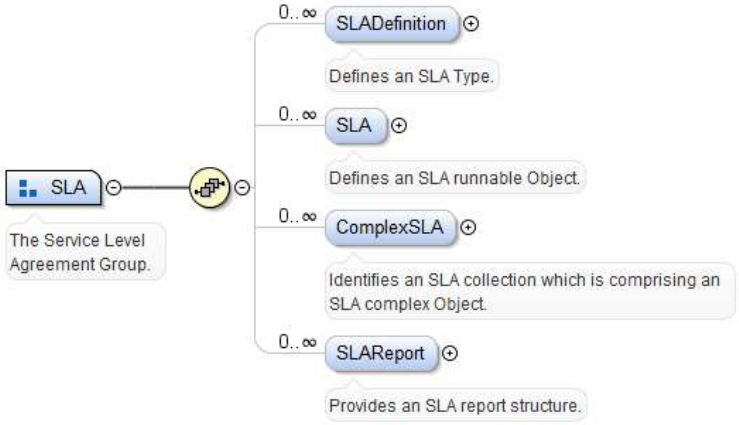


Figure 55: SecureIoT SLA Root Element.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

The root element of the SLA group is depicted in Figure 55 above and consists of:

SLA Definition: provides the definition of an SLA processor capable of calculating an SLA report (Figure 56).

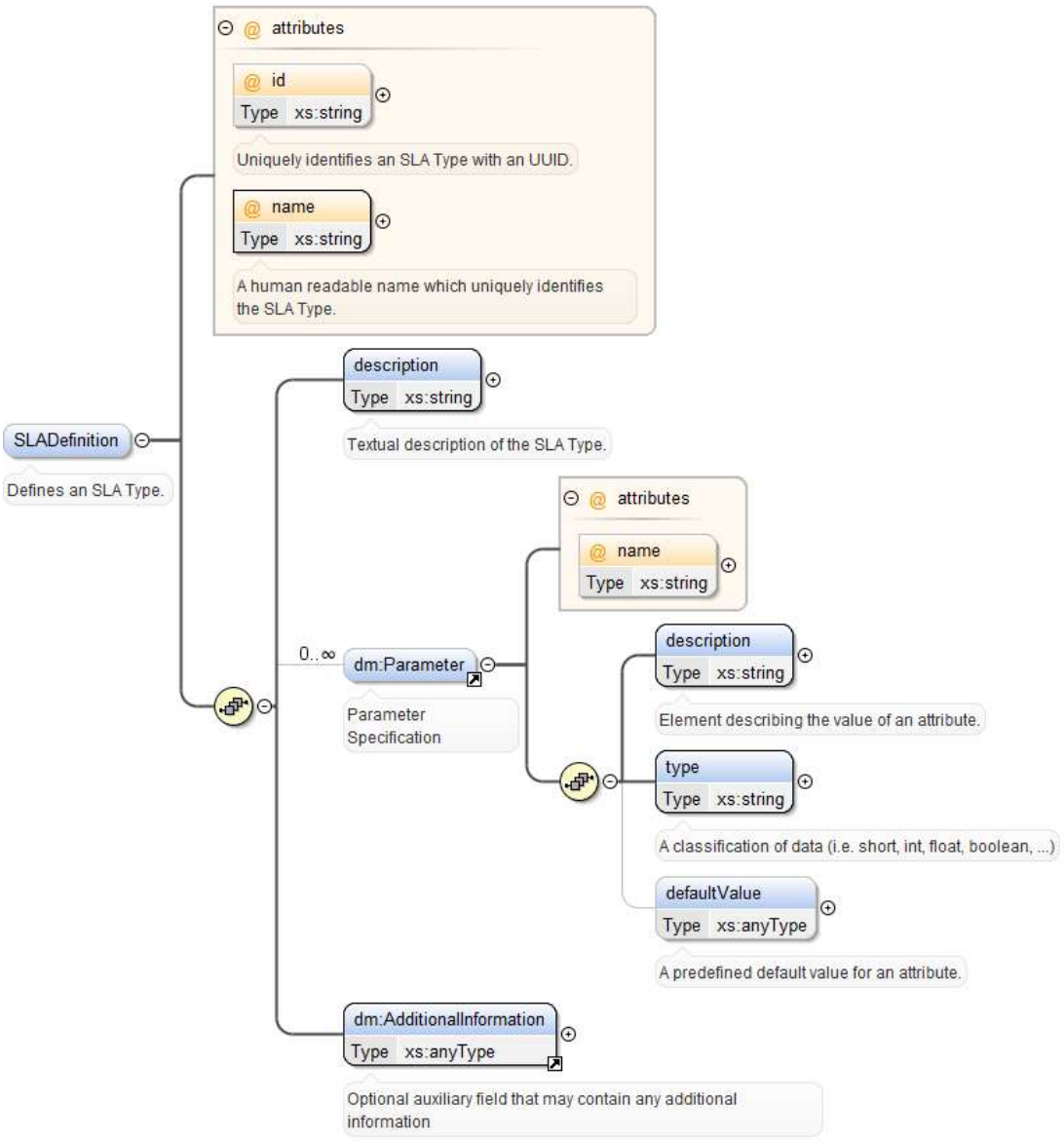


Figure 56: SecureIoT SLA Definition.

SLA Object: models the instantiation of an SLA definition. This object provides all the required configuration parameters for an SLA definition to be executed (Figure 57).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

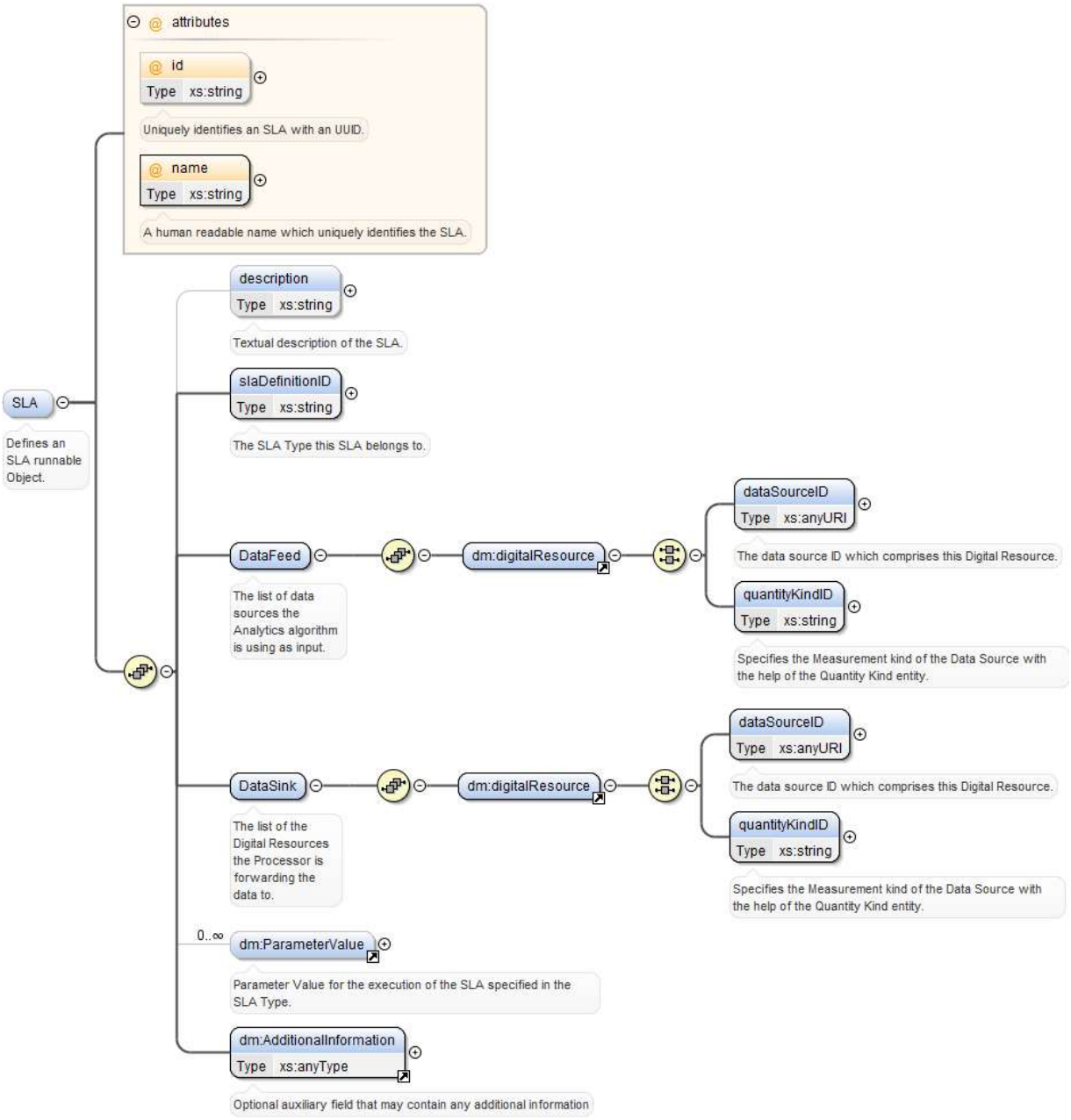


Figure 57: SecureIoT SLA Object.

Complex SLA: Identifies an SLA collection which is used to specify complex SLAs (Figure 58).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

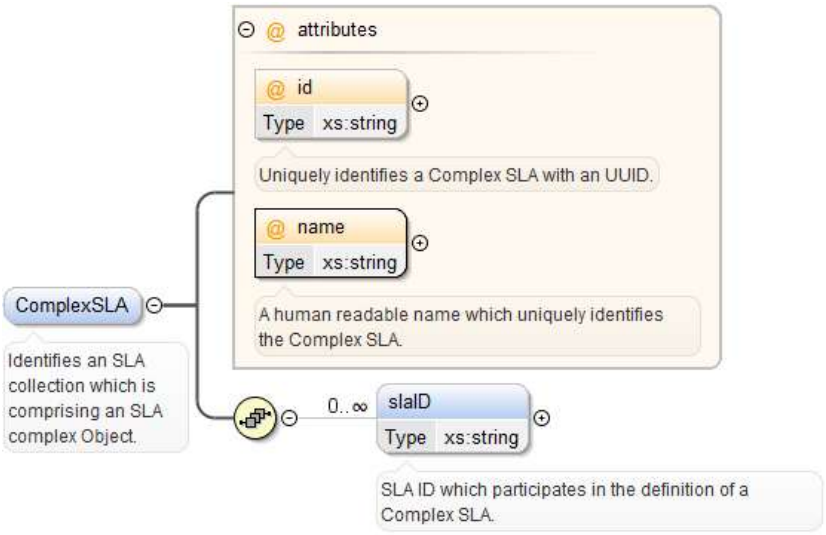


Figure 58: SecureIoT SLA Complex.

SLA Report: provides an SLA report structure. This entity is capable to host results calculated by an SLA object or a Complex SLA (Figure 59).

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

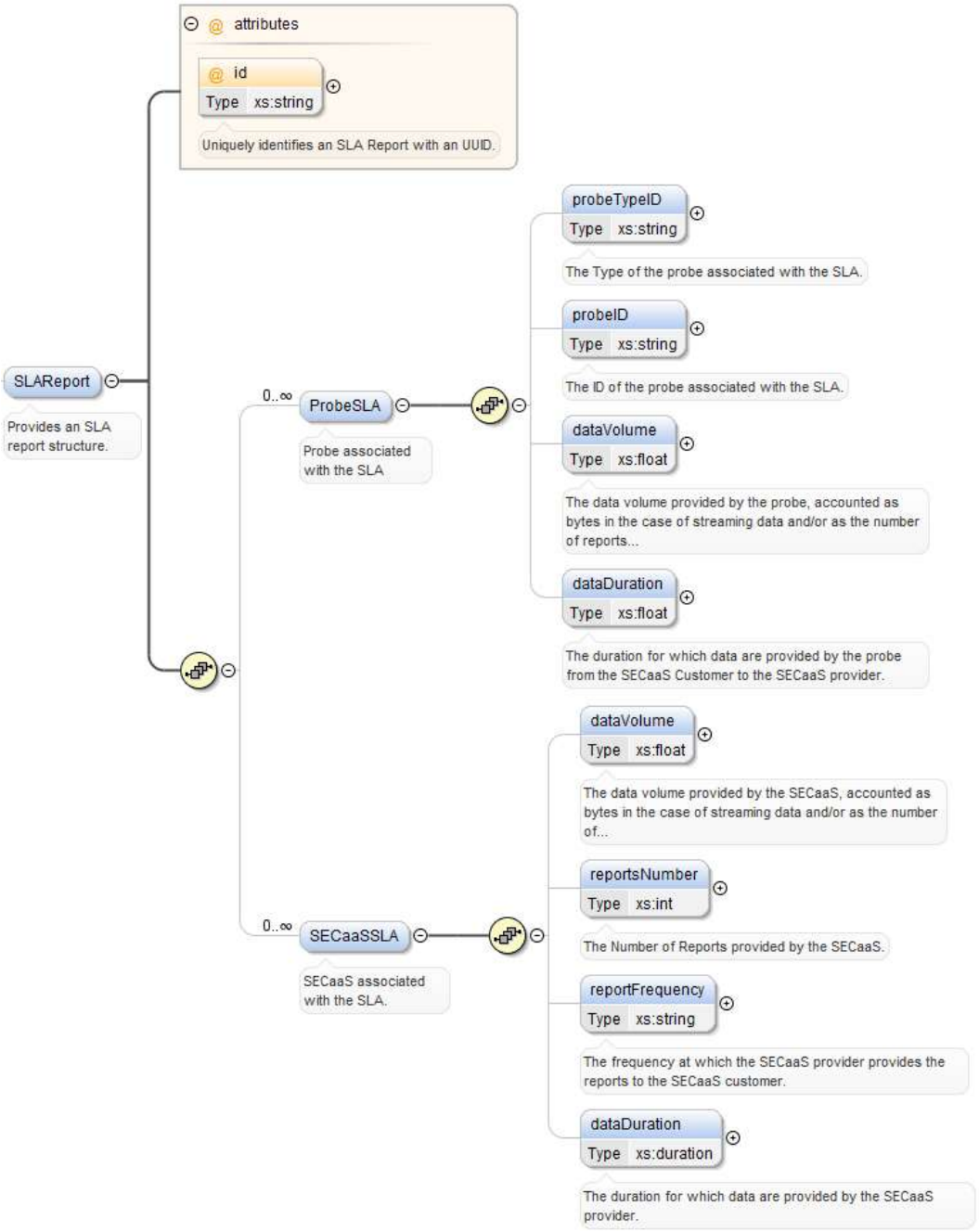


Figure 59: SecureIoT SLA Report.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

