Deliverable

D3.10 Multi Layered security model specification

Project Acronym:	DUET	
Project title:	Digital Urban European Twins	
Grant Agreement No.	870697	
Website:	www.digitalurbantwins.eu	
Version:	1.0	
Date:	30 November 2020	
Responsible Partner:	AEG	
Contributing Partners:	IMEC, ATC	
Reviewers:	Gert Vervaet (AIV) Thanasis Dalianis (ATC) Philippe Michiels (IMEC) *External* Yannis Charalabidis	
Dissemination Level:	Public	
	Confidential – only consortium members and European Commission	X

This project has received financial support from the European Union's Horizon 2020 Programme under grant agreement no. 870697

Revision History

Revision	Date	Author	Organization	Description
0.1	01.10.2020	Leonidas Kallipolitis	AEG	Initial structure
0.2	06.11.2020	Leonidas Kallipolitis, Stathis Dimakos, Andreas Alexopoulos Kostis Michalis	AEG	First Draft
0.3	18.11.2020	Leonidas Kallipolitis, Stathis Dimakos, Andreas Alexopoulos, Kostis Michalis	AEG	v0.1 Ready for Internal Review
0.4	27.11.2020	Gert Vervaet, Thanasis Dalianis, Philippe Michiels, Yannis Charalabidis	AIV, ATC, IMEC, Un. Aegean	Internal Review Comments
1.0	30.11.2020	Leonidas Kallipolitis, Stathis Dimakos, Andreas Alexopoulos, Kostis Michalis	AEG	Final version after integrating review comments

Table of Contents

Tables

Figures

Executive Summary

This report identifies the security and privacy implications and concerns stemming from the activities required to increase a city's performance and growth regarding smart technologies. A Threat Taxonomy pertaining to Digital Twin and Smart City technologies is defined and the required security requirements to address the threats are identified. Confidentiality, Integrity, Availability, Accountability and Authenticity are the key aspects which are covered via the defined security requirements, tackling both development and run-time phases of DUET.

We then present a detailed set of protective measures and good practices that should be followed in order to meet the security requirements. These are reported as Technical Controls (TCs) accompanied by a mapping between these controls and requirements so as to directly show their relation.

Finally, we describe the actual implementations or concrete plans to realise the security measures and the designed privacy mechanisms within DUET. A clear matching of TCs and respective processes, development practises and software components is provided, thus presenting the current status of DUET's Multi-Layered Security and Privacy mechanisms.

1. Introduction

DUET's main objective is to provide the mechanisms that will allow increased visibility into a city's interactions among infrastructure, citizens and technological assets without jeopardising security and privacy. Real-time IoT measurements, spatiotemporal fluctuations and policy-related data will be integrated to a single platform, offering advanced visualisations and enhanced virtual simulations that realise the Smart City Digital Twin concept. For any digital twin to meet its objectives, it must provide a representation of the real system (or selected aspects of it, depending on the purpose of the digital twin) that is as accurate as possible. However, inevitably in the design process, gaps between the digital twin and the actual system will likely exist. These gaps and the necessary amendments triggered should be fully understood and considered in the security strategy.

In this deliverable, we aim at defining a multi-layered security approach which involves the deployment of several security mechanisms and privacy control points based on established approaches that will try to cover the potential threats in an as much as possible unobtrusive way. The first step is to identify the threat landscape in the context of Smart Cities and Digital Twins (Section 2). Based on the security challenges of modern, complex IoT infrastructures and the concerns raised within the software development processes that implement and manage such infrastructure, we define the DUET Threat Taxonomy. This will serve as a basis upon which relevant measures should be taken in order to realise a secure and privacy-preserving Digital Twin implementation.

We then continue in Section 3 with the definition of the Security Requirements that stem from the above mentioned threats and cover the key security aspects required to protect data and processes within DUET, namely Confidentiality, Integrity, Availability, Accountability and Authenticity. Section 4 presents the relevant security and privacy measures that have to be implemented so as to address the requirements identified by defining a set of Technical Controls (TC) which will act as a checklist for DUET's realisation of security and privacy mechanisms.

Finally, Section 5 gives the actual implementation of the security measures in the context of the first prototype of DUET. The description includes currently implemented processes and components or provides concrete plans and next steps to address measures that are not yet covered in the current development phase. It must be noted, that upon development of the first integrated prototype and initial user feedback, the security and privacy mechanisms will be updated and specialised to cover emerging needs. The next version of this deliverable (D.11) will report on these updates.

2. Security in Smart Cities and Digital Twins

2.1 Implications and Concerns

Towards an increasingly efficient Smart City, billions of 'Things' get interconnected, dependent on each other and constantly more intelligent. This leads to a gigantic, complex ecosystem that involves socioeconomic, political and technical challenges that need to be addressed along the paramount need for security, safety and privacy, since decisions made based on IoT are tightly intertwined with the physical world. A digital twin that resembles actual systems with high precision can serve as a blueprint to the real system and result in highly impactful consequences in case of compromise by malicious third parties. Such real systems that are nowadays powered by smart city technologies include traffic control, parking, street lighting, public transportation, energy, water and waste management as well as security systems (e.g. cameras).

Main challenges faced in these areas include lack of cyber security practices and little or no testing of the existing ones. Mitigation plans for security incidents are most of the time non-existent and Computer Emergency Response Teams for cities are hard to formulate. Public sector issues like budget constraints, lack of resources, inadequate training and bureaucracy pose extra challenges to be addressed. When talking about cybersecurity, there are several attributes, properties or goals that exist in the security literature[1,2,3] but most researchers and practitioners agree that Confidentiality, Integrity, and Availability, also known as the Central Intelligence Agency (CIA) triad, are the key ones. In the following we briefly describe those security concepts:

- **Confidentiality** guarantees that even if an unauthorized individual, process, or device manages to access some piece of data (either at rest, in transit or in use), it will not be able to ascertain the meaning of the content itself.
- **Integrity** points out that the data in the system should be protected from modification or deletion by an unauthorized individual, process, or device and ensure that undesirable modifications by authorized ones can be undone.
- **Availability** provides an authorized individual, process, or device access to services or information when legitimately demanded.

Additional desired security requirements include **authenticity** (i.e., the property that an entity is what it claims to be) and **accountability** (i.e., the ability to uniquely trace the actions of an entity to that entity).

A System consists of Assets that may be physical, human or logical ones. Assets in the model may have Weaknesses, which refer to all potential points of attack for Threat Agents, e.g., malicious competitors, unhappy employees, unsatisfied customers, scammers. However, the latter can employ appropriate means and/or realise the required activities (i.e., Threats) to exploit a subset of weaknesses only, which is known as Vulnerabilities. A Threat may lead to an Unwanted Incident breaking one or more security objectives, as listed above, and resulting in Undesired Consequences. For example in the "Ping of Death" DoS attack case, the attacker uses an "illegal packet size" vulnerability of poorly-designed equipment and creates an IP packet

[1] A. J. Neumann, N. Statland and R. D. Webb (1977). "Post-processing audit tools and techniques" *(PDF)*. US Department of Commerce, National Bureau of Standards. pp. 11-3--11-4.

[2] ISO/IEC 27000:2009 (E). (2009). Information technology – Security techniques – Information security management systems – Overview and vocabulary. ISO/IEC.

[3] ISACA. (2008). Glossary of terms, 2008. Retrieved from http://www.isaca.org/Knowledge-Center/Documents/Glossary/glossary.pdf

whose payload exceeds the maximum limit of 65536 bytes, causing the destination to crash or reboot and thus affect its availability. Countermeasures can be activated in order to mitigate those Threats and reduce the Risk, which is interpreted as the likelihood of an unwanted incident weighted by its impacts, so that an unlikely incident will present a high risk if its economic consequences are large (e.g., if value of stolen information is high). Note, however, that some countermeasures can be seen as system assets themselves, which means that one would need to analyse their own vulnerabilities as well. Thus, cyber security consists of a continuing cycle of actions to:

- Identify the risks to the system and the nature of attacks;
- Prevent threats being realised by applying appropriate countermeasures (either at design time or at run-time);
- Prepare/Monitor by measuring how the security of the system is performing, and
- Respond to an attack by restoring impaired assets (if any).

2.2 DUET Threat Taxonomy

Vulnerable legacy systems, insecure communications and unpatched software and hardware elements jeopardise security, while together with low cost, interdependent, widespread deployed devices from various vendors they constitute a large, complex attack surface susceptible to a big range of attacks and abuse methods. The situation gets worse with the current fragmentation of standards and regulations, lack of expertise and unclear liabilities in security incident management. Taking into account these concerns , ENISA recommendations [1], [2] have identified threats in the contexts of IoT-based Critical infrastructures and IoT Software Development Life Cycle. The identified threats are based on actual attacks that have been performed during the last years on systems and humans operating, managing and developing IoT based solutions. In Table 1 we do not include the exhaustive list of threats from the relevant resources but rather identify the ones matching the context of DUET which mainly focuses on elements of the IoT ecosystem like applications, communications, cloud backend/services and maintenance and diagnostic tools, rather than the devices (sensors, actuators and embedded systems). However, this doesn't mean that the security profile of devices generating the data available in DUET is not accounted for. Data sources registered to DUET Catalogue will have to meet certain criteria before being listed for usage and device credibility level will be included as mentioned in Section 4 below. Moreover, DUET's microservices architecture exposes an additional set of specific threats as analysed in[4].

Category	Threat	Description
Nefarious activity / Abuse	Malware and Exploit Kits	Software designed to perform unwanted actions or take control of a system via its vulnerabilities.
	Target attacks / DDoS	Repeated attacks taking place in a long period of time and orchestrated multi-source attacks to a specific target.
	Counterfeit by malicious devices	Devices resembling the original ones that can be used to conduct attacks once placed inside an IoT environment.

[4] Hannousse, Abdelhakim & Salima, Yahiouche. (2020). Securing Microservices and Microservice Architectures: A Systematic Mapping Study.