5 SecureIoT Use Cases Architectures

5.1 Industry4.0 Security Use Cases

The integrated Industry 4.0 usage scenario is comprised of three distinct usage scenarios spanning different levels within the IoT context. The first level is the shop floor with machinery that produces a *product* and *machine operators* who control the machines and have access to human machine interfaces such as *head mounted displays* or IoT devices in general. In this case, the machinery involved is an injection molding machine producing electrical connectors.

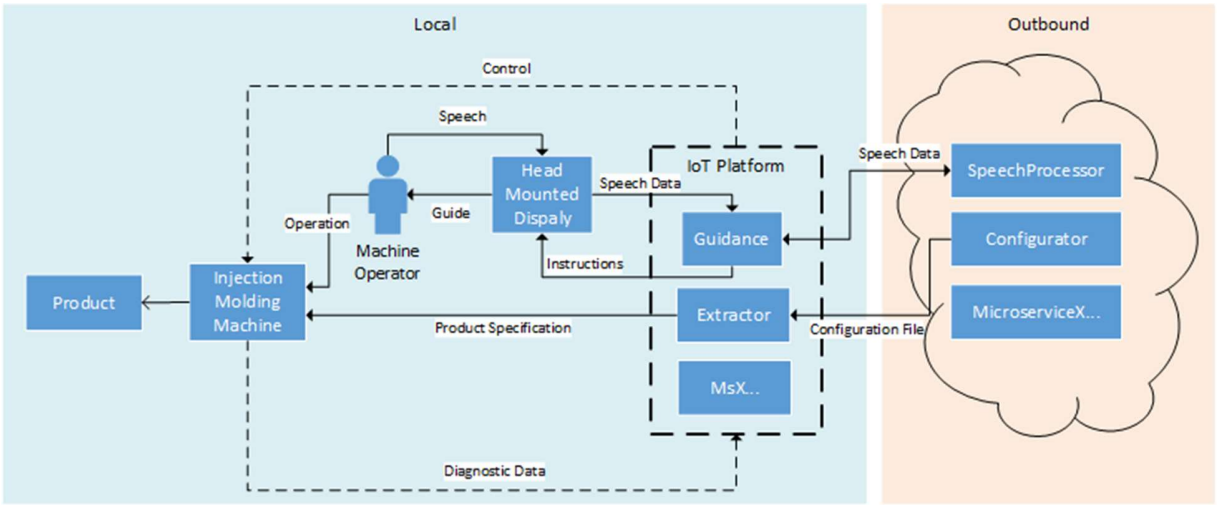


Figure 60: Industry 4.0 Use Case Overview.

The second level consists of the *IoT Platform* which is hosted within the organization that owns the shop floor. The first and second levels are considered the "Local" environment. Lastly, there can also be IoT services provided by external organizations. This third level is considered the "Outbound" environment.

On the shop floor level, the injection molding machine is controlled by the machine operator. In addition, it receives control instructions from the IoT Platform and send diagnostic data to it. The machine operator uses an IoT device with a human machine interface, i.e., a head mounted display. This device guides the machine operator and receives commands and feedback from the operator in turn, e.g., through speech inputs. It may also collect other data, such as location data and visual data. An exemplified data flow for diagnostics that encompasses possible placement of probes is shown in Figure 61.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

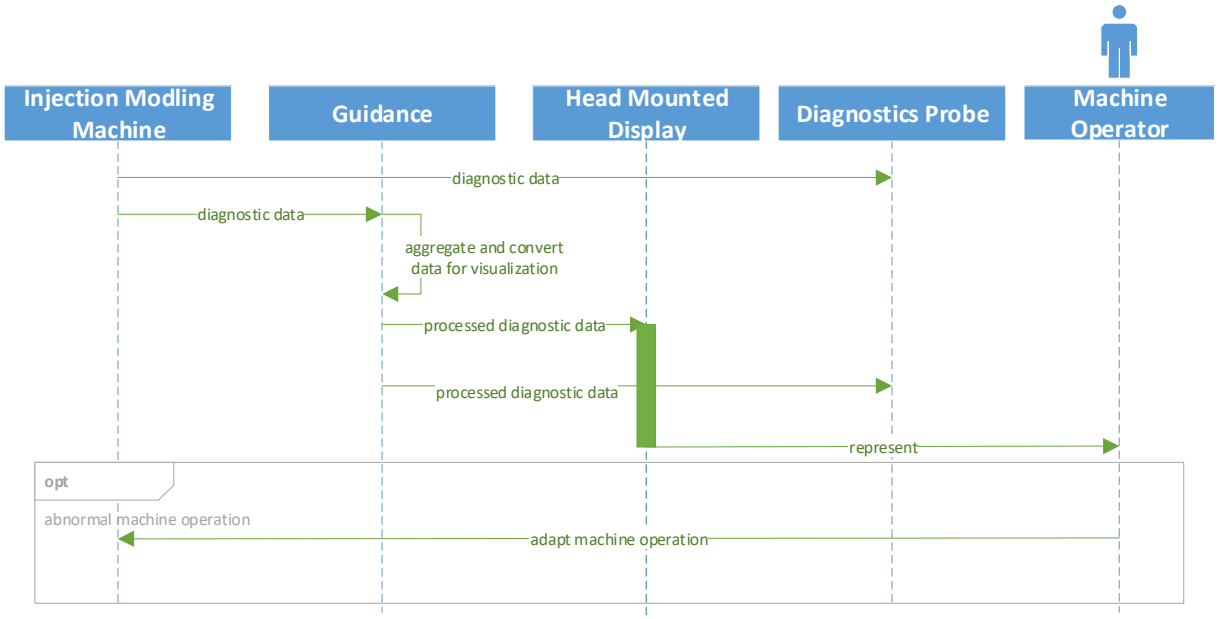


Figure 61: Example Diagnostic Data Probe.

To guide the machine operator, the head mounted display receives instructions from the IoT Platform and sends the machine operator's input to the platform for processing. Responsible for data processing on the IoT platform are microservices, each with a specific set of responsibilities. For example, one microservice may be responsible for speech processing, while another is responsible for sending control commands to the machinery on the shop floor, and yet another guides the machine operator. An exemplified data flow for guidance that encompasses possible placement of probes is shown in Figure 62.

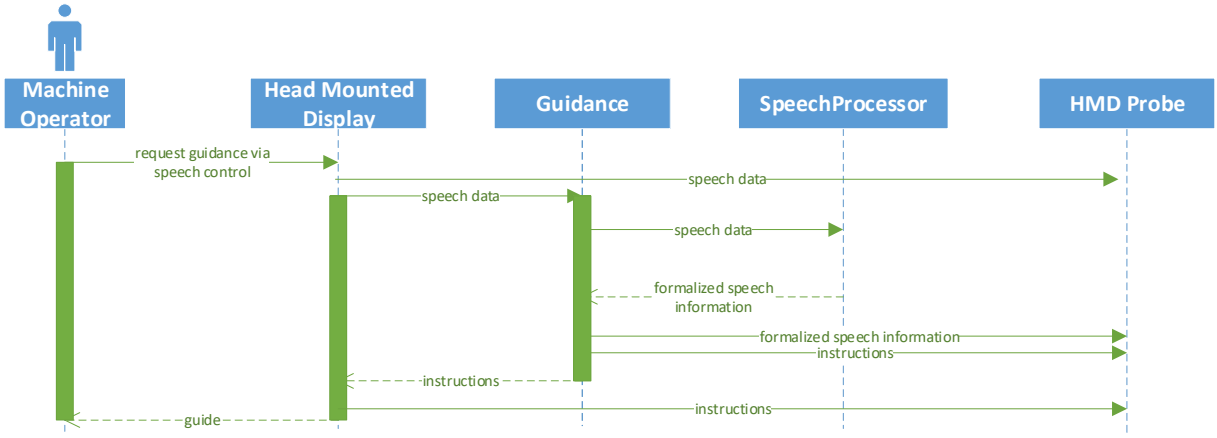


Figure 62: Example HMD Data Probe.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

In addition to the local IoT Platform microservices, it may be possible to integrate third party services from partners outside of the local organization. These services may also influence the control signals from the IoT platform to the shop floor machinery.

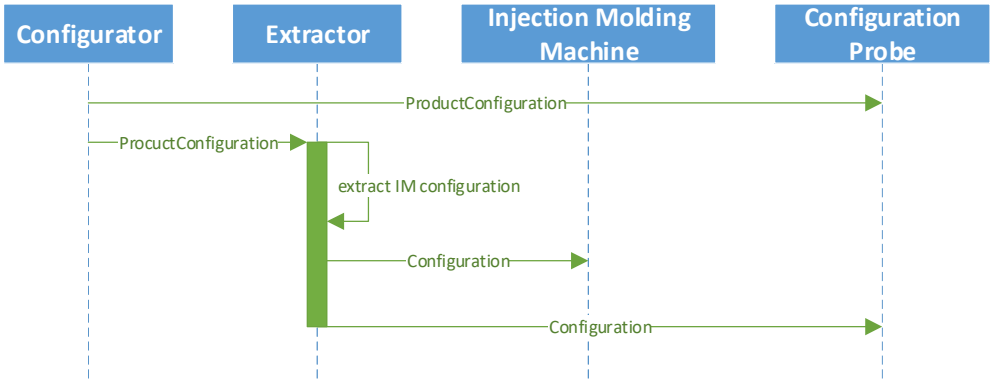


Figure 63: Example Configuration Data Probe.

One example of an outbound service is a product *configurator*. This configurator is used by engineers to virtually assemble a product from supplier provided parts. The result is a product configuration file, that can be send to a production machine to assemble it. In this case, the product configuration file is send to an *extractor*, which extracts parts specifications from the product configuration so that the injection molding machine can produce the required part. An exemplified data flow for configuration that encompasses possible placement of probes is shown in Figure 63.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

5.1.1 Logical View

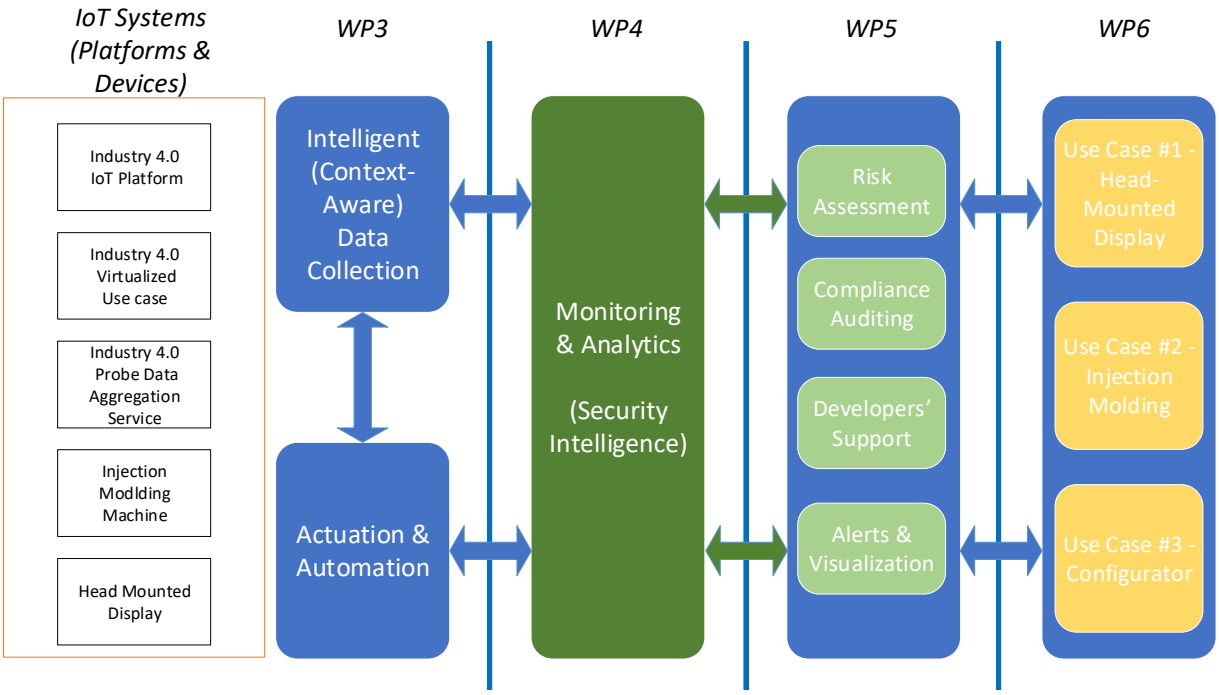


Figure 64: High Level Logical View of the SecureIoT Architecture and Mapping.

5.2 Healthcare and Socially Assistive Robots Use Cases

The Socially Assistive Robots and IoT Applications (SAR) scenario consists of multiple edge devices at edge and different cloud platforms.

The QTrobot itself consists of a main control board (Intel x86 multicore) running a Linux operating system. This board control communicates with each robot motor's controller via specific serial communication protocol to command each motor and read the sensory feedback. Other peripheral devices such as robot 3D camera and microphone array are connected to the main board using USB interfaces. The communication with the external world of the robot is done using two WIFI interface; one is mostly used to connect robot to the QTrobot Tablet and the other one connects robot to different cloud platforms via home network.

QTrobot Tablet is an android-based tabled which communicate with the QTrobot via direct WIFI connection. The communication protocol between robot and tablet is based on Robot Operating System (ROS). This tablet is mostly used to program and launch user's applications using QT graphical programming interface.

QTrobot Cloud is the placeholder for robot applications, user profiles and many other contents. QTrobot communicates with the cloud platform using different web services and RESTful API.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

Home component of the CC2U system includes Ethernet, Bluetooth and Wi-Fi interfaces, multiple USB interfaces to connect with different peripherals such as wearable sync dongles and sensor boards. It collects data from the home sensors, processes it and sends it to the CC2U cloud.

CC2U cloud is a server-based environment that collects data from all CC2U User exposed devices, processes it and feeds it to the user through the web-based UI or to other 3rd party platforms through RESTful API.

Data probes will be implemented at different communication points in edge such as component internal communication (e.g., motor commands/joint positions), QTrobot tablet (ROS messages) and CC2U home gateway (collects data from the home sensors). Moreover, data probes will be also implemented in cloud to, for example, monitors information (REST JSON/data) passed through QTrobot to CC2U and QT cloud platforms. The details of data collection are explained in SAR "Use cases specifications" [SecureIoTD6.1]. These probes provide the logging capabilities of a system to demonstrate accountability for compliance auditing services.

Figure 65 shows an overview of the different subsystems and their connections in the Socially Assistive Robots and IoT Applications scenario. Further details about the architecture and technologies that will be used in the "Socially Assistive Robots and IoT Applications (SAR)" Use Cases are included in [SecureIoT D6.1].

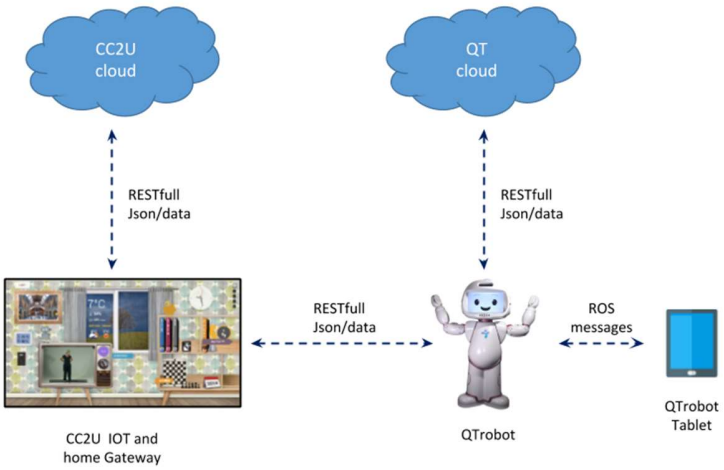


Figure 65: Overview of the different subsystems involved in the Socially Assistive Robots and IoT Applications scenarios.

5.2.1 Logical View

The application of SecureIoT architecture to the Socially Assistive Robots (SAR) use-cases will imply collecting security, system and application level information from the main IoT assets such as QTrobot and CC2U cloud platform, QTrobot, QTrobot Table and CC2U home gateway.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

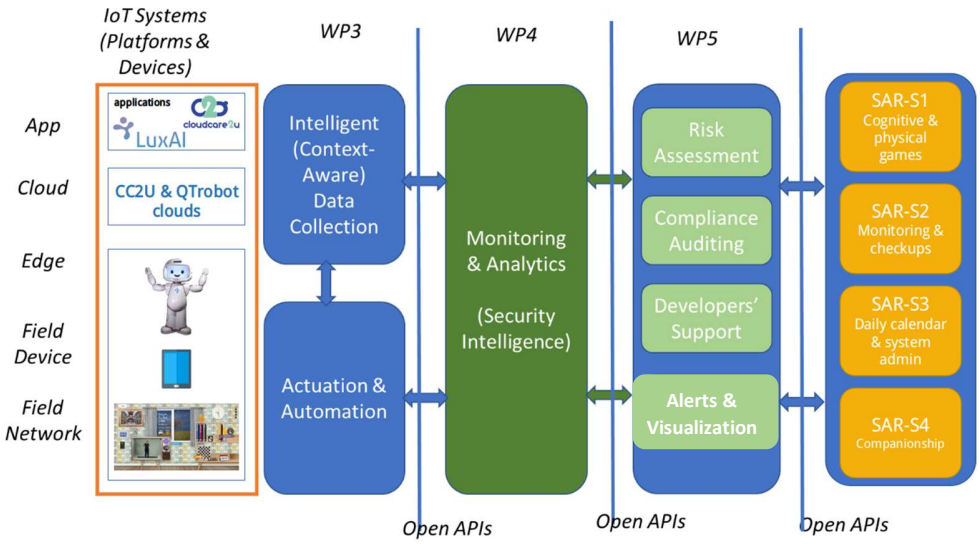


Figure 66: Logical View of SAR Security System.

Figure 66 presents the logical view of SecureIoT mapped to the SAR scenarios and assets. The socially assistive robots and IoT applications are categorized in four different scenarios each consists of multiple use cases. The scenarios have been constructed in an extensible manner to validate different SECaaS services by involving more actors, services and covering different security requirements.

Table 1 summarizes the main coverage of SECaaS services by each scenario.

Table 1: SECaaS Services for SAR Scenarios.

Scenario ID	Scenario Name	Main SECAS services (with order of coverage)
SAR1	Cognitive & physical games	1. Risk Assessment and Mitigation 2. Developers Support
SAR2	Monitoring & check-ups	1. Compliance Auditing 2. Risk Assessment and Mitigation 3. Developers Support
SAR3	Daily Calendar and System Admin	1. Compliance Auditing
SAR4	Companionship	1. Developers Support 2. Compliance Auditing

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

The coverage of the SECaaS services is not limited to what listed in the above table. For example, the IoT risk assessment and mitigation service can be evaluated in each scenario separately. This is due to the extensible nature of the designed scenarios to involve more actors and subsystems (edge and cloud) as scenarios evolve from SAR1 to SAR4.

As we have described before, data probes will be implemented at different edge and cloud levels. The application level data will also be monitored at different level depending on each use cases. An interesting example of application level probe within the Cognitive & physical games scenario is to monitor high-level commands from game application and relevant robot's motor commands/joint positions.

5.3 Connected Car and Autonomous Driving Use Cases

For the "*Connected Car and Autonomous Driving*" (CAD) Use Cases, the *Field Device* is an IoT vehicle OBU (onboard unit), this device is embedded inside a vehicle that performs data capture on vehicle data networks. This device also facilitates the transport of captured data from the vehicle (acting as the *Edge* device) to the IoT Platform.

The IDAPT device (OBU) periodically captures vehicle performance data from the vehicle CAN bus, movement data from the embedded GPS/IMU module and environmental data from nearby vehicles/infrastructure from the embedded V2X module. This data is aggregated in the IDAPT OBU and sent to the IoT platform for data processing. Additionally, the IDAPT OBU will provide probe data to the SecureIoT platform from sources such as the vehicle CAN bus.

The FIWARE IoT platform is used to receive, validate and store data sent by the IDAPT OBU. A back-end application will exploit the FIWARE IoT platform functionalities, capabilities and APIs to gain access to the vehicle data provided by the IDAPT OBU and to apply data aggregation and logic based on the respective Use Case.

Further details about the architecture and technologies that will be used in the "*Connected and Autonomous Driving"* Use Cases are included in subsection 5.2 of SecureIoT D6.1.

Data probes will be implemented in both the IDAPT OBU and in the FIWARE IoT Platform to facilitate the validation of end to end data transmission. An example of an application level probe would be to ensure movement data (such as a vehicle speed) is consistent from the initial data capture (raw vehicle CAN data), IDAPT OBU application processing (application data/JSON), IDAPT OBU network traffic (IP packets), IoT Platform reception (IP Packets) and IoT Platform (FIWARE Cygnus/Orion Context Broker). An example of this can be found below in Figure 67.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

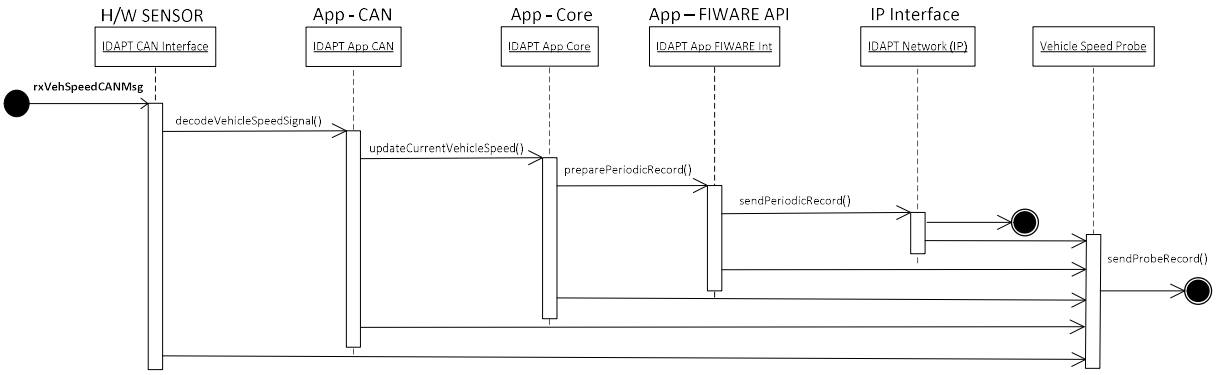


Figure 67: Example Vehicle Speed Probe - IDAPT OBU.

Application level probes such as the example above and system level probes such as CAN bus load will be provided to the SecureIoT SECaaS services to facilitate the monitoring of irregular messaging activity on the vehicle CAN bus for *risk assessment* services (see Figure 68).

Using these probes provides the logging capabilities of a system to demonstrate accountability for *compliance auditing* services (see Figure 68).