	Attacks on privacy	Exposure of network elements and data to unauthorized parties
	Abuse of authorisation	Unauthorised data access, software installation or use of devices and systems.
	Data abuse	Manipulation of data to gain monetary benefits or cause damages, (test) data poisoning
	Identity theft	Stealing of a legitimate user's identity to perform actions on behalf of her.
Eavesdropping / Interception / Hijacking	Man in the middle	Active eavesdropping attack, in which the attacker relays messages from one victim to another, in order to make them believe that they are talking directly to each other
	IoT communication protocol hijacking	Sniffing sensible information including passwords, forcing disconnections or denial of service
	Interception of information / Session hijacking	Unauthorised interception of communications / Stealing of data
	Network reconnaissance	Passive obtaining of internal information of the IoT network, e.g. devices, protocols, etc.
	Replay of messages	Valid data transmission is maliciously or fraudulently repeated or delayed
Outages	Network Outage	Intentional or accidental failure in network supply.
	Failures of devices or system	Hardware or software failures.
	Loss of support services	Unavailability of business software, interruption of cloud services, third-party APIs failures.
Damage / Loss (IT Assets)	Data Disclosure	Disclosure of source code, test/production data, third-party information or backup data.
	Sensitive information leakage	Sensitive data is revealed, intentionally or not, to unauthorised parties due to e.g. corporate espionage or incompetent / inexperienced / demotivated staff.
Failures / Malfunctions	Software vulnerabilities	Weak passwords, software bugs, configuration errors, outdated software, insecure communication protocols, legacy software,
	SDLC process failures	Failures in development, testing and production environments. High complexity, bad software design and inadequate processes.
	Third parties failures	Errors on an active element of the network caused by the misconfiguration of another element that has direct relation with it.
	Infrastructure attacks	Compromise of containers, virtual machines, hypervisor, discovery services, management interfaces and operating systems. Downgrade, port scan and cold boot attacks.

Disasters	Natural and Environmental Disasters	Floods, earthquakes, landslides etc. Disasters in the deployment environments of IoT infrastructure.
Physical attacks	Device modification	Tampering a device to affect communications and data
	Device destruction (sabotage)	Theft, vandalism or sabotage that causes damage to devices.
Unintentional Damages (Accidental)	Unintentional modifications to source code or data	A member of the development team unintentionally introduces mistakes in code, configuration, test data, backup data, documentation or dev environment.
	Erroneous use or administration of devices and systems	Information leakage / sharing / damage or system management misuse that could affect development, testing and production environments.
Legal	Violation of rules and regulations	Lack of compliance with applicable regulations
	Breach of legislation	Lack of compliance with applicable legislative framework
	Contract Requirements	Improper / Incomplete use of definition

Table 1: Threat Taxonomy for DUET Digital Twins

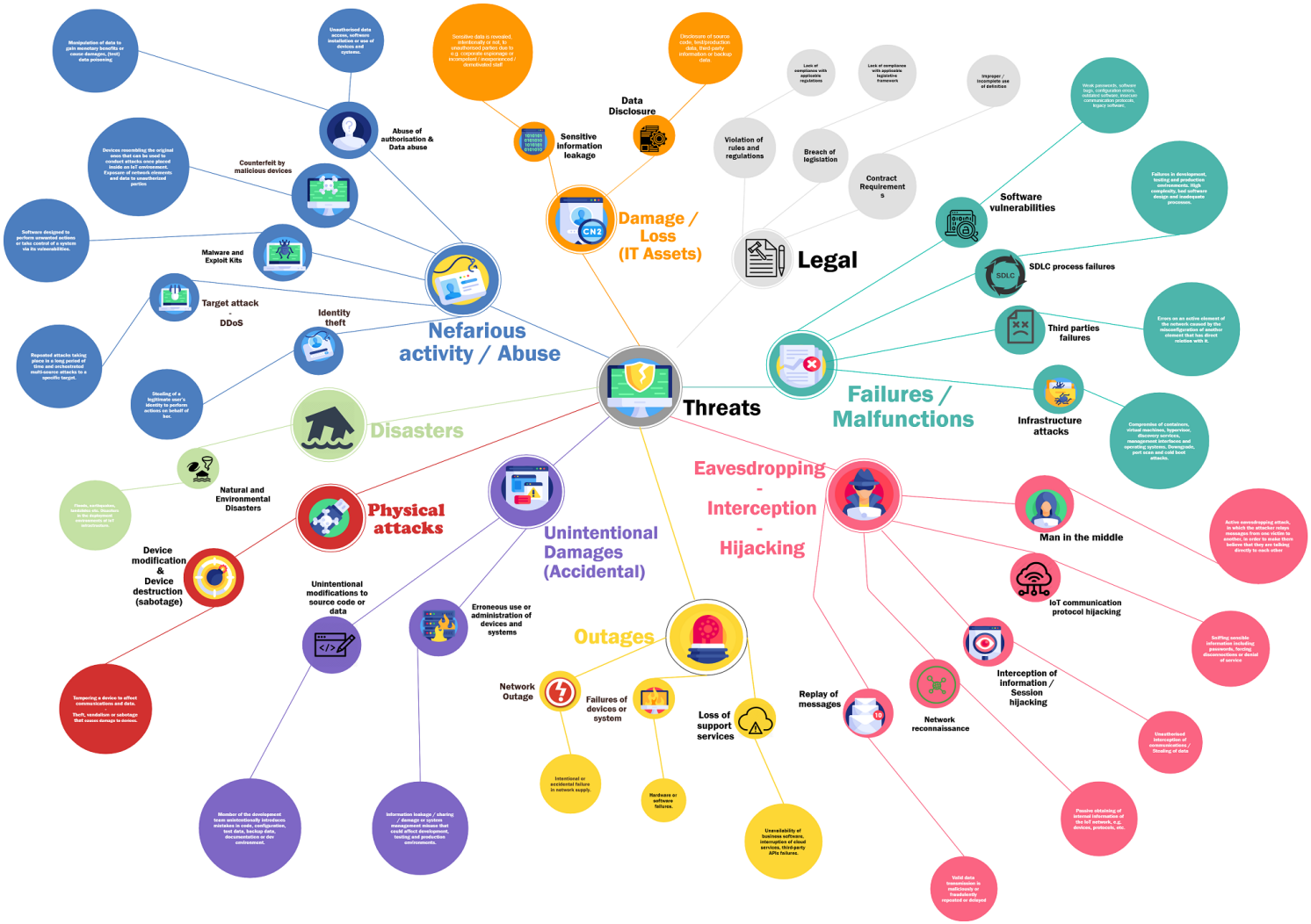

Figure 1: The DUET Threat Taxonomy

2.3 High-level considerations towards a secure Digital Twin

The top security and privacy requirements that must be considered in the context of an IoT-enabled smart city according to Hernanzez-Ramos et al[5] are presented below:

- *Secure Communications for Resource-Constrained Devices and Networks*

Security protocols and cryptographic algorithms need to be adapted to devices with low capabilities in terms of battery endurance and processing power.

- *Automated and Secure Deployment of IoT Devices*

Configurations must not share default credentials and should be equipped with a way to manage secure deployment. A continuous Security Assessment should be in place so as to prevent configuration errors and cascading effects. Furthermore, configuration and updates should be only performed by authorised personnel

- *Transparent and Decentralized Data Sharing using Interoperable and Secure Data Formats*

Data-driven services can enable city authorities and citizens to create innovative services and also identify potential threats to their current applications. A robust, transparent and secure data sharing schema should be in place so as to increase transparency and trust between parties sharing their data. Another big hurdle towards interoperability in IoT environments are the different sensors, communication protocols and data platforms. Usage of common data representations and semantics should be pursued so as to foster sharing of information between systems and development of new services, while preserving integrity and availability of offered data.

- *Access Control Management and Informed Consent*

Access control policies should be in place to control data processing and information sharing between producers and consumers. This involves authentication (i.e., techniques used to verify the identity of users requiring access to system resources and data) and Authorization. Examples of authentication schemes and mechanisms include: Centralized Access Control Manager, Certificates, Open ID, Single Sign On (SSO), White-list HTTP/IP, HIP exchange protocol, J-PAKE protocol, Distribute sessions and HTTP signatures. Similarly, examples of Authorization approaches and mechanisms include Attribute Based Access Control (ABAC), Role Based Access Control (RBAC), R/W Permission to message broker, OAuth 2, JSON Web Token (JWT)[6] and Firewalls[7]. Legal principles and data protection regulations, i.e. GDPR must be respected at all times.

- *Privacy-Preserving Data Analytics according to current Security and Privacy Regulations*

Data analytics used to enable data-driven decision making processes must be carefully designed and executed so as to address privacy concerns. The latter mainly arise from the fact that computational resources to

[5] Hernández-Ramos, José & Martinez, Juan & Savarino, Vincenzo & Angelini, Marco & Napolitano, Vincenzo & Skarmeta, Antonio & Baldini, Gianmarco. (2020). Security and Privacy in Internet of Things-Enabled Smart Cities: Challenges and Future Directions. IEEE Security and Privacy Magazine. PP. 10.1109/MSEC.2020.3012353.

[6] Can be also used for authentication

[7] Can be also used for authentication

perform analytics in vast amounts of IoT data usually require cloud infrastructures to handle the load and often include machine learning techniques to extract useful insights. Privacy-enhancing techniques must be employed to help reconcile massive data collection with privacy requirements while preserving usability of collected data. Therefore, a trade-off between social, legal, ethical and business constraints must be in place so as to support a sustainable business model for smart cities. Smart Cities solutions must guarantee compliance to GDPR and any other applicable privacy regulation as well as following recommendations on cyber security (e.g. Cybersecurity Act).

- *Cybersecurity Awareness*

Citizens and all Smart City stakeholders should gain awareness of cybersecurity aspects involved in a Smart City context and acquire the necessary knowledge regarding security and privacy risks that may arise during the complete lifecycle of IoT enabled smart technologies. For this reason, auditing, mitigation and prevention measures are required. Auditing involves techniques applied at runtime for discovering security gaps, such as Continuous monitoring, Intrusion detection, Scan container images and Static/Dynamic code analysis. Mitigation includes techniques that limit the damage of attacks when they appear. Examples of mitigation techniques for microservice-based systems include Roll-back/Restart microservices, Scale up/down N-variant microservices, Short-lived tokens, Diversification, IP shuffling, Live migration, Deception (e.g., using honeypots) and isolation of suspicious microservices. Prevention refers to techniques that try to stop attacks from happening in the first place, such as encryption of data using TLS protocol and code using SGX technology with enclaves, Hardware Security Module (HMS), No shared memory access, Blockchain technology.

The following sections dive deeper into these considerations and identify the security requirements of the DUET environment, the measures that need to be taken in order to cover them, as well as the currently implemented DUET mechanisms that actualise these measures.

3. DUET Security Requirements

In this section, we list the security requirements stemming from the identified threats presented in the previous paragraphs. We separate the requirements in two main categories, namely the Run-time requirements and the requirements in DUET's Software Development Life Cycle (SDLC). A second categorisation is based on the key security concepts that drive DUET's policies on data and information protection as mentioned in Sect. 2.1.

3.1 DUET Run-time Security Requirements

3.1.1 Confidentiality

- Smart city-related information (or contextual information) held within DUET subsystems (including IoT devices) should be protected from unauthorized access. This should apply not only to personal data, but data-sets combining personal and non-personal (also known as "mixed").
    - **Co1**: A DUET subsystem shall provide a way for stating that certain information is restricted (e.g., due to privacy issues) including any data retention limitations (e.g., data expiry).
    - **Co2**: A DUET subsystem shall permit only authorized parties (e.g., users, applications, etc) to access its restricted information, while unsuccessful attempts should be discouraged.
- Smart city-related information sent to, or from, an authorized DUET subsystem (or an IoT device) should be revealed only to parties authorized to receive the information.
    - **Co3**: Restricted information sent to and from an authorized DUET subsystem (or an IoT device) shall be encrypted using state-of-the-art protocols and following best practices (e.g., regarding key length).
    - **Co4**: Before transmitting restricted information to another party, a DUET subsystem (or an IoT device) shall authenticate itself to the recipient.
    - **Co5**: Before receiving restricted information from another party, a DUET subsystem (or an IoT device) shall be required to authenticate itself to the sender
- Management Information held within a DUET subsystem (or an IoT device) should be protected from unauthorized access
    - **Co6**: A DUET subsystem (or an IoT device) shall provide a way for designating that a certain party (or group of parties) is authorized to access stored management information
    - **Co7**: A DUET subsystem (or an IoT device) shall permit only authorized parties to learn details about stored management information such as service profile data, software version, supported security protocols and service capabilities.
- Management Information sent to, or from, a DUET subsystem (or an IoT device) should be protected from unauthorized access
    - **Co8**: A DUET subsystem (or an IoT device) shall restrict access to transmitted management information to authorized parties
- It should not be possible for an unauthorized party to deduce the identity and other personal identifiable information of an individual
    - **Co9**: DUET shall ensure that unauthorized entities are unable to isolate some or all records which identify another target data subject in the dataset (property frequently known as "Immunity to Singling out").

- **Co10**: DUET shall ensure that unauthorized entities are unable to link two or more records concerning the same data subject either in the same database or in different databases (property frequently known as “Immunity to Linkability”).
- **Co11**: DUET shall ensure that only authorized entities are able to associate a pseudonym with the real username.

- It should not be possible for an unauthorized party to deduce the location and identity of an IoT device by analyzing communications traffic flows to and from the IoT device
  - **Co12**: An IoT device shall have the means to protect location and identifier during transmission

3.1.2 Integrity

The following security objectives related to the integrity of stored and transmitted information are specified:

- Information held within a DUET subsystem (or an IoT device) should be protected from unauthorized addition, modification and deletion.
  - **In1**: A DUET subsystem shall permit only authorized parties to modify or delete historic contextual information. Keeping data up-to-date is important and this is especially true in the case of personal data.
  - **In2**: A DUET subsystem shall permit both authorized and unauthorized parties to add contextual information
  - **In3**: A DUET subsystem shall be able to infer whether some pieces of contextual information originated from an authorized party, or not
- Data and Management Information sent to or from an authenticated party should be protected against unauthorized or malicious modification or manipulation during transmission.
  - **In4**: A DUET subsystem (or an IoT device) shall implement one or more methods to enable the sending/receiving party to detect en-route modification or manipulation of data/Management Information
  - **In5**: A DUET subsystem (or an IoT device) shall implement one or more methods for preventing the modification or manipulation of data/Management Information that it transmits or receives.
- Management Information held within a DUET subsystem (or an IoT device) should be protected from unauthorized modification and deletion.
  - **In6**: A DUET subsystem (or an IoT device) shall permit only authorized parties to add, modify or delete parameters related to security protocols and service capabilities

3.1.3 Availability

- Access to and the operation of a DUET subsystem (or an IoT device) by authorized users should not be prevented by malicious activity.
  - **Av1**: A DUET subsystem (or an IoT device) should be able to detect and confront easily recognizable Denial of Service attack patterns.
  - **Av2**: A DUET subsystem (or an IoT device) should respond to an authorised party within reasonable amount of time (e.g., be scalable to dynamic conditions)

3.1.4 Accountability

- It should be possible to audit all changes to security parameters and applications (updates, additions and deletions).
    - **Ac1**: A DUET subsystem (or an IoT device) shall record all requests for changes to supported security protocols and service capabilities
    - **Ac2**: A DUET subsystem (or an IoT device) shall record the results of all requests for changes to supported security protocols and service capabilities
- It should be possible to acknowledge the receipt or transmission of information to a party
    - **Ac3**: A DUET subsystem (or an IoT device) should be able to indicate to another party that exchanged information should be acknowledged and agree on the backlog size
    - **Ac4**: A DUET subsystem (or an IoT device) shall be able to acknowledge the receipt of the last N pieces of information sent by the other party
    - **Ac5**: A DUET subsystem (or an IoT device) shall be able to acknowledge the submission of the last N pieces of information sent to the other party

3.1.5 Authenticity

- It should not be possible for a party to pose as a different entity when communicating with DUET backend services or IoT devices.
    - **Au1**: A party shall have the means to prove its identity, ideally by presenting more than one type of evidence: knowledge (something only the user knows), possession (something only the user has), and inherence (something only the user is).
    - **Au2**: A party shall have the ability to update the credentials (e.g., in case of forgotten details or for security purposes)
    - **Au3**: A party shall reject a request or information received from an unauthorised DUET back-end service.

3.2 Security requirements in DUET's Software Development Life Cycle (SDLC)

This section lists the requirements and good practises that will drive the Software Development Life Cycle (SDLC) of DUET. They follow ENISA's[8] logical domain categorisation which is grouped in three main groups: People, Processes and Technologies.

3.2.1 Confidentiality

- Sensitive technical information of DUET subsystems should be protected from unauthorized access
    - **Co13**: The DUET development environment shall provide a way for restricting access to the source code of particular subsystems only to authorised parties. Parties may take on one or more ICT roles (as shown in the figure below) and/or may belong to different entities.

[8] https://www.enisa.europa.eu/publications/good-practices-for-security-of-iot-1

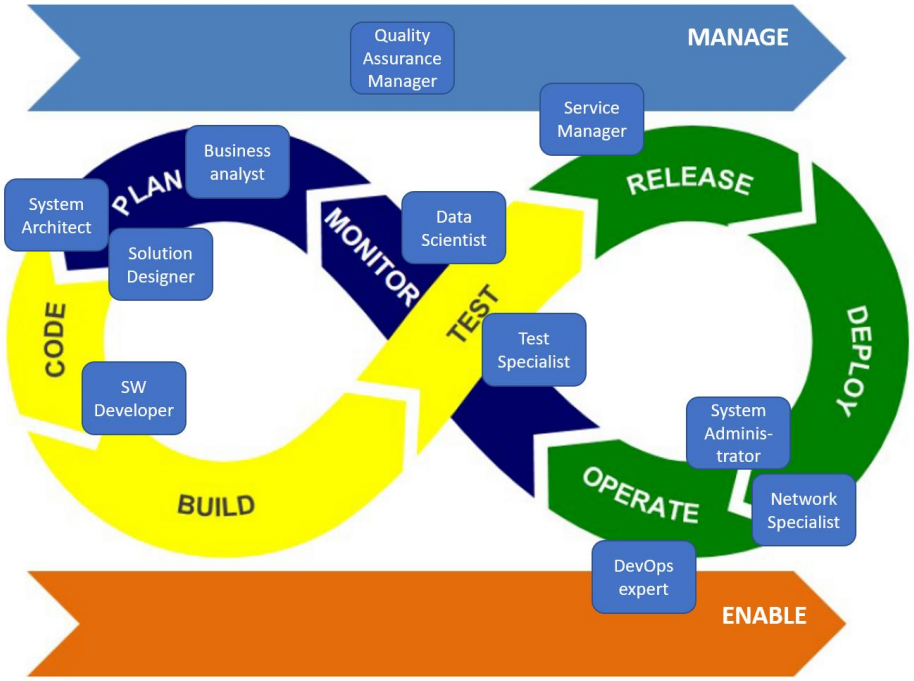

Figure 2: ICT Roles in the Development Environment[9]

- **Co14**: The DUET operations environment shall provide a way for restricting access to sensitive performance statistics of DUET subsystems only to authorised parties (e.g., to System Administrators, Test Specialists, Network Specialists, Data Scientists, Quality Assurance Managers).

3.2.2 Integrity

- Sensitive technical information of DUET subsystems should be protected from unauthorized addition, modification and deletion.
  - **In7**: The DUET development environment shall permit only authorized developers to add new features, or modify, delete existing ones.
  - **In8**: The DUET operations environment shall permit only authorized administrators to modify or delete critical configuration options of DUET subsystems.
  - **In9**: The DUET operations environment shall permit only authorized managers to modify or delete traces and performance statistics of DUET subsystems.
  - **In10**: The DUET development environment shall implement measures against rogue code(e.g. backdoors, time bombs) and tampering.

3.2.3 Availability

- Access to the DUET development and operations environment by authorized users should not be prevented by malicious activity.
  - **Av3**: The DUET development and operations environment should be able to detect and confront easily recognizable Denial of Service attack patterns.

[9] Based on COMITÉ EUROPÉEN DE NORMALISATION, European ICT Professional Role Profiles CWA 16458. Available at http://www.ecompetences.eu/ict-professional-profiles/

- **Av4**: The DUET development and operations environment should respond to an authorised party within reasonable amount of time

3.2.4 Accountability

- **Ac6**: It should be possible to audit all changes to DUET subsystems (e.g., updates to docker images).