5.3.1 Logical View

The application of SecureIoT architecture to the CAD use-cases will imply collecting security, system and application level information from the main IoT assets involved: the IDAPT OBU and the different components or Generic Enablers (GEs) or FIWARE. Figure 69 presents the logical view of SecureIoT (mapped to the use-cases assets.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

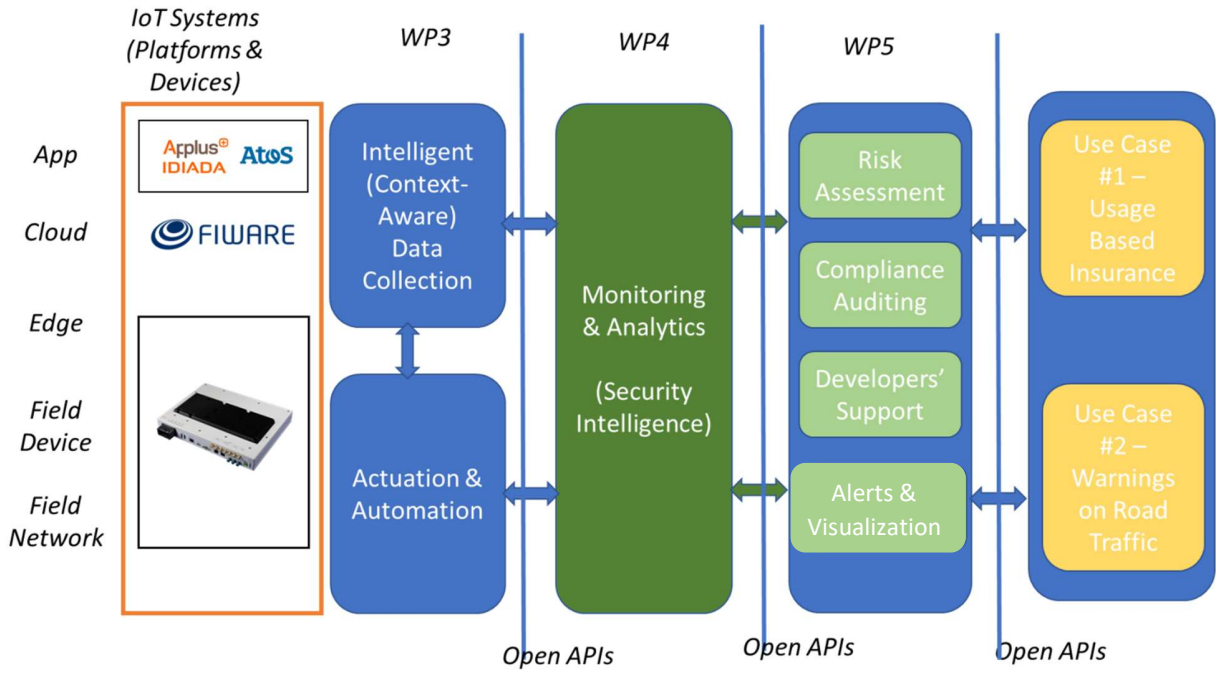


Figure 68: High Level Logical View of the SecureIoT Architecture and mapping

At the same time, at the monitoring and analytics layer, specific security template shall be developed in order to extract advanced knowledge, to detect abnormalities and threats from the collected data or even to predict them in a running deployment. Nevertheless, CAD security templates will be executed by SecureIoT execution engine (Figure 69) and therefore additional component at this tier will not be needed.

Finally, the holistic security of the CAD use-cases will be improved, monitored and assessed thanks to the SECaaS developed within WP5.

As it can be seen, the main adaptations needed to apply SecureIoT to WP6 CAD use-cases are limited to the development, customization and deployment of appropriate data probes. A more detailed view of the data collection layer is presented in Figure 69, highlighting the new components explicitly created for the sake of the CAD scenarios.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

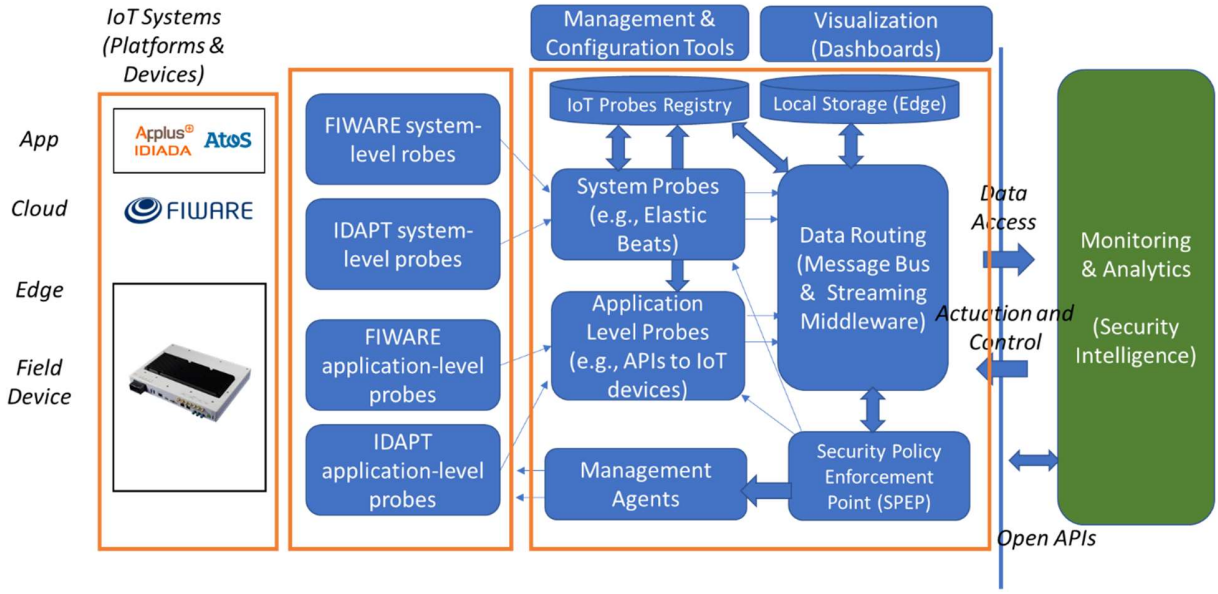


Figure 69: Anatomy of the Data Collection and Actuation Layer including the system and application probes to be developed and deployed for the *Connected Car and Autonomous Driving* use-cases.

Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

6 Conclusions

This deliverable presents the SecureIoT architecture, which specifies the main building blocks of the security systems that will be developed in the course of the project and drives their integration. The SecureIoT architecture specification has been driven by a number of requirements and concepts that have been introduced by standards-based reference architectures and their security frameworks. In particular, it has been specified in order to support end-to-end security, based on monitor→analyse→act functionalities. Moreover, the SecureIoT architecture is specified as a complementary, overlay and cross-cutting systems to IoT platforms.

As a first step to presenting the architecture of SecureIoT systems the documents has presented a high-level reference architecture, which serves as communication device, by introducing the various tiers/layers and cross-cutting functions of any SecureIoT system. Accordingly, a more detailed technical architecture for SecureIoT systems is presented, focusing on its main modules in the form of its logical, physical, development and process view of the architecture.

The alignment to standards-based architecture has led to the reuse of some quite well-known concepts and building blocks in the IoT Security community. Nevertheless, some novel concepts have been also introduced, such as a mechanism for data-driven extraction of security events and behaviours, based on powerful analytics algorithms. This mechanism has been introduced based on the notion of templates, which specify the behaviours that the system is destined to capture. By supporting an arbitrary number of different templates (including their automated extraction), the SecureIoT systems could be become flexibly and intelligently configurable to address many different threats, vulnerabilities and attacks. Most important, templates can be "contextualized", i.e., tailored to operate in different environments and based on different parameters. Overall, this mechanism has reinforced the data-driven nature of the project, while transferring a great part of the system's security intelligent in the deployment and use of advanced data analytics algorithms.

The architecture has also introduced other important modules that contribute to the system intelligence. As a prominent example, the introduction of a Security Knowledge Base allows the system to benefit from accumulated knowledge about threats and attacks, as a means of correlating identified behaviours with known information about security incidents that have occurred in the past. Likewise, the specification of automation modules provides the means for increasing the efficiency of the system, through reducing or eliminated human intervention whenever possible.

For validating the architecture, the document presents the logical architectural view of the use cases, which make use of the SecureIoT platform based on the defined architecture. Detailed technical architectures for each use case are however produced as part of the use cases design

and implementation in WP6 of the project. Moreover, the detailed specification and implementation of the security modules of the SecureIoT architecture has been reported in the other technical workpackages of the project (WP3, WP4 and WP5). This document has presented the enhancements and refinements of the architecture resulting from feedback received by the development tasks of its components. The final validation of the SecureIoT architecture will result from the completion of the project trial scenarios.

References

[Kruchten95] Kruchten, "Architectural blueprints - The "4+ 1" view model of software architecture," IEEE Software, 1995.

[Schweichhart16] Karsten Schweichhart: Reference Architectural Model Industrie 4.0 - An Introduction, April 2016, Deutsche Telekom, online resource: https://ec.europa.eu/futurium/en/system/files/ged/a2-schweichhart-reference_architectural_model_industrie_4.0_rami_4.0.pdf

[OASIS] https://oasis-open.github.io/cti-documentation/stix/intro

[SecureIoTD2.1] SecureIoT D2.1 – Reference Scenarios and Use Cases, April 2018

[SecureIoTD2.2] SecureIoT D2.2 – Analysis of Stakeholders' Requirements, May 2018

[SecureIoTD2.4] SecureIoT D2.4 – Architecture and Technical Specifications of SecureIoT Services, First version. September 2018

[SecureIoTD3.3] SecureIoT D3.3 – Intelligent data collection mechanisms and APIs, J. Soldatos and all, 2018.

[SecureIoTD3.9] SecureIoT D3.9 – Report on SLAs for Security Monitoring, First Version, June 2019

[SecureIoTD5.1] SecureIoT D5.1 – IoT Risk Assessment and Mitigation as a Service, First Version, December 2018

[SecureIoTD5.4] SecureIoT D5.4 – IoT Complaiance Auditing as a Service, First Version, December 2018

[SecureIoTD5.7] SecureIoT D5.7 – IoT Developers Support as a Service, First Version, December 2018

[SecureIoTD6.1] SecureIoT D6.1 – Detailed specification of usage scenarios and planning of validation activities First version – S. Kyriazakos and all, 2018.

Appendix A. Data Models

This Appendix presents the data models that have been specified and accompany the architecture.

Table 2 shows the complete SecureIoT XML schema (XSD).

Table 2: SecureIoT Data Model Schema.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="eu:secureiot:global:dm"
   elementFormDefault="qualified" attributeFormDefault="unqualified"
   xmlns:dm="eu:secureiot:global:dm" xmlns:dk="eu:secureiot:data:dk"
   xmlns:cir="http://www.openoandm.org/ws-cir/" xmlns:capec="http://capec.mitre.org/capec-3"
   xmlns:cvss="http://scap.nist.gov/schema/cvss-v2/0.2">
   <xs:import namespace="http://www.openoandm.org/ws-cir/"
      schemaLocation="thirdparty/cir/CommonInteroperabilityRegistry.xsd"/>
   <xs:import namespace="http://capec.mitre.org/capec-3"
      schemaLocation="thirdparty/mitre/capec/ap_schema_v3.0.xsd"/>
   <xs:import namespace="http://scap.nist.gov/schema/cvss-v2/0.2"
      schemaLocation="thirdparty/nist/nvd/cvss/cvss-v2_0.2.xsd"/>

   <xs:element name="SecIoT-DM">
      <xs:annotation>
         <xs:documentation>SecureIoT Data Model</xs:documentation>
      </xs:annotation>
      <xs:complexType>
         <xs:sequence maxOccurs="1">
            <xs:element minOccurs="0" name="InteroperabilityRegistry" type="cir:Registry"
               maxOccurs="unbounded"/>
            <xs:group ref="dm:DataManagement"/>
            <xs:group ref="dm:Analytics"/>
            <xs:group ref="dm:KnowledgeBase"/>
            <xs:group ref="dm:RiskAssessmentEngine"/>
            <xs:group ref="dm:CMDB"/>
            <xs:group ref="dm:SLA"/>
            <xs:element minOccurs="0" ref="dm:Observation"/>
         </xs:sequence>
      </xs:complexType>
   </xs:element>
   <xs:group name="DataManagement">
      <xs:annotation>
         <xs:documentation>Data Management data models</xs:documentation>
      </xs:annotation>
      <xs:sequence>
         <xs:element ref="dm:QuantityKind"/>
         <xs:element ref="dm:ConnectionInterfaceType"/>
         <xs:element ref="dm:DataSource"> </xs:element>
         <xs:element ref="dm:DataDestination"/>
         <xs:element ref="dm:ProbeType"/>
         <xs:element ref="dm:Probe"/>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
        <xs:element ref="dm:System"/>
    </xs:sequence>
</xs:group>
<xs:group name="Analytics">
    <xs:annotation>
        <xs:documentation>Analytics data models</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:PO"/>
        <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:PM"/>
        <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:PD"/>
        <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:SecurityTemplate">
            <xs:annotation>
                <xs:documentation/>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:group>
<xs:group name="KnowledgeBase">
    <xs:annotation>
        <xs:documentation>Knowledge Base data models</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="CAPECcolection">
            <xs:annotation>
                <xs:documentation>MITRE Common Attack Pattern Enumeration and Classification
                    (CAPEC) collection</xs:documentation>
            </xs:annotation>
            <xs:complexType>
                <xs:sequence>
                    <xs:element maxOccurs="unbounded" minOccurs="0"
                        ref="capec:Attack_Pattern_Catalog"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="CVEcollection">
            <xs:annotation>
                <xs:documentation>MITRE Common Vulnerabilities and Exposures (CVE)
                    collection</xs:documentation>
            </xs:annotation>
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="CVE" maxOccurs="unbounded" minOccurs="0">
                        <xs:annotation>
                            <xs:documentation>Common Vulnerability and
                                Exposure</xs:documentation>
                        </xs:annotation>
                        <xs:complexType>
                            <xs:sequence>
                                <xs:element name="_id" type="xs:string">
                                    <xs:annotation>
                                        <xs:documentation>Uniquely identifies a CVE as an
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                UUID</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="1" maxOccurs="1" name="id"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>ID of the CVE in the format
                    CVE-year-number. For example: "CVE-2018-
                    13074".</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="1" maxOccurs="1" name="summary"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>A brief summary of the
                    CVE</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="capecID"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>ID of the CAPECs associated to the
                    CVE</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="cweID"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>ID of the CWEs associated to the
                    CVE</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="cpeID"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>ID of the CPEs associated to the
                    CVE</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="1" maxOccurs="1" name="cvss"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>CVSS score associated to the
                    CVE</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="1" maxOccurs="1" name="lastModified"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>Date of the last modification in the
                    official source</xs:documentation>
            </xs:annotation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                    </xs:element>
                    <xs:element minOccurs="0" maxOccurs="unbounded"
                      name="references" type="xs:string">
                      <xs:annotation>
                        <xs:documentation>List of sites referring to this
                          CVE</xs:documentation>
                      </xs:annotation>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="CWEcollection">
          <xs:annotation>
            <xs:documentation>MITRE Common Weakness Enumeration(CWE)
              collection</xs:documentation>
          </xs:annotation>
          <xs:complexType>
            <xs:sequence>

              <xs:element name="CWE" maxOccurs="unbounded" minOccurs="0">
                <xs:annotation>
                  <xs:documentation>Common Weakness Enumeration</xs:documentation>
                </xs:annotation>
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="_id" type="xs:string">
                      <xs:annotation>
                        <xs:documentation>Uniquely identifies a CWE as an
                          UUID</xs:documentation>
                      </xs:annotation>
                    </xs:element>
                    <xs:element minOccurs="1" maxOccurs="1" name="id"
                      type="xs:string">
                      <xs:annotation>
                        <xs:documentation>ID of the CWE</xs:documentation>
                      </xs:annotation>
                    </xs:element>
                    <xs:element minOccurs="1" maxOccurs="1" name="name"
                      type="xs:string">
                      <xs:annotation>
                        <xs:documentation>The quick name for the
                          CWE</xs:documentation>
                      </xs:annotation>
                    </xs:element>
                    <xs:element minOccurs="1" maxOccurs="1" name="description"
                      type="xs:string">
                      <xs:annotation>
                        <xs:documentation>A quick description of the
                          CWE</xs:documentation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                    </xs:annotation>
                  </xs:element>
                  <xs:element minOccurs="1" maxOccurs="1"
                    name="extendedDescription" type="xs:string">
                    <xs:annotation>
                      <xs:documentation>Description of the
                        CWE</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                  <xs:element minOccurs="0" maxOccurs="unbounded" name="id"
                    type="xs:string">
                    <xs:annotation>
                      <xs:documentation>List of sites referring to this
                        CWE</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                  <xs:element minOccurs="0" maxOccurs="unbounded" name="cweID"
                    type="xs:string">
                    <xs:annotation>
                      <xs:documentation>IDs of the CWEs associated to this
                        CWE</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="CPEcollection">
        <xs:annotation>
          <xs:documentation>MITRE Common Platform Enumeration (CPE)
            collection</xs:documentation>
        </xs:annotation>
        <xs:complexType>
          <xs:sequence>
            <xs:element name="CPE" maxOccurs="unbounded" minOccurs="0">
              <xs:annotation>
                <xs:documentation>Common Platform Enumeration</xs:documentation>
              </xs:annotation>
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="_id" type="xs:string">
                    <xs:annotation>
                      <xs:documentation>Uniquely identifies a CPE as an
                        UUID</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                  <xs:element minOccurs="1" maxOccurs="1" name="id"
                    type="xs:string">
                    <xs:annotation>
                      <xs:documentation>ID of the CPE in the official
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                sources</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="1" maxOccurs="1" name="title"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>Name of the CPE</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="0" maxOccurs="unbounded"
            name="references" type="xs:string">
            <xs:annotation>
                <xs:documentation>List of sites referring to this
                    CPE</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="1" maxOccurs="1" name="vendor"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>Name of the vendor linked to this
                    CPE</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="1" maxOccurs="1" name="product"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>Name of the product linked to this
                    CPE</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="1" maxOccurs="1" name="version"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>Version of the product to which this
                    CPE refers</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="1" maxOccurs="1" name="type"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>Type of the product to which this CPE
                    refers. For instance:
                    "application".</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="cpeID"
            type="xs:string">
            <xs:annotation>
                <xs:documentation>IDs of the CPEs associated to this
                    CPE</xs:documentation>
            </xs:annotation>
        </xs:element>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
          <xs:element name="CVSScollection">
            <xs:annotation>
              <xs:documentation>NIST Common Vulnerability Scoring System (CVSS)
                collection</xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:sequence>
                <xs:element maxOccurs="unbounded" minOccurs="0" name="CVSS">
                  <xs:annotation>
                    <xs:documentation>NIST Common Vulnerability Scoring System
                      (CVSS)</xs:documentation>
                  </xs:annotation>
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="cvssType" type="cvss:cvssType"/>
                      <xs:element name="cvssImpactType" type="cvss:cvssImpactType"/>
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:group>
      <xs:group name="RiskAssessmentEngine">
        <xs:annotation>
          <xs:documentation>Risk Assessment Engine data models</xs:documentation>
        </xs:annotation>
        <xs:sequence>
          <xs:element ref="dm:TargetInfrastructure"/>
          <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:RiskModel"/>
          <xs:element maxOccurs="unbounded" minOccurs="0" name="Indicator">
            <xs:annotation>
              <xs:documentation>Represents an indicator that can be used for risk
                assessment.</xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:sequence>
                <xs:element name="question" type="xs:string"/>
                <xs:element name="dataType" type="xs:string"/>
                <xs:element name="motivation" type="xs:string"/>
                <xs:element name="indicatorType">
                  <xs:annotation>
                    <xs:documentation>Represents an indicator type.</xs:documentation>
                  </xs:annotation>
                  <xs:complexType>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                <xs:attribute name="id" type="xs:string">
                  <xs:annotation>
                    <xs:documentation>Uniquely identifies an Indicator with an
                      UUID.</xs:documentation>
                  </xs:annotation>
                </xs:attribute>
                <xs:attribute name="name" type="xs:string"/>
              </xs:complexType>
            </xs:element>
            <xs:element name="means" type="xs:string"/>
            <xs:element name="Targets" minOccurs="0">
              <xs:complexType>
                <xs:sequence>
                  <xs:element maxOccurs="unbounded" minOccurs="0" name="targetID"
                  />
                </xs:sequence>
              </xs:complexType>
            </xs:element>
            <xs:element name="rule" type="xs:string"/>
            <xs:element maxOccurs="unbounded" minOccurs="0" name="indicatorValue">
              <xs:annotation>
                <xs:documentation>Represents a concrete value of an
                  indicator.</xs:documentation>
              </xs:annotation>
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="value" type="xs:string"/>
                  <xs:element name="timestamp" type="xs:dateTime"/>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="id" type="xs:string">
            <xs:annotation>
              <xs:documentation>Uniquely identifies an Indicator with an
                UUID.</xs:documentation>
            </xs:annotation>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
      <xs:element name="Event" maxOccurs="unbounded" minOccurs="0">
        <xs:complexType>
          <xs:attribute name="id" type="xs:string">
            <xs:annotation>
              <xs:documentation>Uniquely identifies an Event with an
                UUID.</xs:documentation>
            </xs:annotation>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
      <xs:element name="Alarm" maxOccurs="unbounded" minOccurs="0">
        <xs:complexType>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            <xs:sequence>
              <xs:element name="date" type="xs:dateTime"/>
              <xs:element name="srcIP" type="xs:string">
                <xs:annotation>
                  <xs:documentation>Source IP (where the attack is comming
                    from)</xs:documentation>
                </xs:annotation>
              </xs:element>
              <xs:element name="srcPort" type="xs:int">
                <xs:annotation>
                  <xs:documentation>Source Port (where the attack is comming
                    from)</xs:documentation>
                </xs:annotation>
              </xs:element>
              <xs:element name="dstIP" type="xs:string">
                <xs:annotation>
                  <xs:documentation>Destination IP (target afected)</xs:documentation>
                </xs:annotation>
              </xs:element>
              <xs:element name="dstPort" type="xs:int">
                <xs:annotation>
                  <xs:documentation>Destination Port (target
                    afected)</xs:documentation>
                </xs:annotation>
              </xs:element>
              <xs:element name="plugInID" type="xs:string">
                <xs:annotation>
                  <xs:documentation>Data source ID (XL-SIEM identification of the
                    attack)</xs:documentation>
                </xs:annotation>
              </xs:element>
              <xs:element name="pluginSID" type="xs:string">
                <xs:annotation>
                  <xs:documentation>Alarm ID (identification of the engine i.e.
                    XL-SIEM)</xs:documentation>
                </xs:annotation>
              </xs:element>
              <xs:element name="Risk" type="xs:string">
                <xs:annotation>
                  <xs:documentation>risk value evaluated by the plug-in, associated to
                    the raised alarm</xs:documentation>
                </xs:annotation>
              </xs:element>
              <xs:element name="Organization" type="xs:string">
                <xs:annotation>
                  <xs:documentation>ID of the Organization that owns the
                    sensor</xs:documentation>
                </xs:annotation>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="id" type="xs:string">
              <xs:annotation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            <xs:documentation>Uniquely identifies an Alarm with an
              UUID.</xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:complexType>
    </xs:element>
    <xs:element maxOccurs="unbounded" minOccurs="0" name="SpecificRiskModel">
      <xs:annotation>
        <xs:documentation>Represents a specific risk model (associated to a custom risk
          model that is only available to users of the same
          organization).</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="systemID" type="xs:string"/>
          <xs:element ref="dm:RiskModel"/>
        </xs:sequence>
        <xs:attribute name="id" type="xs:string">
          <xs:annotation>
            <xs:documentation>Uniquely identifies a Specific Risk Model with an
              UUID.</xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:complexType>
    </xs:element>
    <xs:element maxOccurs="unbounded" minOccurs="0" name="RiskPattern">
      <xs:annotation>
        <xs:documentation>Represents a risk pattern (associated to a generic risk model
          that is available to users of all organizations).</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="dm:RiskModel"/>
        </xs:sequence>
        <xs:attribute name="id" type="xs:string">
          <xs:annotation>
            <xs:documentation>Uniquely identifies a RiskPattern with an
              UUID.</xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:complexType>
    </xs:element>
    <xs:element maxOccurs="unbounded" minOccurs="0" name="RAE">
      <xs:annotation>
        <xs:documentation>Risk Assessment Engine </xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="shortDescription" type="xs:string" minOccurs="0">
            <xs:annotation>
              <xs:documentation>The RAE's short description</xs:documentation>
            </xs:annotation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
        </xs:element>
        <xs:element name="longDescription" type="xs:string" minOccurs="0">
          <xs:annotation>
            <xs:documentation>The RAE's long description</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="ipAddress" type="xs:string">
          <xs:annotation>
            <xs:documentation>The RAE's IP Address</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="port" type="xs:int">
          <xs:annotation>
            <xs:documentation>The RAE's port number</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="deleted" type="xs:boolean">
          <xs:annotation>
            <xs:documentation>Specifies if the RAE is deletted or
              not</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="systemID" type="xs:string">
          <xs:annotation>
            <xs:documentation>Provides the ID of the System this RAE belongs
              to</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="availability" type="xs:int">
          <xs:annotation>
            <xs:documentation>Provides the RAE's availability level (range
              0-10)</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="confidentiality" type="xs:int">
          <xs:annotation>
            <xs:documentation>Provides the RAE's confidentiality level (range
              0-10)</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="integrity" type="xs:int">
          <xs:annotation>
            <xs:documentation>Provides the RAE's integrity level (range
              0-10)</xs:documentation>
          </xs:annotation>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="id" type="xs:string">
        <xs:annotation>
          <xs:documentation>Uniquely identifies a Risk Assessment Engine with an
            UUID.</xs:documentation>
        </xs:annotation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
        </xs:attribute>
      </xs:complexType>
    </xs:element>
    <xs:element name="RiskReport" maxOccurs="unbounded" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Represents a result of the risk assessment
          procedure</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="qualitativeAssessment" type="xs:string"/>
          <xs:element name="AssetID" type="xs:string">
            <xs:annotation>
              <xs:documentation>Provides the ID of the System this RAE report was
                generated for</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="mitigationMeasures">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="mitigationMeasure" maxOccurs="unbounded"
                  minOccurs="0">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="shortDescription" type="xs:string"
                        minOccurs="0">
                        <xs:annotation>
                        <xs:documentation>The Mitigation Measure short
                        description</xs:documentation>
                        </xs:annotation>
                      </xs:element>
                      <xs:element name="longDescription" type="xs:string"
                        minOccurs="0">
                        <xs:annotation>
                        <xs:documentation>The Mitigation Measure long