3.2.5 Authenticity

- It should not be possible for a party to pose as a different entity when interacting with the DUET development and operations environment.
  - **Au4**: A party shall have the means to prove its identity
  - **Au5**: A party shall have the ability to update the credentials
  - **Au6**: The DUET development and operations environment shall reject a request or information received from an unauthorised DUET back-end service

The following table presents a concise list of all the identified requirements:

Confidentiality	Integrity
Co1: Support DUET subsystems in stating that certain contextual information is restricted Co2: Permit only authorized parties to access restricted contextual information in DUET Co3: Support state-of-the-art encryption for contextual information in transit Co4: A DUET backend service should send restricted contextual information only after authentication has taken place Co5: A DUET backend service should receive restricted contextual information only after authentication has taken place Co6: Designate which parties are authorized to access stored management information Co7: Restrict access to sensitive stored management information Co8: Restrict access to transmitted management information only to authorized parties Co9: Immunity to Singling out Co10:Immunity to Linkability Co11: Permit only authorized parties to determine the personal identifiable information based on a pseudonym Co12: Protect sensitive management information of IoT devices Co13: Permit only authorized parties to access source code of DUET sub-system(s) Co14: Protect sensitive management information of DUET development environment	In1: Only authorized parties should modify or delete historic contextual information In2: Permit both authorized and unauthorized parties to add contextual information In3: Able to discriminate authorized and unauthorized contextual information In4: detect en-route modification or manipulation of data/Management Information In5: Prevent the modification or manipulation of data/Management Information In6: Only authorized parties should add, modify or delete parameters related to security protocols and service capabilities In7: Only authorized parties should manage critical DUET configuration options In8: Only authorized administrators should modify or delete critical configuration options In9: Only authorized managers should modify or delete performance statistics In10: Prevent rogue code

Availability	Authenticity
Av1: Detect and confront simple Denial of Service attacks Av2: respond to an authorized party within reasonable amount of time Av3: Detect and confront simple Denial of Service attacks on the DUET development environment Av4: DUET development environment should respond to an authorized party within reasonable amount of time	Au1: Present 1 or more credential types Au2: Ability to update credentials Au3: Reject unauthorized request or information Au4: Prove identity of DUET contributor Au5: Ability to update a DUET contributor's credentials Au6: Reject unauthorized request or information from unauthorized DUET back-end services
Accountability	
Ac1: Log all requests for changes to supported security protocols and service capabilities Ac2: Log results of all requests for changes to supported security protocols and service capabilities Ac3: Negotiate details for acknowledging information to be exchanged Ac4: Acknowledge receipt of information Ac5: Acknowledge submission of information	

Table 2: List of security and privacy requirements for DUET Digital Twins

4. DUET Security Measures

Below we provide an overview of technical measures to preserve and protect the DUET ecosystem. We code them as Technical Controls (TCs) which will then serve as a checklist during the implementation of the platform. This way, we will be able to perform periodical checks and verification of the employed means of realisation for every TC, thus ensuring a robust and well-maintained security and privacy model.

4.1 Run-time Authentication (when connecting to a DUET backend service, visualisation system or sensor)

TC.1. Authentication mechanisms will use strong passwords by enforcing policies such as minimum password length, minimum number of symbols, mix of upper lower and upper case, etc. Furthermore, two-factor authentication (2FA) should be enabled for critical DUET subsystems and actions.

TC.2. Use state-of-the-art, standardised and effective cryptography and security protocols, such as TLS for encryption in order to protect the confidentiality, authenticity and/or integrity of data and information (including control messages) in transit.

TC.3. Ensure password recovery or reset mechanism is robust and does not supply an attacker with information indicating a valid account.

TC.4. Countermeasures for detecting and stopping “brute force” attacks should be in place. For example, rate limiting could be applied for controlling the requests to backend service to reduce the risk of automated attacks.

TC.5. Secure storage of users' credentials. Ensure that user credentials of IoT systems (and other underlying infrastructure) are protected from disclosure. Authentication credentials must be salted[10], hashed and/or encrypted.

TC.6. Encryption keys that are stored in devices or DUET subsystems should be protected and securely managed.

The following figure presents how technical measures related to run-time authentication are mapped into security objectives from Section 3.

[10] “Salt” is a random set of characters that is added to the user’s password before a one-way hashing function is used in order to be stored. The idea is to avoid two users choosing the same password and thus better withstand attacks based on dictionaries and rainbow tables.

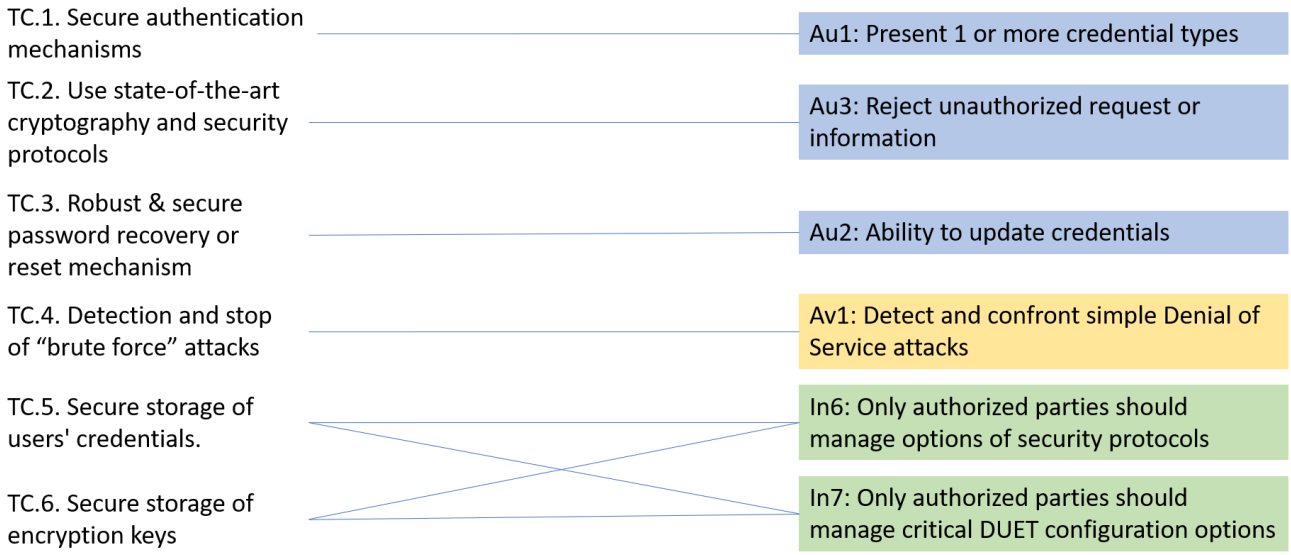

Figure 3: Mapping of run-time authentication measures and security objectives

4.2 Run-time Authorisation

TC.7. Implement authorisation: Implement access control in DUET backend services to ensure that the system verifies that users and applications have the right permissions. Security roles and privileges should be established for both systems or users and fine-grained authorisation mechanisms should be in place for limiting the actions allowed. Furthermore, applications and users shall follow the Principle of least privilege (POLP) and operate at the lowest privilege level possible. Related security objectives are shown in the following figure.

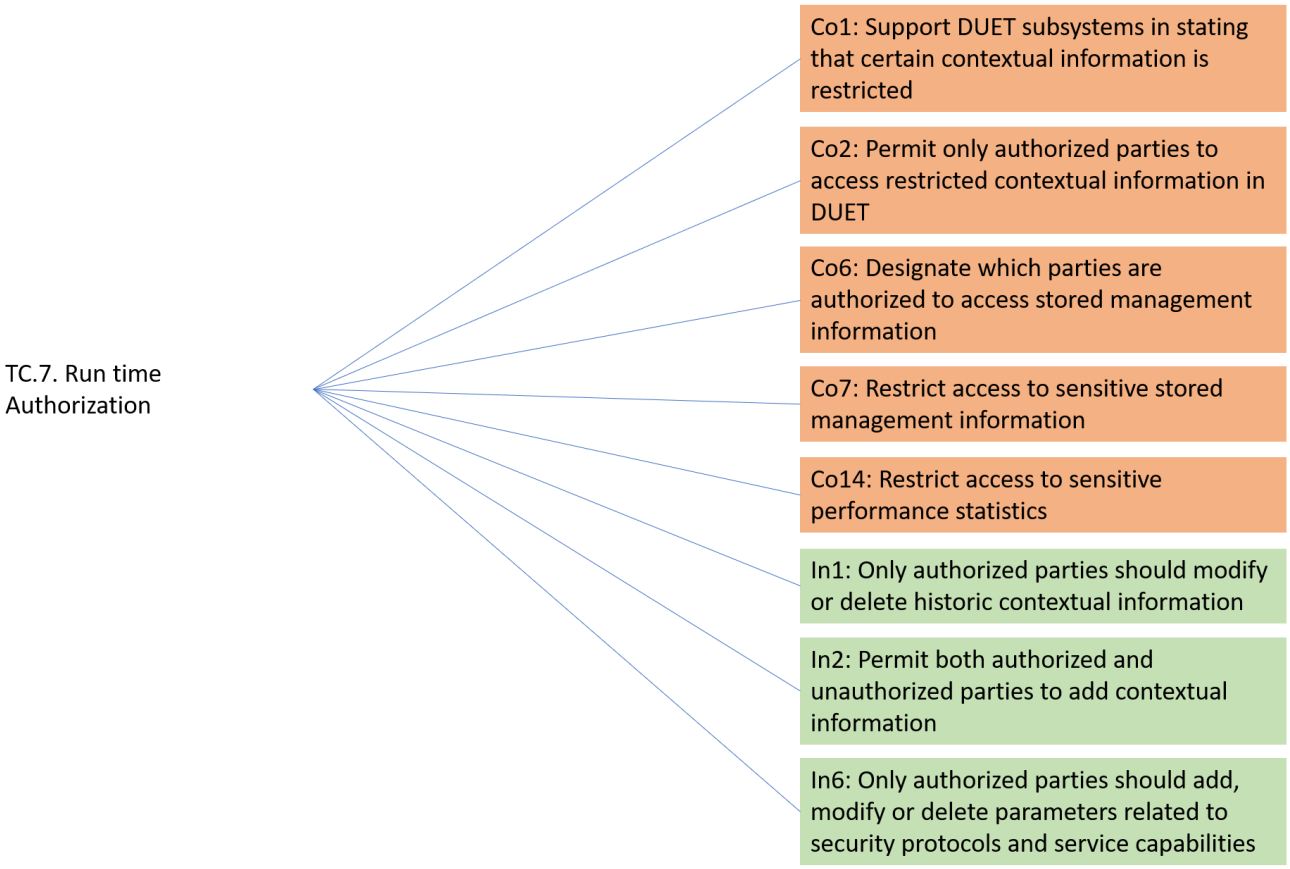

Figure 4: Mapping of run-time authorisation measures and security objectives

4.3 Secure and trusted communications

TC.8. Only Accept devices/APIs via https in order to guarantee authenticity of the accessed service, protection of the integrity and confidentiality of the information exchanged.

TC.9. Use a modern cryptographic hash algorithm to guarantee integrity of the information received. This is done by producing a fingerprint such that it is non-tractable to retrieve the source file using only the hash, the probability of creating two different files that result in the same hash is extremely low and any modification to the source file will produce a substantially different hash. Failing to do so may result in compromised IoT devices sending poisoned data to the backend systems so that the latter takes wrong decisions; a threat commonly known as a poison attack.