                        description</xs:documentation>
                        </xs:annotation>
                      </xs:element>
                      <xs:element name="extendedDescription"
                        type="xs:string" minOccurs="0">
                        <xs:annotation>
                        <xs:documentation>The Mitigation Measure extended
                        description</xs:documentation>
                        </xs:annotation>
                      </xs:element>
                    </xs:sequence>
                    <xs:attribute name="id" type="xs:string"/>
                  </xs:complexType>
                </xs:element>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            <xs:element name="timestamp" type="xs:dateTime"/>
          </xs:sequence>
          <xs:attribute name="id" type="xs:string">
            <xs:annotation>
              <xs:documentation>Uniquely identifies a RAE's report with an
                UUID.</xs:documentation>
            </xs:annotation>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:group>
  <xs:element name="TargetInfrastructure">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="ip_address" type="xs:string"/>
        <xs:element name="port" type="xs:int"/>
        <xs:element name="shortDescription" type="xs:string" minOccurs="0">
          <xs:annotation>
            <xs:documentation>The Target Infrastructure short
              description</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="longDescription" type="xs:string" minOccurs="0">
          <xs:annotation>
            <xs:documentation>The Target Infrastructure long
              description</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="organization" type="xs:string">
          <xs:annotation>
            <xs:documentation>Identifier of the organization</xs:documentation>
          </xs:annotation>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="id" type="xs:string">
        <xs:annotation>
          <xs:documentation>Uniquely identifies the Target Infrastructure with an
            UUID.</xs:documentation>
        </xs:annotation>
      </xs:attribute>
    </xs:complexType>
  </xs:element>
  <xs:group name="CMDB">
    <xs:annotation>
      <xs:documentation>Configuration Management Database. CMDB is suitable for capturing any
        data, including security-relevant attributes</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element ref="dm:System"/>
      <xs:element name="AssetCategory">
        <xs:complexType>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            <xs:attribute name="id" type="xs:string" use="optional">
              <xs:annotation>
                <xs:documentation>Uniquely identifies an Asset Category as a
                  UUID</xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="name" type="xs:string" use="required">
              <xs:annotation>
                <xs:documentation>A human readable name which uniquely identifies the
                  IoT Asset Category.</xs:documentation>
              </xs:annotation>
            </xs:attribute>
          </xs:complexType>
        </xs:element>
        <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:Asset">
          <xs:annotation>
            <xs:documentation>A cyber, physical or cyber-physical element within an
              organization that is used in the frame of business
              operations</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element maxOccurs="unbounded" minOccurs="0" name="Vendors">
          <xs:annotation>
            <xs:documentation>Specifies an Asset Vendor. </xs:documentation>
          </xs:annotation>
          <xs:complexType>
            <xs:sequence>
              <xs:element maxOccurs="unbounded" minOccurs="1" name="Product">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element maxOccurs="unbounded" minOccurs="0" name="cpeID"
                      type="xs:string"/>
                  </xs:sequence>
                  <xs:attribute name="id" type="xs:string" use="optional">
                    <xs:annotation>
                      <xs:documentation>Uniquely identifies a Product as a
                        UUID</xs:documentation>
                    </xs:annotation>
                  </xs:attribute>
                  <xs:attribute name="name" type="xs:string" use="required">
                    <xs:annotation>
                      <xs:documentation>A human readable name which uniquely
                        identifies the Product.</xs:documentation>
                    </xs:annotation>
                  </xs:attribute>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute name="id" type="xs:string" use="optional">
              <xs:annotation>
                <xs:documentation>Uniquely identifies a Vendor as a
                  UUID</xs:documentation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            </xs:annotation>
          </xs:attribute>
          <xs:attribute name="name" type="xs:string" use="required">
            <xs:annotation>
              <xs:documentation>A human readable name which uniquely identifies the
                Vendor.</xs:documentation>
            </xs:annotation>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
      <xs:element maxOccurs="unbounded" minOccurs="0" name="Control">
        <xs:annotation>
          <xs:documentation>Control element to prevent attack impact. Note that a control
            element can be applied at the Vulnerability Level or at the Threat Level
            (never to both of them).</xs:documentation>
        </xs:annotation>
        <xs:complexType>
          <xs:choice>
            <xs:element name="ThreatBased">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="Mitigation">
                    <xs:complexType>
                      <xs:sequence maxOccurs="unbounded">
                        <xs:element name="level" type="xs:string"/>
                        <xs:element maxOccurs="1" name="attackID"
                          type="xs:string"/>
                      </xs:sequence>
                    </xs:complexType>
                  </xs:element>
                  <xs:element maxOccurs="unbounded" minOccurs="0"
                    name="assetCategoryID" type="xs:string">
                    <xs:annotation>
                      <xs:documentation>The Asset Categories this threat-based
                        Control can be applied to as mitigation
                        action.</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
            <xs:element name="VulnerabilityBased">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="Mitigation">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element maxOccurs="unbounded" name="cveID"
                          type="xs:string">
                          <xs:annotation>
                          <xs:documentation>The IDs of the CVEs (Common
                          Vulnerabilities and Exposures) that this
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                                Vulnerability Control can be applied to as
                                mitigation action.</xs:documentation>
                                </xs:annotation>
                              </xs:element>
                            </xs:sequence>
                          </xs:complexType>
                        </xs:element>
                      </xs:sequence>
                    </xs:complexType>
                  </xs:element>
                </xs:choice>
                <xs:attribute name="id" type="xs:string" use="optional">
                  <xs:annotation>
                    <xs:documentation>Uniquely identifies a Controll as a
                      UUID</xs:documentation>
                  </xs:annotation>
                </xs:attribute>
                <xs:attribute name="name" type="xs:string" use="required">
                  <xs:annotation>
                    <xs:documentation>A human readable name which uniquely identifies the
                      Control.</xs:documentation>
                  </xs:annotation>
                </xs:attribute>
              </xs:complexType>
            </xs:element>
            <xs:element name="MitigationPlan">
              <xs:complexType>
                <xs:sequence>
                  <xs:element maxOccurs="unbounded" name="ControlEnforcement">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="controlID" type="xs:string"/>
                        <xs:element name="assetID" type="xs:string"/>
                      </xs:sequence>
                    </xs:complexType>
                  </xs:element>
                </xs:sequence>
                <xs:attribute name="id" type="xs:string"/>
                <xs:attribute name="name" type="xs:string"/>
              </xs:complexType>
            </xs:element>
            <xs:element name="mitigationMeasures">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="mitigationMeasure" maxOccurs="unbounded" minOccurs="0">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="shortDescription" type="xs:string"
                          minOccurs="0">
                          <xs:annotation>
                            <xs:documentation>The Mitigation Measure short
                              description</xs:documentation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                    </xs:annotation>
                  </xs:element>
                  <xs:element name="longDescription" type="xs:string"
                    minOccurs="0">
                    <xs:annotation>
                      <xs:documentation>The Mitigation Measure long
                        description</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                  <xs:element name="extendedDescription" type="xs:string"
                    minOccurs="0">
                    <xs:annotation>
                      <xs:documentation>The Mitigation Measure extended
                        description</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                </xs:sequence>
                <xs:attribute name="id" type="xs:string"/>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element ref="dm:Vulnerabilities"/>
      <xs:element name="AttackScenarios">
        <xs:annotation>
          <xs:documentation>Use Case Data</xs:documentation>
        </xs:annotation>
        <xs:complexType>
          <xs:sequence>
            <xs:element name="AbuseCase" maxOccurs="unbounded" minOccurs="0">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="AssetApplicability">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element minOccurs="0" name="assetID"
                          type="xs:anyURI" maxOccurs="unbounded">
                          <xs:annotation>
                          <xs:documentation>The reference ID of an
                          Asset</xs:documentation>
                          </xs:annotation>
                        </xs:element>
                        <xs:element minOccurs="0" name="assetCategoryID"
                          type="xs:anyURI" maxOccurs="unbounded">
                          <xs:annotation>
                          <xs:documentation>The reference ID of an
                          Asset</xs:documentation>
                          </xs:annotation>
                        </xs:element>
                      </xs:sequence>
                    </xs:complexType>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                    </xs:element>
                    <xs:element name="vulnerabilityID" type="xs:string">
                      <xs:annotation>
                        <xs:documentation>The ID of the Vulnerability this Abuse
                          Case is monitoring.</xs:documentation>
                      </xs:annotation>
                    </xs:element>
                    <xs:element name="attackID" type="xs:string">
                      <xs:annotation>
                        <xs:documentation>CAPEC MITRE attack
                          Identifyer</xs:documentation>
                      </xs:annotation>
                    </xs:element>
                  </xs:sequence>
                  <xs:attribute name="id" type="xs:string">
                    <xs:annotation>
                      <xs:documentation>Uniquely identifies an abuse Use Case as a
                        UUID</xs:documentation>
                    </xs:annotation>
                  </xs:attribute>
                  <xs:attribute name="name" type="xs:string" use="required">
                    <xs:annotation>
                      <xs:documentation>A human readable name which uniquely
                        identifies the abuse Use Case.</xs:documentation>
                    </xs:annotation>
                  </xs:attribute>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:group>
    <xs:group name="SLA">
      <xs:annotation>
        <xs:documentation>The Service Level Agreement Group.</xs:documentation>
      </xs:annotation>
      <xs:sequence>
        <xs:element maxOccurs="unbounded" minOccurs="0" name="SLADefinition">
          <xs:annotation>
            <xs:documentation>Defines an SLA Type.</xs:documentation>
          </xs:annotation>
          <xs:complexType>
            <xs:sequence>
              <xs:element name="description" type="xs:string">
                <xs:annotation>
                  <xs:documentation>Textual description of the SLA Type.</xs:documentation>
                </xs:annotation>
              </xs:element>
              <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:Parameter"/>
              <xs:element ref="dm:AdditionalInformation"/>
            </xs:sequence>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            <xs:attribute name="id" type="xs:string" use="optional">
              <xs:annotation>
                <xs:documentation>Uniquely identifies an SLA Type with an UUID.</xs:documentation>
              </xs:annotation>
            </xs:attribute>
            <xs:attribute name="name" type="xs:string" use="required">
              <xs:annotation>
                <xs:documentation>A human readable name which uniquely identifies the SLA
                                  Type.</xs:documentation>
              </xs:annotation>
            </xs:attribute>
          </xs:complexType>
        </xs:element>
        <xs:element maxOccurs="unbounded" minOccurs="0" name="SLA">
          <xs:annotation>
            <xs:documentation>Defines an SLA runnable Object.</xs:documentation>
          </xs:annotation>
          <xs:complexType>
            <xs:sequence>
              <xs:element name="description" minOccurs="0" type="xs:string">
                <xs:annotation>
                  <xs:documentation>Textual description of the SLA.</xs:documentation>
                </xs:annotation>
              </xs:element>
              <xs:element name="slaDefinitionID" type="xs:string">
                <xs:annotation>
                  <xs:documentation>The SLA Type this SLA belongs to.</xs:documentation>
                </xs:annotation>
              </xs:element>
              <xs:element name="DataFeed">
                <xs:annotation>
                  <xs:documentation>The list of data sources the Analytics algorithm is using as
                                    input.</xs:documentation>
                </xs:annotation>
                <xs:complexType>
                  <xs:sequence>
                    <xs:element ref="dm:digitalResource"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
              <xs:element maxOccurs="1" name="DataSink">
                <xs:annotation>
                  <xs:documentation>The list of the Digital Resources the Processor is forwarding the data
                                    to.</xs:documentation>
                </xs:annotation>
                <xs:complexType>
                  <xs:sequence>
                    <xs:element ref="dm:digitalResource"/>
                  </xs:sequence>
                </xs:complexType>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            </xs:element>
            <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:ParameterValue">
              <xs:annotation>
                <xs:documentation>Parameter Value for the execution of the SLA specified in the SLA

                    Type.</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element ref="dm:AdditionalInformation"/>
          </xs:sequence>
          <xs:attribute name="id" type="xs:string" use="optional">
            <xs:annotation>
              <xs:documentation>Uniquely identifies an SLA with an UUID.</xs:documentation>
            </xs:annotation>
          </xs:attribute>
          <xs:attribute name="name" type="xs:string" use="required">
            <xs:annotation>
              <xs:documentation>A human readable name which uniquely identifies the

                  SLA.</xs:documentation>
            </xs:annotation>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
      <xs:element maxOccurs="unbounded" minOccurs="0" name="ComplexSLA">
        <xs:annotation>
          <xs:documentation>Identifies an SLA collection which is comprising an SLA complex

              Object.</xs:documentation>
        </xs:annotation>
        <xs:complexType>
          <xs:sequence>
            <xs:element maxOccurs="unbounded" minOccurs="0" name="slaID"
              type="xs:string">
              <xs:annotation>
                <xs:documentation>SLA ID which participates in the definition of a Complex

                    SLA.</xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="id" type="xs:string" use="optional">
            <xs:annotation>
              <xs:documentation>Uniquely identifies a Complex SLA with an UUID.</xs:documentation>
            </xs:annotation>
          </xs:attribute>
          <xs:attribute name="name" type="xs:string" use="required">
            <xs:annotation>
              <xs:documentation>A human readable name which uniquely identifies the Complex

                  SLA.</xs:documentation>
            </xs:annotation>
          </xs:attribute>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
      </xs:complexType>
    </xs:element>
    <xs:element maxOccurs="unbounded" minOccurs="0" name="SLAReport">
      <xs:annotation>
        <xs:documentation>Provides an SLA report structure.</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element maxOccurs="unbounded" minOccurs="0" name="ProbeSLA">
            <xs:annotation>
              <xs:documentation>Probe associated with the SLA</xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:sequence>
                <xs:element name="probeTypeID" type="xs:string" maxOccurs="1"
                  minOccurs="0">
                  <xs:annotation>
                    <xs:documentation>The Type of the probe associated with the

                        SLA.</xs:documentation>
                  </xs:annotation>
                </xs:element>
                <xs:element minOccurs="0" name="probeID" type="xs:string">
                  <xs:annotation>
                    <xs:documentation>The ID of the probe associated with the SLA.</xs:documentation>
                  </xs:annotation>
                </xs:element>
                <xs:element minOccurs="0" name="dataVolume" type="xs:float">
                  <xs:annotation>
                    <xs:documentation>The data volume provided by the probe, accounted as bytes in the

                        case of streaming data and/or as the number of reports in case of batch

                        data.</xs:documentation>
                  </xs:annotation>
                </xs:element>
                <xs:element minOccurs="0" name="dataDuration" type="xs:float">
                  <xs:annotation>
                    <xs:documentation>The duration for which data are provided by the probe from the

                        SECaaS Customer to the SECaaS provider.</xs:documentation>
                  </xs:annotation>
                </xs:element>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
          <xs:element maxOccurs="unbounded" minOccurs="0" name="SECaaSSLA">
            <xs:annotation>
              <xs:documentation>SECaaS associated with the SLA.</xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:sequence>
                <xs:element minOccurs="0" name="dataVolume" type="xs:float">
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                    <xs:annotation>
                      <xs:documentation>The data volume provided by the SECaaS, accounted as bytes in the
                          case of streaming data and/or as the number of reports in case of
                          batch data.</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                  <xs:element minOccurs="0" name="reportsNumber" type="xs:int">
                    <xs:annotation>
                      <xs:documentation>The Number of Reports provided by the
                          SECaaS.</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                  <xs:element maxOccurs="1" minOccurs="0" name="reportFrequency"
                    type="xs:string">
                    <xs:annotation>
                      <xs:documentation>The frequency at which the SECaaS provider provides the reports to
                          the SECaaS customer.</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                  <xs:element maxOccurs="1" minOccurs="0" name="dataDuration"
                    type="xs:duration">
                    <xs:annotation>
                      <xs:documentation>The duration for which data are provided by the SECaaS
                          provider.</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="id" type="xs:string" use="optional">
            <xs:annotation>
              <xs:documentation>Uniquely identifies an SLA Report with an UUID.</xs:documentation>
            </xs:annotation>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:group>
  <xs:element name="Observation">
    <xs:annotation>
      <xs:documentation>A generic type of Data Source Live measurements (physical or
        virtual)</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="dataSourceID" type="xs:anyURI">
          <xs:annotation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            <xs:documentation>The ID of the Probe (physical or virtual) these
              observations refaire to.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="systemID" type="xs:string"/>
        <xs:element name="quantityKindID" type="xs:anyURI">
          <xs:annotation>
            <xs:documentation>The ID of the Data Kind this observation refaire
              to.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element minOccurs="0" name="timestamp" type="xs:dateTime">
          <xs:annotation>
            <xs:documentation>Timestamp is the dateTime of when an observation was
              captured.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element ref="dm:Location" minOccurs="0">
          <xs:annotation>
            <xs:documentation>The location at which the measurement was taken if
              SecureIoT Probe mobile attribute is true</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="value" type="xs:anyType" minOccurs="1">
          <xs:annotation>
            <xs:documentation>Element providing the value of a
              measurement.</xs:documentation>
          </xs:annotation>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="id" type="xs:string">
        <xs:annotation>
          <xs:documentation>Uniquely identifies an Observation with a
            UUID</xs:documentation>
        </xs:annotation>
      </xs:attribute>
    </xs:complexType>
  </xs:element>
  <xs:element name="System">
    <xs:annotation>
      <xs:documentation>An observed IoT System characteristics</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="description" minOccurs="1" type="xs:string">
          <xs:annotation>
            <xs:documentation>Textual description for the IoT System.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element minOccurs="0" ref="dm:Location">
          <xs:annotation>
            <xs:documentation>The System's location Physical or
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                Virtual</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="organization" type="xs:string" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Identifier of the organization</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="additionalInformation" type="xs:anyType" maxOccurs="unbounded"
            minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="optional">
        <xs:annotation>
            <xs:documentation>Uniquely identifies a System as a UUID</xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="name" type="xs:string" use="required">
        <xs:annotation>
            <xs:documentation>A human readable name which uniquely identifies the IoT
                System.</xs:documentation>
        </xs:annotation>
    </xs:attribute>
  </xs:complexType>
</xs:element>
<xs:element name="Asset">
  <xs:complexType>
    <xs:sequence>
        <xs:element name="systemID" type="xs:string">
            <xs:annotation>
                <xs:documentation>The ID of the IoT System this Asset belongs
                    to</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="0" name="assetCategoryID" type="xs:string">
            <xs:annotation>
                <xs:documentation>The ID of the Asset Category the Asset belongs
                    to.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="0" name="productName" type="xs:string">
            <xs:annotation>
                <xs:documentation>The Asset given name from the Vendor.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="description" minOccurs="0" type="xs:string">
            <xs:annotation>
                <xs:documentation>Textual description for the Asset.</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="longDescription" type="xs:string" minOccurs="0">
            <xs:annotation>
                <xs:documentation>Long textual description for the Asset.</xs:documentation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
      </xs:annotation>
    </xs:element>
    <xs:element minOccurs="0" name="instalationDate" type="xs:dateTime">
      <xs:annotation>
        <xs:documentation>The date which the Asset was installed.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element minOccurs="0" name="inventoryDate " type="xs:dateTime">
      <xs:annotation>
        <xs:documentation>The last date the data/information of the Asset has been

            revisited.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element minOccurs="0" ref="dm:Location">
      <xs:annotation>
        <xs:documentation>Provides the Assets's Physical or Virtual
          location</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element maxOccurs="unbounded" name="relantionship" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Specifies the relationship with other
          Assets</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="RelantionshipType">
            <xs:annotation>
              <xs:documentation>Specifies the relantionship
                type</xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:choice>
                <xs:element name="isInstalledOn"/>
                <xs:element name="isLocatedIn"/>
                <xs:element name="isConnectedTo"/>
              </xs:choice>
            </xs:complexType>
          </xs:element>
          <xs:element name="AssetID" type="xs:string">
            <xs:annotation>
              <xs:documentation>The ID of the Asset this Asset is related
                with.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="AssetNature">
      <xs:annotation>
        <xs:documentation>Specifies the nature of the Asset (Tangible or
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            Intangible)</xs:documentation>
        </xs:annotation>
        <xs:complexType>
          <xs:choice>
            <xs:element name="tangible">
              <xs:complexType>
                <xs:choice>
                  <xs:element name="physical"/>
                  <xs:element name="syberphysical"/>
                </xs:choice>
              </xs:complexType>
            </xs:element>
            <xs:element name="intangible"/>
          </xs:choice>
        </xs:complexType>
      </xs:element>
      <xs:element maxOccurs="1" minOccurs="0" name="Patches">
        <xs:annotation>
          <xs:documentation>A list of patches applied to this asset.</xs:documentation>
        </xs:annotation>
        <xs:simpleType>
          <xs:list itemType="dm:patch"/>
        </xs:simpleType>
      </xs:element>
      <xs:element name="ProbeIDs" maxOccurs="1" minOccurs="0">
        <xs:annotation>
          <xs:documentation>The ID of the Probe deployed to monitor this
            asset</xs:documentation>
        </xs:annotation>
        <xs:simpleType>
          <xs:list itemType="dm:probeID"/>
        </xs:simpleType>
      </xs:element>
      <xs:element minOccurs="0" name="businessValue" type="xs:int"/>
      <xs:element minOccurs="0" name="cpeID" type="xs:string">
        <xs:annotation>
          <xs:documentation>Provides the Common Platform Enumeration (CPE) ID that
            describes this Asset.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="appliedControls" minOccurs="0">
        <xs:complexType>
          <xs:sequence>
            <xs:element maxOccurs="unbounded" minOccurs="0" name="controlID"
              type="xs:string"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="KvAttribute" maxOccurs="unbounded" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Provide adittional parameters for describing this Asset.
            Among the parameters the "ip_adress" and "port" number should be
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                provided where applicable.</xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:sequence>
                <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:ParameterValue"
                />
              </xs:sequence>
              <xs:attribute name="name" type="xs:string">
                <xs:annotation>
                  <xs:documentation>The name of the KvAttribute group these parameter Values belong to

                    (i.e. network, general,...).</xs:documentation>
                </xs:annotation>
              </xs:attribute>
            </xs:complexType>
          </xs:element>
          <xs:element name="additionalInformation" type="xs:anyType" maxOccurs="unbounded"
            minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="id" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation>Uniquely identifies an Asset as a UUID</xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="name" type="xs:string" use="required">
          <xs:annotation>
            <xs:documentation>A human readable name which uniquely identifies the IoT
              Asset.</xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="patch">
      <xs:restriction base="xs:string"/>
    </xs:simpleType>
    <xs:simpleType name="probeID">
      <xs:restriction base="xs:string"/>
    </xs:simpleType>
    <xs:element name="Vulnerabilities">
      <xs:annotation>
        <xs:documentation>Vulnerabilities is used to extend the MITRE Common Vulnerabilities and Exposures

          (CVE) collection.</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="Vulnerability">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="CalculatedSubscore">
                  <xs:complexType>
                    <xs:sequence>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            <xs:element name="exploitability" type="xs:float"/>
            <xs:element name="impact" type="xs:float"/>
            <xs:element name="CVSS" type="xs:float"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="accessVector">
        <xs:annotation>
          <xs:documentation>Choose for Access Vector one of: "Local",
            "AdjucentNetwork", "Network"</xs:documentation>
        </xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Local"/>
            <xs:enumeration value="AdjucentNetwork"/>
            <xs:enumeration value="Network"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="accessComplexity">
        <xs:annotation>
          <xs:documentation>Choose for Access Complexity one of: "Low",
            "Medium", "High"</xs:documentation>
        </xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Low"/>
            <xs:enumeration value="Medium"/>
            <xs:enumeration value="High"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="authentication">
        <xs:annotation>
          <xs:documentation>Choose for Authentication one of: "None",
            "Single_Factor_Authenticated",
            "Multy_Factor_Authenticated"</xs:documentation>
        </xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="None"/>
            <xs:enumeration value="Single_Factor_Authenticated"/>
            <xs:enumeration value="Multy_Factor_Authenticated"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="confidentialityImpact">
        <xs:annotation>
          <xs:documentation>Choose for Confidentiality Impact one of:
            "None", "Partial", "Complete"</xs:documentation>
        </xs:annotation>
        <xs:simpleType>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
                    <xs:restriction base="xs:string">
                      <xs:enumeration value="None"/>
                      <xs:enumeration value="Partial"/>
                      <xs:enumeration value="Complete"/>
                    </xs:restriction>
                  </xs:simpleType>
                </xs:element>
                <xs:element name="integrityImpact">
                  <xs:annotation>
                    <xs:documentation>Choose for Integrity Impact one of: "None",
                      "Partial", "Complete"</xs:documentation>
                  </xs:annotation>
                  <xs:simpleType>
                    <xs:restriction base="xs:string">
                      <xs:enumeration value="None"/>
                      <xs:enumeration value="Partial"/>
                      <xs:enumeration value="Complete"/>
                    </xs:restriction>
                  </xs:simpleType>
                </xs:element>
                <xs:element name="availabilityImpact">
                  <xs:annotation>
                    <xs:documentation>Choose for Impact Subscore one of: "None",
                      "Partial", "Complete"</xs:documentation>
                  </xs:annotation>
                  <xs:simpleType>
                    <xs:restriction base="xs:string">
                      <xs:enumeration value="None"/>
                      <xs:enumeration value="Partial"/>
                      <xs:enumeration value="Complete"/>
                    </xs:restriction>
                  </xs:simpleType>
                </xs:element>
                <xs:element name="AffectedAssetCategory">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element maxOccurs="unbounded" minOccurs="0"
                        name="assetCategoryID" type="xs:string"/>
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:sequence>
              <xs:attribute name="cveID" type="xs:string" use="required"/>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="RiskModel">
      <xs:annotation>
        <xs:documentation>Represents a risk model</xs:documentation>
      </xs:annotation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
    <xs:complexType>
      <xs:sequence>
        <xs:element name="description" type="xs:string"/>
        <xs:element name="ModelType" type="xs:string">
          <xs:annotation>
            <xs:documentation>Specifies the model type. i.e. dexi_model, r_model,
              coras_model </xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="modelScript" type="xs:string"/>
        <xs:element name="indicators">
          <xs:complexType>
            <xs:sequence>
              <xs:element maxOccurs="unbounded" minOccurs="0" name="indicatorID"
                type="xs:string">
                <xs:annotation>
                  <xs:documentation>References an indicator that can be used for
                    risk assessment.</xs:documentation>
                </xs:annotation>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="risks">
          <xs:complexType>
            <xs:sequence>
              <xs:element maxOccurs="unbounded" minOccurs="0" name="riskID"
                type="xs:string"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="id" type="xs:string">
        <xs:annotation>
          <xs:documentation>Uniquely identifies a RiskModel with an
            UUID.</xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="name" type="xs:string"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="QuantityKind">
    <xs:annotation>
      <xs:documentation>A QuantityKind is an abstract classifier that represents the concept
        of "kind of quantity". A QuantityKind represents the essence of a quantity without
        any numerical value or unit. (e.g. A sensor -sensor1- measures temperature: sensor1
        has quantityKind temperature) </xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="description" type="xs:string"/>
      </xs:sequence>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
        <xs:attribute name="id" type="xs:string">
          <xs:annotation>
            <xs:documentation>Uniquely identifies a Quantity Kind with an
              UUID</xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="name" type="xs:string"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="DataSource">
      <xs:annotation>
        <xs:documentation>DataSource provides the characteristics and connection interface of a
          data source</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="description" type="xs:string"/>
          <xs:element name="systemID" type="xs:string"/>
          <xs:element ref="dm:Location"/>
          <xs:element ref="dm:ConnectionInterface"/>
          <xs:element name="characteristics" type="xs:anyType"/>
          <xs:element name="additionalInformation" type="xs:anyType"/>
        </xs:sequence>
        <xs:attribute name="id" type="xs:string">
          <xs:annotation>
            <xs:documentation>Uniquely identifies a Data Source with an
              UUID.</xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="name" type="xs:string"/>
      </xs:complexType>
    </xs:element>