TC.10. Data should always be signed whenever and wherever it is captured and stored in order to guarantee data authenticity. In the case of Symmetric key cryptography a single key is used, which is only known to the corresponding parties. In the case of asymmetric key cryptography a pair of key exists, in which one part, the secret key, is known only to the holder, and the second part, the public key, can be known to anybody (i.e. made public).

TC.11. Data exchange should be acknowledged in order to guarantee accountability of the information.

TC.12. Disable specific ports and/or network connections for selective connectivity by including firewalls and virtual private networks.

TC.13. Implement a DDoS-resistant and Load-Balancing infrastructure.

TC.14. Ensure that errors are handled correctly, that all input/output data are validated before accepting it, and that queries use parameterisation (or other equivalent security measure) to avoid code injections (XSS, CSRF, SQL injection, etc). In particular, when designing error messages, stack trace, debug information and other information that malicious users may exploit to gain detailed understanding of the system should be excluded. Information that should be included in the error messages is a generic description of the problem (e.g., database connection problem) and suggestions to the users for fixing the problem (e.g., checking the connection string on the configuration file). Note that if the system is vulnerable to SQL injection attacks then a malicious user could obtain valuable information even if error messages are carefully crafted (e.g., by asking the system a series of true or false questions in the case of blind SQL injection attacks).

The following figure presents how technical measures related to secure and trusted communications are mapped into security objectives from Section 3.

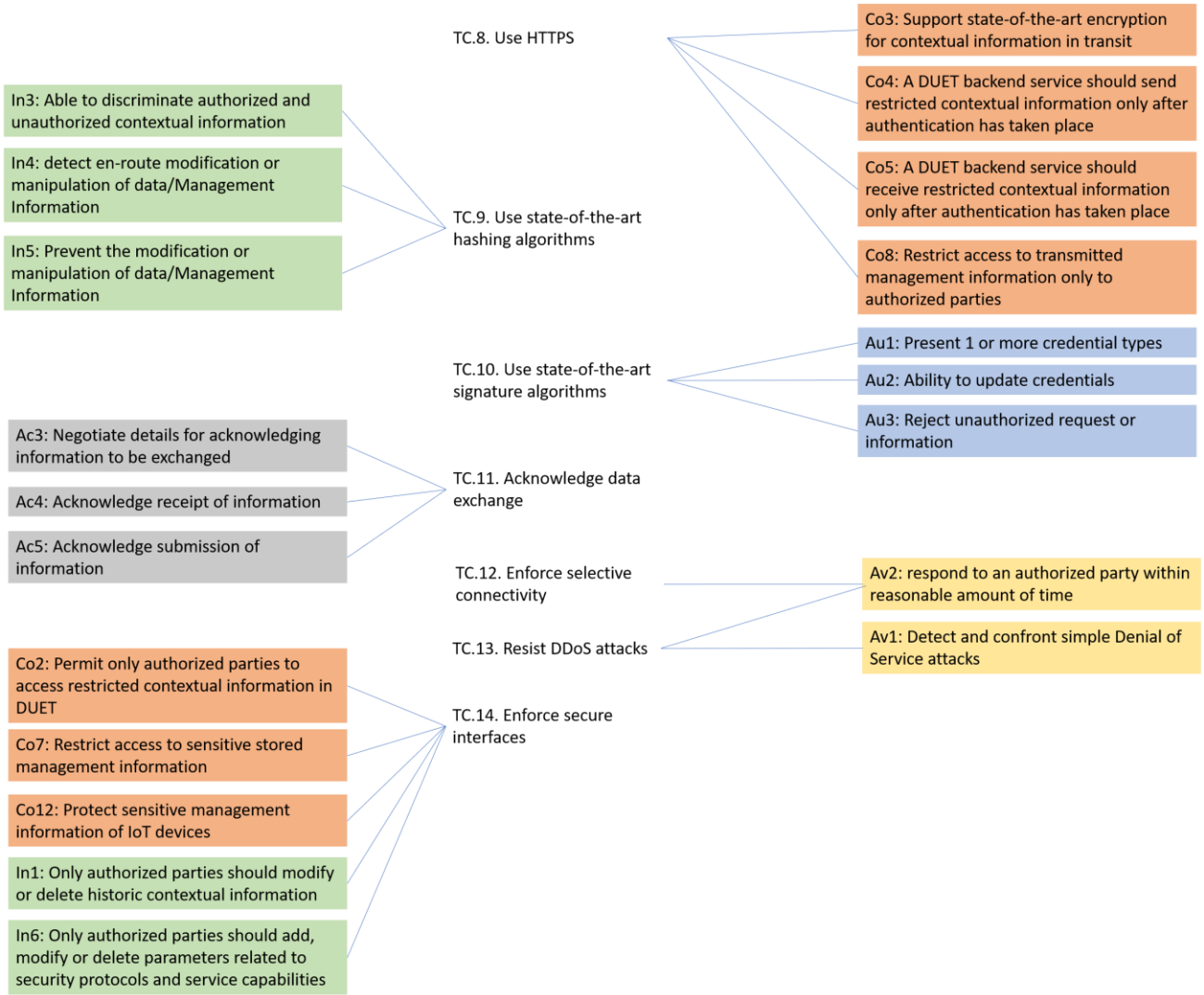

Figure 5: Mapping of communication measures and security objectives

4.4 Run-time Monitoring and Auditing

TC.15. Implement regular monitoring to verify the device behaviour, detect malware and discover integrity errors. For example, “anomaly-based” methods compare the observed network traffic with normal traffic and attacks (e.g., DoS) are detected when irregular activities are witnessed.

TC.16. Implement a logging system that records events relating to user authentication, management of accounts and access rights, modifications to security rules, and the functioning of the system. Logs must be preserved on durable storage and retrievable via authenticated connections.

TC.17. Conduct periodic audits and reviews of security controls to ensure that the controls are effective. Perform penetration tests regularly.

Candidate technical controls for achieving run-time monitoring and auditing and their relationship to security objectives appear on the following figure.

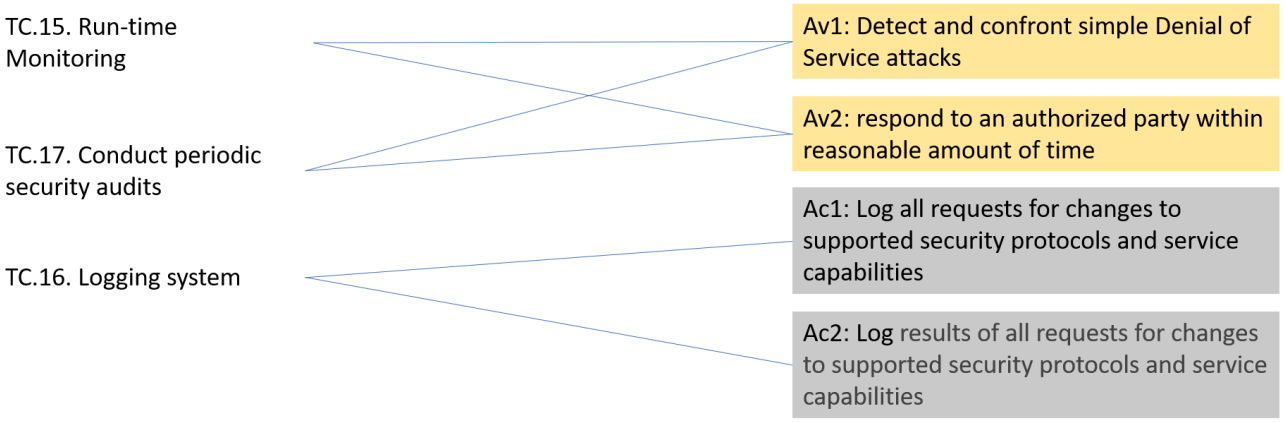

Figure 6 Mapping of monitoring/auditing measures and security objectives

4.5 Data protection and compliance

TC.18. Personal data must be collected and processed fairly, lawfully and in a transparent manner; it should never be collected and processed without the data subject’s consent.

TC.19. Make sure that personal data is used for the specified purposes for which they were collected, and that any further processing of personal data is compatible and that the data subjects are well informed.

TC.20. Anonymise personal data related to an action (e.g., service request) if the person's identity should be unknown or pseudo-anonymised in case the user can be reidentified if necessary by authorised users. Popular security measures for mitigating re-identification are i) k-anonymity, where attributes related to data subjects are suppressed or generalized until each row is identical with at least k-1 other rows and ii) Noise injection, where the actual values are modified in order to prevent linking between the anonymized data and the original.

TC.21. Data subjects must be able to exercise their rights to information, access, erasure, rectification, data portability, restriction of processing, objection to processing, and their right not to be evaluated on the basis of automated processing.

The following figure presents how technical measures related to data privacy are mapped into security objectives from Section 3.

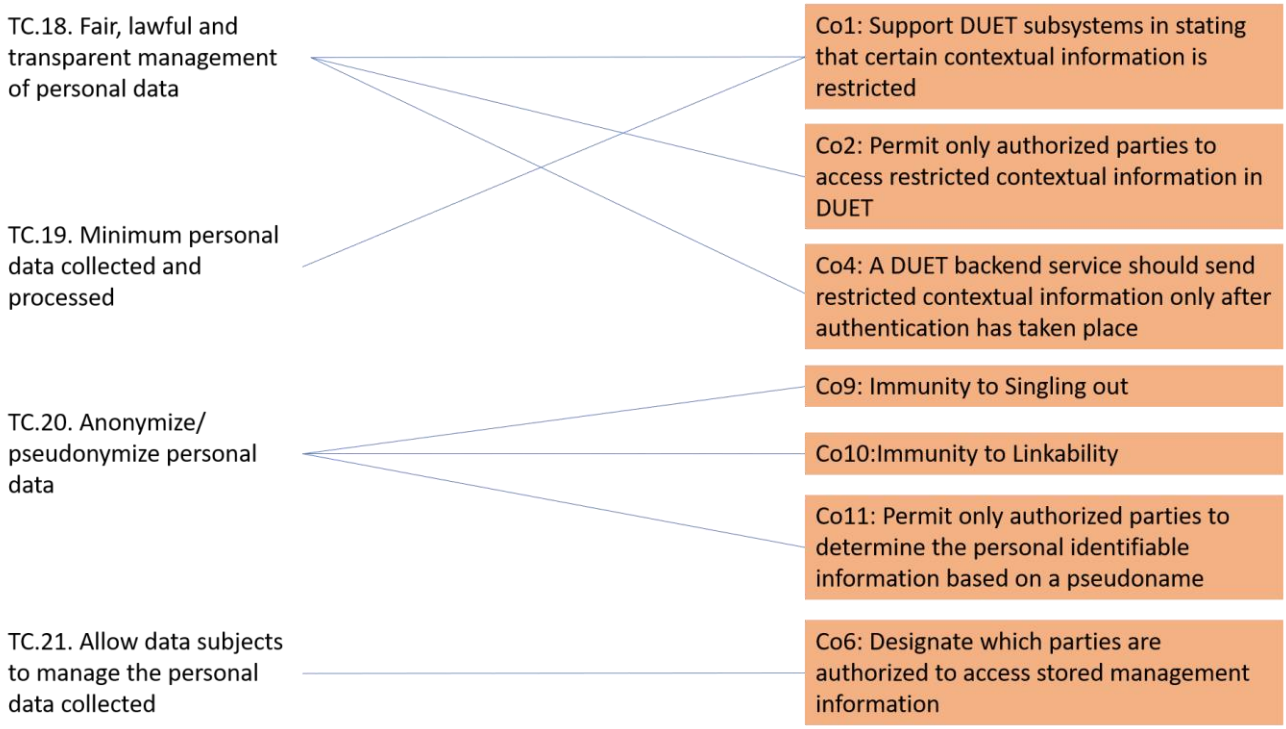

Figure 7: Mapping of data protection measures and security objectives

4.6 Software Development Lifecycle Security

4.6.1 Plan

TC.22. Design with system and operational disruption in mind, preventing the system from causing an unacceptable risk of injury or physical damage. Furthermore, allocate resources for process monitoring: Propose improvements to ensure that a problem during the SDLC process can not cause an interruption of business continuity.

TC.23. Include mechanisms for self-diagnosis and self-repair/healing to recover from failure, malfunction or a compromised state.

TC.24. Ensure standalone operation - essential features should continue to work with a loss of communications and chronicle negative impacts from compromised devices or cloud-based systems.

TC-25. Specify security requirements: Identify security requirements prior to development to implement features that ensure regulatory compliance and avoid vulnerabilities throughout the process.

TC-26. Use established software development techniques: Choose software development techniques (e.g. microservices) or architecture that produce clean and maintainable code.

The following figure presents the subset of DUET security objectives that are addressed by technical measures related to planning a secure development lifecycle process.

Figure 8: Mapping of SDLC planning measures and security objectives

4.6.2 SDLC Authentication and Authorisation

TC-27. Establish security roles and privileges within the development project of a certain DUET subsystem: Carry out a segregation of duties in order to enable the collusion-resistant processes in SDLC and to minimise the risk exposure of its processes. After implementing a separation of duties in the work team, define roles and responsibilities within the process so that the minimum sufficient level of privilege for each duty can be identified and assigned to the relevant person. It is important to note that different DUET subsystems can be developed by different providers and the latter can define the roles and associated privileges according to their own policies.

TC-28. Ensure that default passwords and even default usernames are changed during the initial setup, and that weak or blank passwords are not allowed.

The following figure presents how the technical measures above contribute to the security of the DUET platform.

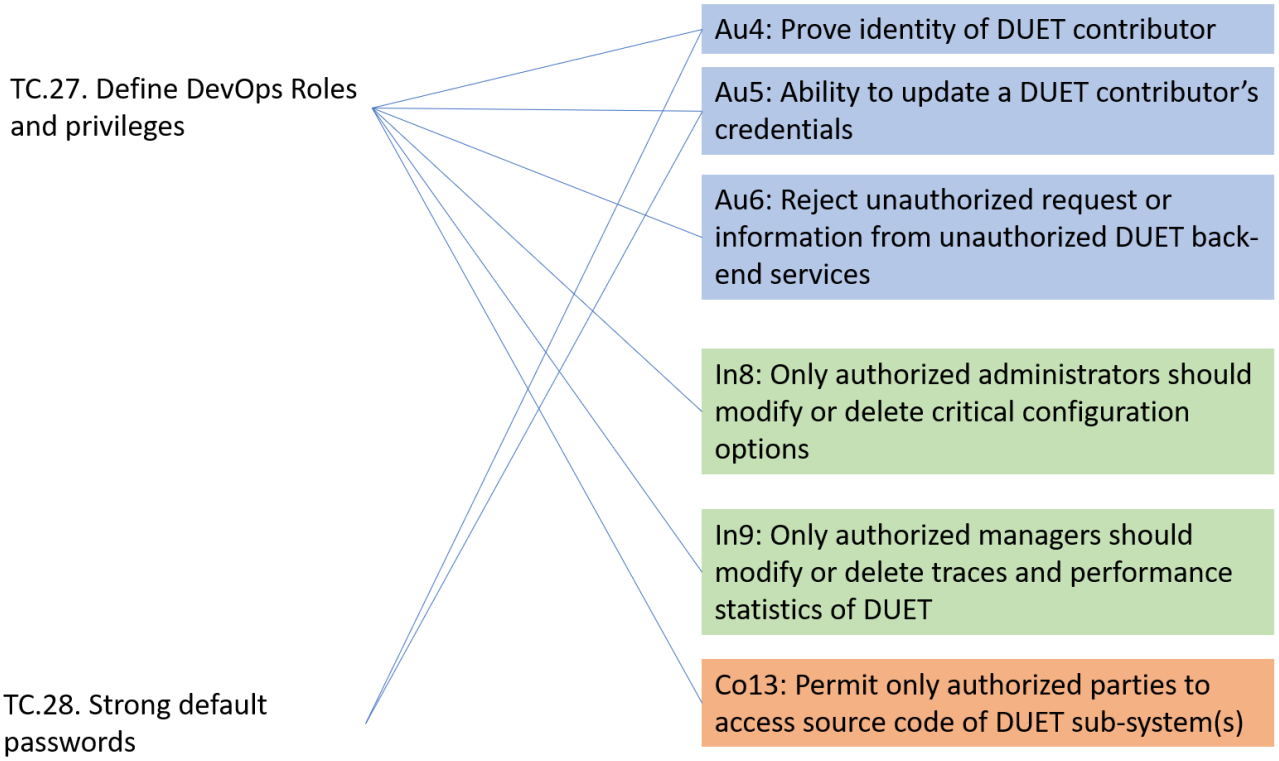

Figure 9: Mapping of SDLC authentication/authorisation measures and security objectives

4.6.3 Secure Development

TC-29. Use libraries and third-party components that are patched for latest known vulnerabilities.

TC-30. Use known secure frameworks with long-term support and ensure that foundation technologies of the software will be maintained in the long term.

TC-31. Any unused functionalities should be disabled by default. Ensure security for patches and updates, i.e., ensure that the SDLC model always allows for modification/patching/update of software in a secure fashion (tested, reviewed, etc.) before deploying any software change.

The following figure presents how the technical controls TC-29, TC-30 and TC-31 contribute to the security of the DUET platform.

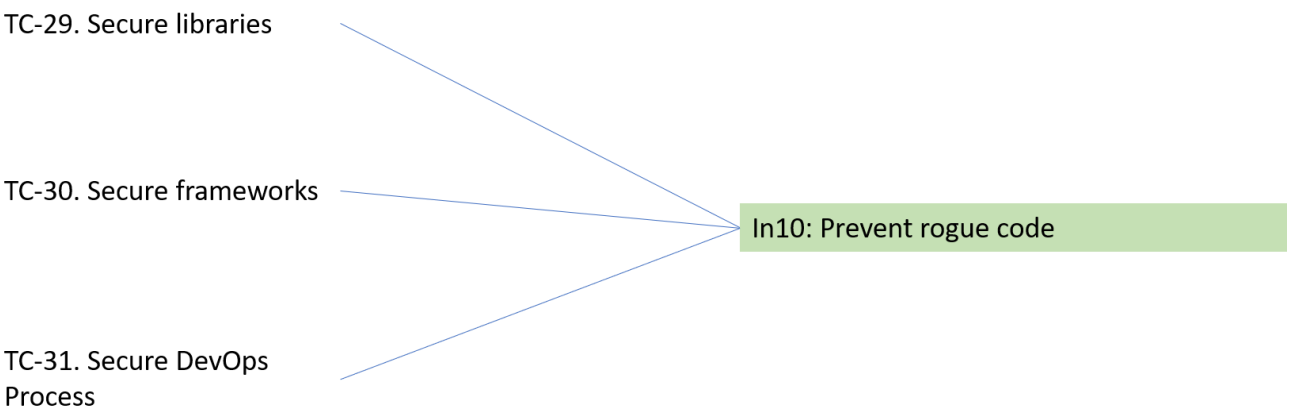

Figure 10: Mapping of SDLC development measures and security objectives

4.6.4 SDLC Monitoring and Auditing

TC-32. Protect the SDLC process against privilege abuse: Implement security controls to prevent the process from being compromised by any user with legitimate rights. Furthermore, adequately manage the integrity of the system by ensuring that no unauthorised changes are made to the configuration.

TC-33. Automate the SDLC process: Automate processes supported by tested tools to improve availability, while reducing errors, costs and human efforts.

TC-34. Provide audit capability: During the design, implement/develop and test the software under development, ensuring that relevant security events are registered in software logs.

The following figure presents how the technical controls above contribute to the security of the DUET platform.