    <xs:element name="DataDestination">
      <xs:annotation>
        <xs:documentation>DataDestination provides the characteristics and connection interface
          of a data destination</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="description" type="xs:string"/>
          <xs:element ref="dm:ConnectionInterface"/>
          <xs:element name="characteristics" type="xs:anyType"/>
          <xs:element name="additionalInformation" type="xs:anyType"/>
        </xs:sequence>
        <xs:attribute name="id" type="xs:string">
          <xs:annotation>
            <xs:documentation>Uniquely identifies a Data Destination with an
              UUID</xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="name" type="xs:string"/>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
    </xs:complexType>
  </xs:element>
  <xs:element name="ConnectionInterface">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="connectionInterfaceTypeID" type="xs:string"/>
        <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:ParameterValue"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ConnectionInterfaceType">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="description" type="xs:string"/>
        <xs:element ref="dm:Parameter" maxOccurs="unbounded" minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="id" type="xs:string">
        <xs:annotation>
          <xs:documentation>Uniquely identifies a ConnectionInterfaceType with an
            UUID</xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="name" type="xs:string"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="ParameterValue">
    <xs:complexType>
      <xs:all>
        <xs:element type="xs:string" name="name"/>
        <xs:element type="xs:anyType" name="value"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <xs:element name="Parameter">
    <xs:annotation>
      <xs:documentation>Parameter Specification</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="description" type="xs:string" minOccurs="1">
          <xs:annotation>
            <xs:documentation>Element describing the value of an
              attribute.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element minOccurs="1" name="type" type="xs:string">
          <xs:annotation>
            <xs:documentation>A classification of data (i.e. short, int, float, boolean,
              ...)</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="defaultValue" type="xs:anyType" minOccurs="0">
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            <xs:annotation>
              <xs:documentation>A predefined default value for an
                attribute.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:sequence>
        <xs:attribute name="name" type="xs:string"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="Location">
      <xs:annotation>
        <xs:documentation>Indicating the location at which something took
          place</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:choice>
          <xs:element maxOccurs="1" minOccurs="0" name="geoLocation">
            <xs:annotation>
              <xs:documentation>Specifying a physical location (a pair of coordinates)
              </xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:all>
                <xs:element name="latitude" type="xs:string"/>
                <xs:element name="longitude" type="xs:string"/>
              </xs:all>
            </xs:complexType>
          </xs:element>
          <xs:element minOccurs="0" name="virtualLocation" type="xs:anyURI">
            <xs:annotation>
              <xs:documentation>Specifying a virtual location (it could be the ID of a
                resource or subsystem)</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:choice>
      </xs:complexType>
    </xs:element>
    <xs:element name="ProbeType">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="description" type="xs:string"/>
          <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:Parameter"/>
        </xs:sequence>
        <xs:attribute name="id" type="xs:string">
          <xs:annotation>
            <xs:documentation>Uniquely identifies a ProbeType with an
              UUID</xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="name" type="xs:string"/>
      </xs:complexType>
    </xs:element>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
<xs:element name="Probe">
  <xs:annotation>
    <xs:documentation>The SecureIoT Probe</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="description" minOccurs="0" type="xs:string">
        <xs:annotation>
          <xs:documentation>Textual description for the SecureIoT
            Probe</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="dataSourceID" type="xs:anyURI">
        <xs:annotation>
          <xs:documentation>The ID of the Data Source this probe is collecting data.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element maxOccurs="unbounded" name="dataDestinationID" type="xs:anyURI">
        <xs:annotation>
          <xs:documentation>The ID of the Data Destination(s) the Probe is forwarding
            the data</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="quantityKindID" type="xs:string"/>
      <xs:element name="probeTypeID" type="xs:string"/>
      <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:ParameterValue"/>
      <xs:element name="additionalInformation" type="xs:anyType"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="optional">
      <xs:annotation>
        <xs:documentation>Uniquely identifies a Probe with an UUID</xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="name" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="PO">
  <xs:annotation>
    <xs:documentation>Processor Orchestrator</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="DataProcessors">
        <xs:complexType>
          <xs:sequence>
            <xs:element maxOccurs="unbounded" ref="dm:PM"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string">
      <xs:annotation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            <xs:documentation>Uniquely identifies a Processor Orchestrator with an UUID.</xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="name" type="xs:string"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="PM">
      <xs:annotation>
        <xs:documentation>Processor Manifest</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="processorDefinitionReferenceID" type="xs:anyURI">
            <xs:annotation>
              <xs:documentation>The Reference ID of the Processor Definition(PD) this PM
                is using.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="description" minOccurs="0" type="xs:string">
            <xs:annotation>
              <xs:documentation>Textual description for the Processor
                Manifest</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="DataFeed">
            <xs:annotation>
              <xs:documentation>The list of data sources the Analytics algorithm is using as

                                input.</xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:sequence>
                <xs:element ref="dm:digitalResource" maxOccurs="unbounded" minOccurs="0"
                />
              </xs:sequence>
            </xs:complexType>
          </xs:element>
          <xs:element maxOccurs="1" name="DataSink">
            <xs:annotation>
              <xs:documentation>The list of the Digital Resources the Processor is forwarding the data

                              to.</xs:documentation>
            </xs:annotation>
            <xs:complexType>
              <xs:sequence>
                <xs:element ref="dm:digitalResource" maxOccurs="unbounded" minOccurs="0"
                />
              </xs:sequence>
            </xs:complexType>
          </xs:element>
          <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:ParameterValue"/>
          <xs:element name="additionalInformation" type="xs:anyType" maxOccurs="unbounded"
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            minOccurs="0"/>

        </xs:sequence>
        <xs:attribute name="id" type="xs:string" use="optional">
          <xs:annotation>
            <xs:documentation>Uniquely identifies a Processor Manifest with an UUID.</xs:documentation>
          </xs:annotation>
        </xs:attribute>
        <xs:attribute name="name" type="xs:string" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="digitalResource">
      <xs:complexType>
        <xs:all>
          <xs:element name="dataSourceID" type="xs:anyURI" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>The data source ID which comprises this Digital Resource.</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="quantityKindID" type="xs:string">
            <xs:annotation>
              <xs:documentation>Specifies the Measurement kind of the Data Source with the help of the Quantity

                                Kind entity.</xs:documentation>
            </xs:annotation>
          </xs:element>
        </xs:all>
      </xs:complexType>
    </xs:element>
    <xs:element name="PD">
      <xs:annotation>
        <xs:documentation>Processor Definition</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element minOccurs="1" maxOccurs="1" name="processorType" type="xs:string">
            <xs:annotation>
              <xs:documentation>The type of the Processor</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="version" type="xs:string" minOccurs="0">
            <xs:annotation>
              <xs:documentation>The version of the processor type</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="copyright" type="xs:string" maxOccurs="1" minOccurs="0">
            <xs:annotation>
              <xs:documentation>Processor Owner</xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element minOccurs="0" maxOccurs="1" name="description" type="xs:string">
            <xs:annotation>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
            <xs:documentation>Deacription of the Processor</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element minOccurs="1" maxOccurs="1" name="processorLocation" type="xs:string">
          <xs:annotation>
            <xs:documentation>The physical or relative location of the processor
              library</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element minOccurs="0" maxOccurs="1" ref="dm:Parameters"/>
        <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:AdditionalInformation"/>
      </xs:sequence>
      <xs:attribute name="id" type="xs:anyURI" use="optional">
        <xs:annotation>
          <xs:documentation>The Processor ID</xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="name" type="xs:string">
        <xs:annotation>
          <xs:documentation>The processor name</xs:documentation>
        </xs:annotation>
      </xs:attribute>
    </xs:complexType>
  </xs:element>
  <xs:element name="SecurityTemplate">
    <xs:annotation>
      <xs:documentation>An Algorithm trained Machine Learning Security Model.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="SecurityTemplateScript" type="xs:anyType"/>
      </xs:sequence>
      <xs:attribute name="id" type="xs:string">
        <xs:annotation>
          <xs:documentation>Uniquely identifies a Security Template with an UUID.</xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="name" type="xs:string">
        <xs:annotation>
          <xs:documentation>A human readable form of the Security Template name</xs:documentation>
        </xs:annotation>
      </xs:attribute>
    </xs:complexType>
  </xs:element>
  <xs:element name="Parameters">
    <xs:annotation>
      <xs:documentation> Processor list of required parameters</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element maxOccurs="unbounded" minOccurs="0" ref="dm:Parameter"/>
      </xs:sequence>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
    </xs:complexType>
  </xs:element>
  <xs:element name="AdditionalInformation" type="xs:anyType">
    <xs:annotation>
      <xs:documentation>Optional auxiliary field that may contain any additional
        information</xs:documentation>
    </xs:annotation>
  </xs:element>

  <xs:element name="VulnerabilityProfile">
    <xs:annotation>
      <xs:documentation>Provides the Asset's Vulnerability Profile.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="cpeID" type="xs:string" minOccurs="0" maxOccurs="unbounded">
          <xs:annotation>
            <xs:documentation>The IDs of the CPEs (Common Platform
                Enumeration) associated to this asset.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="cveID" type="xs:string" minOccurs="0" maxOccurs="unbounded">
          <xs:annotation>
            <xs:documentation>The IDs of the CVEs (Common Vulnerabilities
                and Exposures) that can be associated to this asset. A
                vulnerability is characterized by two properties;
                exploitability, i.e. how easily you can make use of the
                vulnerability and impact i.e. how dangerous is the
                exploitation of this vulnerability. The model defines
                sub-scores for Confidentiality, Integrity and Availability
                (a.k.a. CIA) consequences. An open repository for disclosed
                vulnerabilities can be found here:
                https://www.cvedetails.com</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="capecID" type="xs:string" minOccurs="0" maxOccurs="unbounded">
          <xs:annotation>
            <xs:documentation>The IDs of the CAPECs (Common Attack Pattern
                Enumeration and Classification) that can be associated to
                this asset. Note that in the cyber security domain, the
                terms Attack and Threat coincide.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="cweID" type="xs:string" minOccurs="0" maxOccurs="unbounded">
          <xs:annotation>
            <xs:documentation>The IDs of the CWEs (Common Weakness
                Enumeration) that can be associated to this
                asset.</xs:documentation>
          </xs:annotation>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.

```
    </xs:element>
</xs:schema>
```


Project Title: SecureIoT
Contract No. 779899
Project Coordinator: INTRASOFT International S.A.