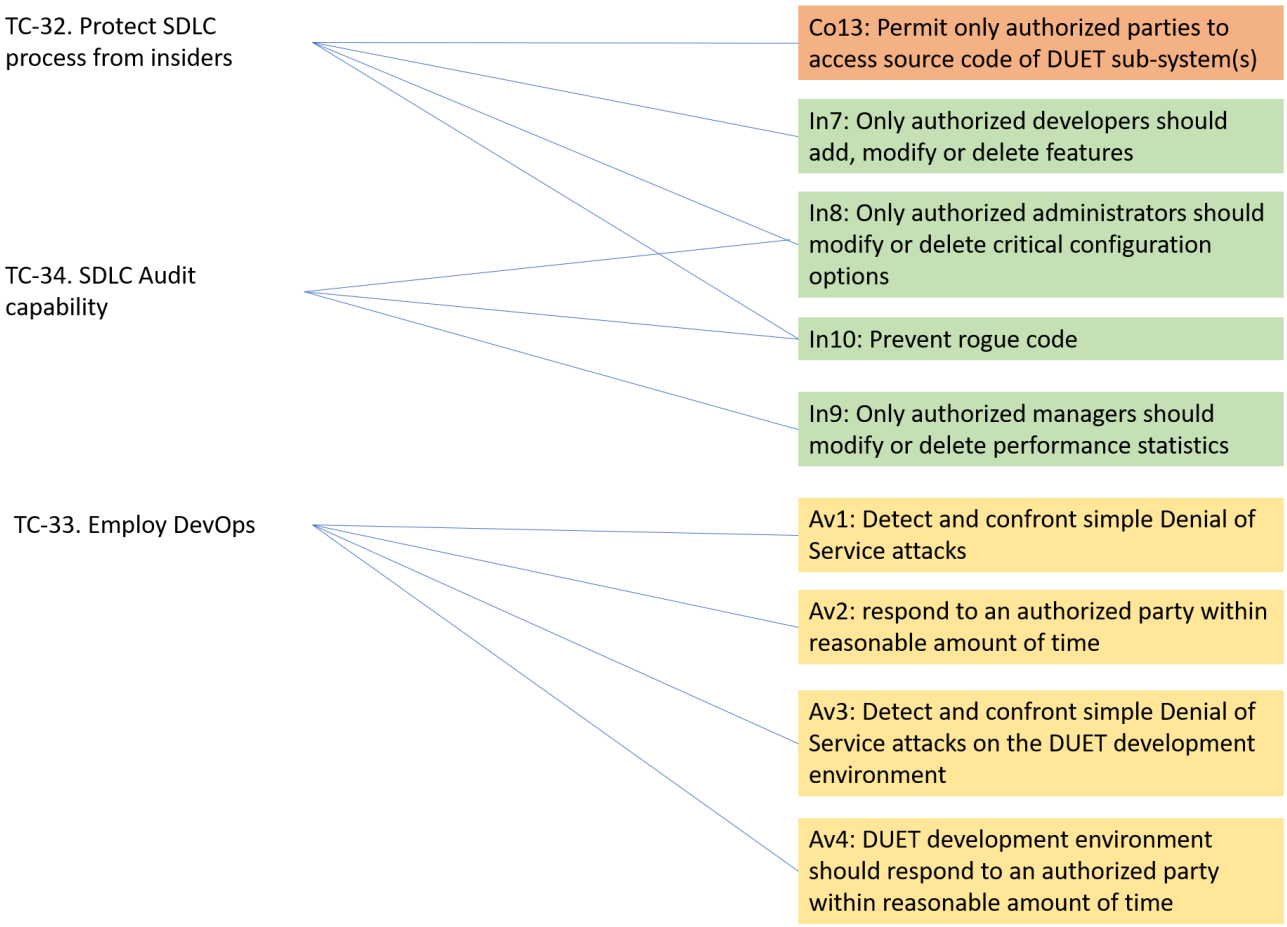

Figure 11: Mapping of SDLC monitoring measures and security objectives

The following map depicts the complete tree of the DUET Security Measures:

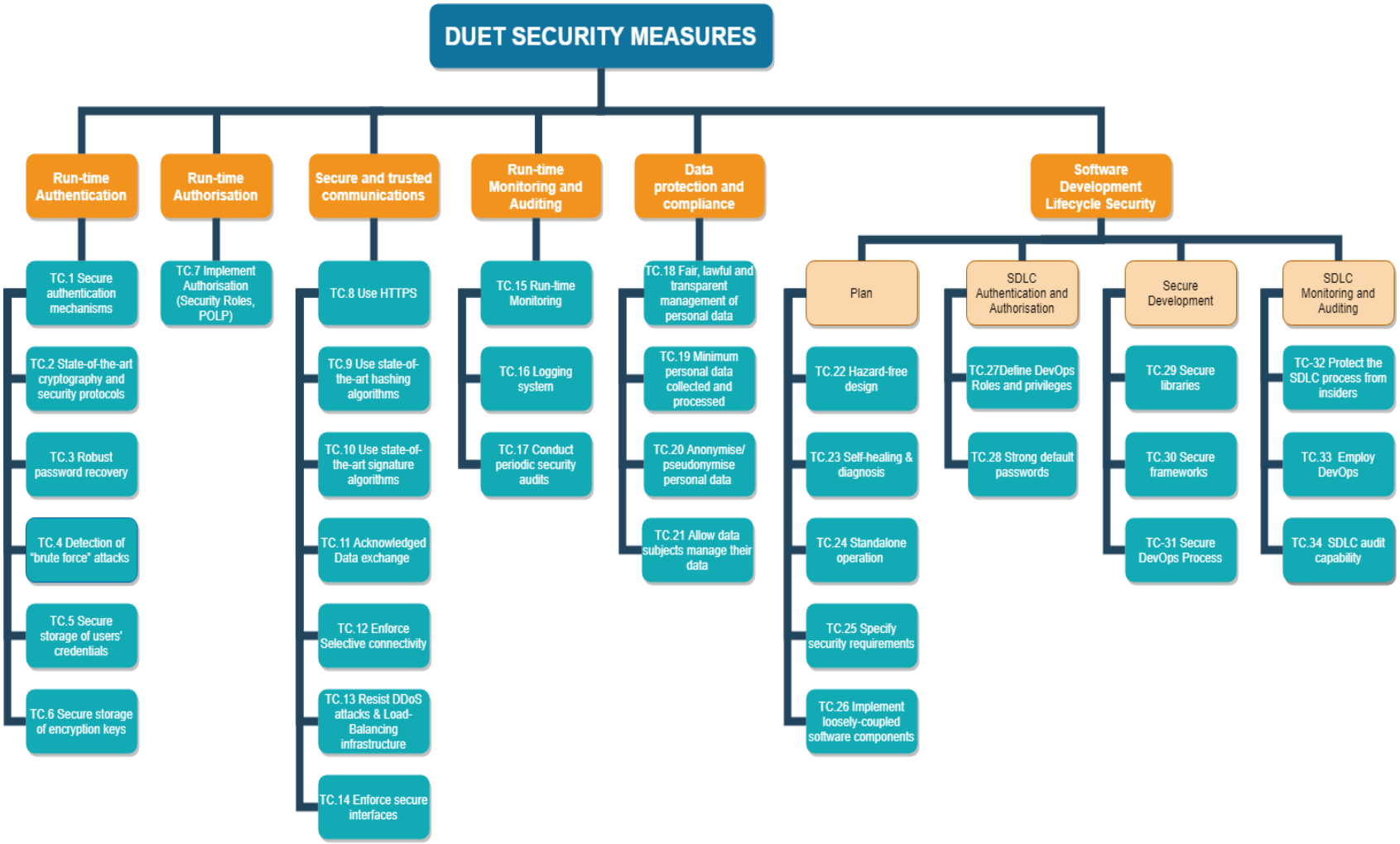

Figure 12: Taxonomy of DUET security measures

5. DUET Multi-layered Security

DUET will implement a Multi-layered Security mechanism including all the security measures required to address the security requirements identified through the defined Threat Taxonomy. The Security Layer runs vertically through all the layers of DUET's architecture as seen in Figure 13 (and analysed in D5.1).

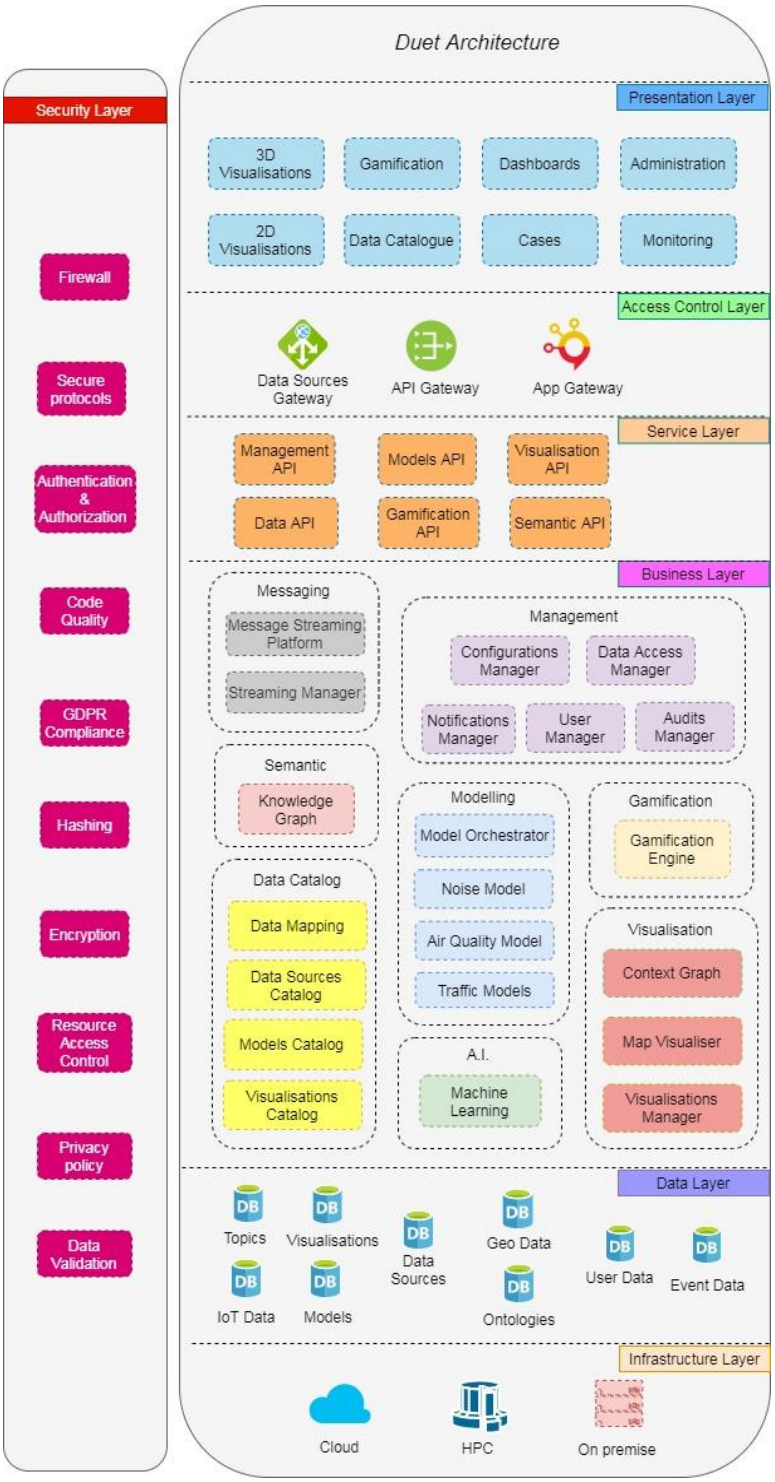

Figure 13: Security Layer in DUET high level architecture

The implemented security measures will cover every underlying asset in the DUET ecosystem. These assets include:

- IoT devices
  - Sensors refer to special purpose hardware and related firmware that collect information about environmental conditions, traffic dynamics, etc. and then generate associated quantitative data, which can be processed (e.g., transmitted in real-time), or stored for later usage. Examples of sensors are acoustic sensors, pollution sensors, humidity sensors, accelerometers among others.
  - Actuators that take an electrical input and turn it into physical action, e.g., controlling a system or mechanism.
  - Embedded systems that, apart from sensors and actuators, are equipped with a micro-computer that can run software. An example of devices that contain embedded systems are smart lights.
- External systems containing historical context data
- Network elements (physical devices or virtualized functions), such as routers, gateways, load balancers, hypervisors, virtual machines, etc.
- Computing infrastructure (either hardware or software), such as bare metal, hypervisors, virtual machines, orchestrators, etc.
- Information
  - At rest
  - In transit
  - In use
  - Meta-data
- Middleware, such as event-buses
  - Business Layer components (following the micro-services paradigm or not), such as Data mining, Data processing and computing, with special emphasis on APIs.
  - Presentation components (following the micro-services paradigm or not), such as Data analytics and visualisation, with special emphasis on APIs.

5.1 DUET Security and Privacy Mechanisms

The following table presents the implemented solutions for all security measures of Section 4. We describe the component or process that is currently in place to cover every measure. Given the fact that we are still at the early stage of development and integration has just started, the table also presents the concrete plans to address measures that haven't been realised yet. Having said that, it must be also noted that early implementations described hereafter will be further refined and specialised in the next development phase.

Security Measure	Description of Implemented Solution
TC1, TC7, TC28, TC31	DUET will implement the DUET Identity and Access Mechanism which will define the platform roles, access rights and policies that will govern authentication and authorisation in the platform.
TC2	HTTPS and other secure protocols (e.g. MQTTS) will be applied to all communications and APIs (internal and external) within the platform.

TC3, TC31	Password recovery mechanisms in all components of the platform implement best practices
TC4, TC12, TC13, TC23, TC24	DUET plans to have a portable solution easily deployable to cloud infrastructures. Therefore, protection from DdoS attacks, load balancing and scaling to support increased traffic are inherent advantages of a cloud deployment.
TC5, TC6	Credentials in DUET's IAM will be stored securely, hashed and salted.
TC8, TC9, TC10, TC11	The DUET Data Catalogue and underlying Knowledge Graph will provide a detailed representation of data sources which will allow consistent identification of the sources and verification of the information they send.
TC14	The DUET API will follow the Open API specification to guarantee quality, collaborations and development effectiveness. Code quality controls will be put in place for individual components and the respective results will be reported (e.g. code vulnerabilities, technical debt, etc)
TC15	DUET Catalogue will periodically check the health of connected sources
TC11, TC16, TC17, TC22, TC34	DUET will put in place an Audit Service which will log all actions taking place in the platform. The specific actions will be defined in the course of development of the first prototype. Collected information will allow periodical audits of the various subsystems.
TC18, TC19, TC20	Work in WP1 (D1.1 [3]) will drive the requirements regarding personal data. A solid privacy policy will be in place so that usage of any data is transparent to the users and potential personal data usage is kept at a minimum level. Moreover, the data management plan of the project will be documented in D8.3 [4] and will be continuously updated in the iterations of this report. Paragraph 5.3 below elaborates on the implemented mechanisms for data privacy in DUET.
TC21	If any personal data will be required for use within the platform, a mechanism to allow deletion of them will be put in place (Section 5.3).
TC22	DUET aims at providing a Digital-Twin to simulate the environment of a smart city and help stakeholders make decisions based on data. Therefore no risks for physical damage exist.
TC25	Task 3.6 with D3.10 and D3.11 [5] describe the security specifications and the relevant implemented measures.
TC26	A microservices-based architecture has been designed for the DUET platform.
TC27, TC32	The technical partners have established a regular communication process (technical meetings) and a common planning scheme following the agile methodology (sprints and epics). Teams having access to the relevant task in the project's task management tool (Jira) are defined and all members are aware of their duties. Internal team organisation is managed by the respective project managers. Github repositories and a Docker hub have been set up.
TC29, TC30, TC31	Code quality controls, individual component documentation and development guidelines will be employed by DUET to ensure secure development of software components.
TC22, TC33, TC32	Automation in deployment will be actively seeked during the integration process as reported in D5.1 [6].
TC32	A configuration service will allow the controlling of a DUET platform instance.

Table 3: DUET security & privacy implementation

5.2 DUET Identity and Access Management

One of the important developments towards securing the DUET platform and processes is a mechanism that will provide Identity management and Access Controls based on user roles and security policies. The DUET

Identity and Access Management (DUET IAM) will handle these tasks by offering access control and identity management delegation to all layers of the platform so as to allow for auditing of data access and support conditional access to data and generated analytics.
Looking for a solution that can accommodate a central point for user and role management in various applications while offering features like Single Sign On (SSO), standard protocols for authorization (e.g. Oauth 2.0) and a wide coverage of supported applications and services we decided to use Keycloak[11] as the basis for DUET IAM. Keycloak major features include:

- Single-Sign On, which means that once logged-in to Keycloak, users don't have to login again to access a different application.
- Identity Brokering and Social Login, supporting all the major social networks as authentication providers.
- User Federation, enabling linking with existing authentication services, e.g. LDAP
- Client Adapters, offering out-of-the-box integration with popular platforms and programming languages
- Administration Console, which offers a user interface for managing the Keycloak server configuration, access policies, user roles, etc.
- User Account Console, which offers users a graphical interface to manage their account
- Standard Protocols implementation, namely OpenID Connect, OAuth 2.0, and SAML.

Based on the above features and an active community of supporters, Keycloak seems to be the best fit for a platform such as DUET. We will utilise existing client adapters for the various components of DUET's architecture (e.g. the "SSO Authentication with Keycloak" extension can be used to integrate with CKAN which will serve as DUET's Data Catalogue as described in D3.8 [7]) or implement additional modules based on the standard authentication protocol specifications which are natively supported by Keycloak.

5.3 DUET Privacy Mechanisms

Technical Controls TC18, TC19, TC20 and TC21 refer to privacy-protecting measures which are vital for a secure, trustworthy implementation of a Smart City Digital Twin. Therefore, in this paragraph we elaborate on the implemented methods to ensure privacy protection and the followed principles that adhere to a privacy-by-design approach.

Data collected in the first period of the project will include datasets already available to the pilots (e.g. traffic models) or datasets offered with an open access licence (D2.3 [8]). Nevertheless, personal data from citizens acting as 'sensors' (i.e., following the crowd-sourcing paradigm) is also envisaged and thus associated privacy requirements must be tackled. To this end, DUET's first point of interaction with data publishers will play an important role. The DUET Data Catalogue, based on CKAN, will offer the required data source registration capabilities using the DCAT vocabulary as analytically described in D3.8. DCAT supports the attribution of data and metadata to various participants of the publishing process and also supports the association of rights and licenses with cataloged Resources, i.e. data sources. Furthermore, CKAN's fine grained administration control in conjunction with DUET's IAM policies will satisfy single users' data protection requirements:

[11] https://www.keycloak.org/

- Personal data dissociation will be ensured since no personal data will be required to register a new data source. A unique system identifier will be the only information kept by the system to allow the connection of the data source with the rest of the components of the platform.
- Anonymity will be also maintained through DUET since users won't need to provide their real name in order to register to the system and will be able to use the majority of the offered functionality without ever providing it (different access rules might apply for license-protected or business critical datasets).
- Informed Consent will be requested by the platform by explicitly asking the users if they agree to linking their data source or providing any other data to be used by the platform. An analytical data protection notice will be also available containing the ways that requested information is used under the relevant privacy laws and must be up-to-date with any possible changes.
- Full user control over connected data sources will be guaranteed via the implemented Management Component (D5.1) which will give users the capability to add, edit, change any parameter and also completely delete their data sources.
- Apart from connected data sources, User profiles will be also created, managed and deleted by their owners without any intervention by DUET. Especially for the deletion of a user profile and its associated data (i.e. right to be forgotten), no official reason or prior notification will be requested by DUET in the case where a user would like to remove their profile. The platform will allow cascaded deletion of the stored data which will be primarily performed by the system administrators upon request of the user and could potentially take effect automatically, depending on the final implementation details of the prototype.

6. Conclusion

This deliverable describes the first version of the security and privacy model of DUET. An initial assessment of the challenges and concerns regarding cybersecurity in a Digital Twin environment have been performed. This resulted in the definition of the DUET Threat Taxonomy. Based on this taxonomy, we defined a set of Security Requirements that cover the basic aspects of cybersecurity in the Smart City and Digital Twin contexts: Confidentiality, Integrity, Availability, Accountability and Authenticity.
We then identified the available measures that DUET should undertake in order to satisfy the security requirements. These measures refer to good practices on software development, process management and information handling. They form a technical checklist which will drive the implementation of the security mechanisms of DUET and contribute to the security-by-design concept.

Finally, the deliverable reports on the executed actions, defined processes and clear roadmaps towards realising all the identified security measures. This way, we present a clear definition of the implementation status and upcoming steps for the full deployment of the Multi-Layered Security and Privacy mechanisms of DUET.

7. References

[1] ENISA, Baseline Security Recommendations for IoT, November 20, 2017, https://www.enisa.europa.eu/publications/baseline-security-recommendations-for-iot

[2] ENISA, Good Practices for Security of IoT - Secure Software Development Lifecycle, November 19, 2019, https://www.enisa.europa.eu/publications/good-practices-for-security-of-iot-1

[3] DUET project, Deliverable D1.1: “Legal Landscape and Requirements Plan”, March 2020

[4] DUET project, Deliverable D8.3: “Data Management and Modeling Plan”, May 2020

[5] DUET project, Deliverable D3.11: “Multi Layered security model specification”, Nov 2021

[6] DUET project, Deliverable D5.1: “System Architecture & Implementation Plan”, Nov 2020

[7] DUET project, Deliverable D3.8: “Digital Twin data broker specifications and tools v1”, Nov 2020

[8] [DUET project, Deliverable D2.3: “Final list of user requirements for the DUET solution”, July 2020