Deliverable

D3.11 Multi Layered security model specification (final)

Project Acronym:	DUET	
Project title:	Digital Urban European Twins	
Grant Agreement No.	870697	
Website:	www.digitalurbantwins.eu	
Version:	1.0	
Date:	30 November 2021	
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	13.10.2021	Leonidas Kallipolitis	AEG	Updates with respect to D3.10
0.2	16.11.2021	Leonidas Kallipolitis	AEG	First Internal Draft consolidating partners input
0.3	23.11.2021	Leonidas Kallipolitis, Stathis Dimakos, Andreas Alexopoulos, Kostis Michalis	AEG	Draft version for internal and external review
0.4	28.11.2021	Gert Vervaet, Thanasis Dalianis, Philippe Michiels, Yannis Charalabidis	AIV, ATC, IMEC, (external)	Internal & External Review Comments
1.0	30.11.2021	Leonidas Kallipolitis	AEG	Final Version

Table of Contents

Executive Summary 5

1. Introduction 6

2. Security in Smart Cities and Digital Twins 7

2.1 Implications and Concerns 7

2.2 DUET Threat Taxonomy 8

2.3 High-level considerations towards a secure Digital Twin 12

- *Automated and Secure Deployment of IoT Devices* 12
- *Access Control Management and Informed Consent* 12
- *Privacy-Preserving Data Analytics according to current Security and Privacy Regulations* 12
- *Cybersecurity Awareness* 13

3. DUET Security Requirements 14

3.1 DUET Run-time Security Requirements 14

3.1.1 Confidentiality 14

3.1.2 Integrity 15

3.1.3 Availability 15

3.1.4 Accountability 16

3.2 Security requirements in DUET’s Software Development Life Cycle (SDLC) 16

3.2.1 Confidentiality 16

3.2.2 Integrity 17

3.2.3 Availability 17

3.2.4 Accountability 18

3.2.5 Authenticity 18

4. DUET Security Measures 19

4.1 Run-time Authentication (when connecting to a DUET backend service, visualisation system or sensor) 19

4.2 Run-time Authorisation 20

4.3 Secure and trusted communications 21

4.4 Run-time Monitoring and Auditing 23

4.5 Data protection and compliance 23

4.6 Software Development Lifecycle Security 24

4.6.1 Plan 24

4.6.2 SDLC Authentication and Authorisation 25
4.6.3 Secure Development 26
4.6.4 SDLC Monitoring and Auditing 27
5. DUET Multi-layered Security 30
5.1 Implementation of TCs by DUET components 30
5.2 Implementation of TCs by DUET as a platform 34
5.3 DUET Identity and Access Management 35
5.4 DUET Privacy Mechanisms 36
6. Conclusion 37
7. References 38

Tables

Table 3.1 List of security and privacy requirements for DUET Digital Twins 18
Table 5.1 Security measures mapping (part1) 30
Table 5.2 Security measures mapping (part2) 31
Table 5.3 DUET overall security & privacy implementation 34

Table of Figures

Figure 1 The DUET Threat Taxonomy 11
Figure 2 ICT Roles in the Development Environment 17
Figure 3 Mapping of run-time authentication measures and security objectives 20
Figure 4 Mapping of run-time authorisation measures and security objectives 21
Figure 5 Mapping of communication measures and security objectives 22
Figure 6 Mapping of monitoring/auditing measures and security objectives 23
Figure 7 Mapping of data protection measures and security objectives 24
Figure 8 Mapping of SDLC planning measures and security objectives 25
Figure 9 Mapping of SDLC authentication/authorisation measures and security objectives 26
Figure 10 Mapping of SDLC development measures and security objectives 27
Figure 11 Mapping of SDLC monitoring measures and security objectives 28
Figure 12 Taxonomy of DUET security measures and respective TCs 29

Executive Summary

This report provides an update to the first version submitted one year ago, namely D3.10. Sections 2, 3 and 4 have been retained due to their core role in specifying the security and privacy model of DUET. They present the DUET Threat Taxonomy for Digital Twins and Smart Cities, the Security Requirements and the Security Measures respectively.

In the current version of the document, the Technical Controls (TCs) that drive the implementation of the identified measures are mapped with the components of the current version of the DUET platform, namely the Open Beta Release (Nov 2021). Moreover, processes and functionalities pertaining to the overall platform and not to specific components are also important in addressing several of the TCs and therefore described separately. Finally, the mechanisms and design approaches tackling privacy considerations are described so as to present how the multi-layered security and privacy model is actually realised in DUET. The document concludes with proposed future steps that will safeguard the privacy and security stance of DUET.

1. Introduction

This deliverable constitutes an update of D3.10 [1] which was submitted at the end of the first year of the project. The rationale is to provide updates and progress on the implementation of the security and privacy measures reported in D3.10 and how the received experts' recommendations after the first project review have been taken into consideration during the second development phase of the project, towards the open beta version of the DUET platform. Therefore, the content of the first sections of D3.10 (Sections 2-4) has been preserved in this document, whereas Section 5 has been updated to reflect how the components of DUET have addressed the security and privacy requirements to realise the foreseen specifications. The goal remains unchanged, DUET must define and put in place 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 sections of D3.10 that have been retained in the current document firstly refer to 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, the DUET Threat Taxonomy is defined to serve as a basis upon which relevant measures should be taken in order to realise a secure and privacy-preserving Digital Twin implementation. Section 3 defines 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 and privacy measures in the context of the open beta version of DUET. The description includes a mapping matrix among DUET components and implemented TCs. Being a modular, loosely coupled platform, DUET relies on individual components and services to follow the defined TCs while at the same time making sure that its core applies proper security measures and enforces privacy-preserving techniques. Hence, the envisioned multi-layered security approach is realised in all platform layers and appropriate measures are clearly defined to cover current and upcoming needs. To provide for the latter, this report presents a clearly defined approach to handle emerging needs regarding security and privacy towards the final release of the DUET platform.

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 [2], [3] 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].

Table 1: Threat Taxonomy for DUET Digital Twins

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

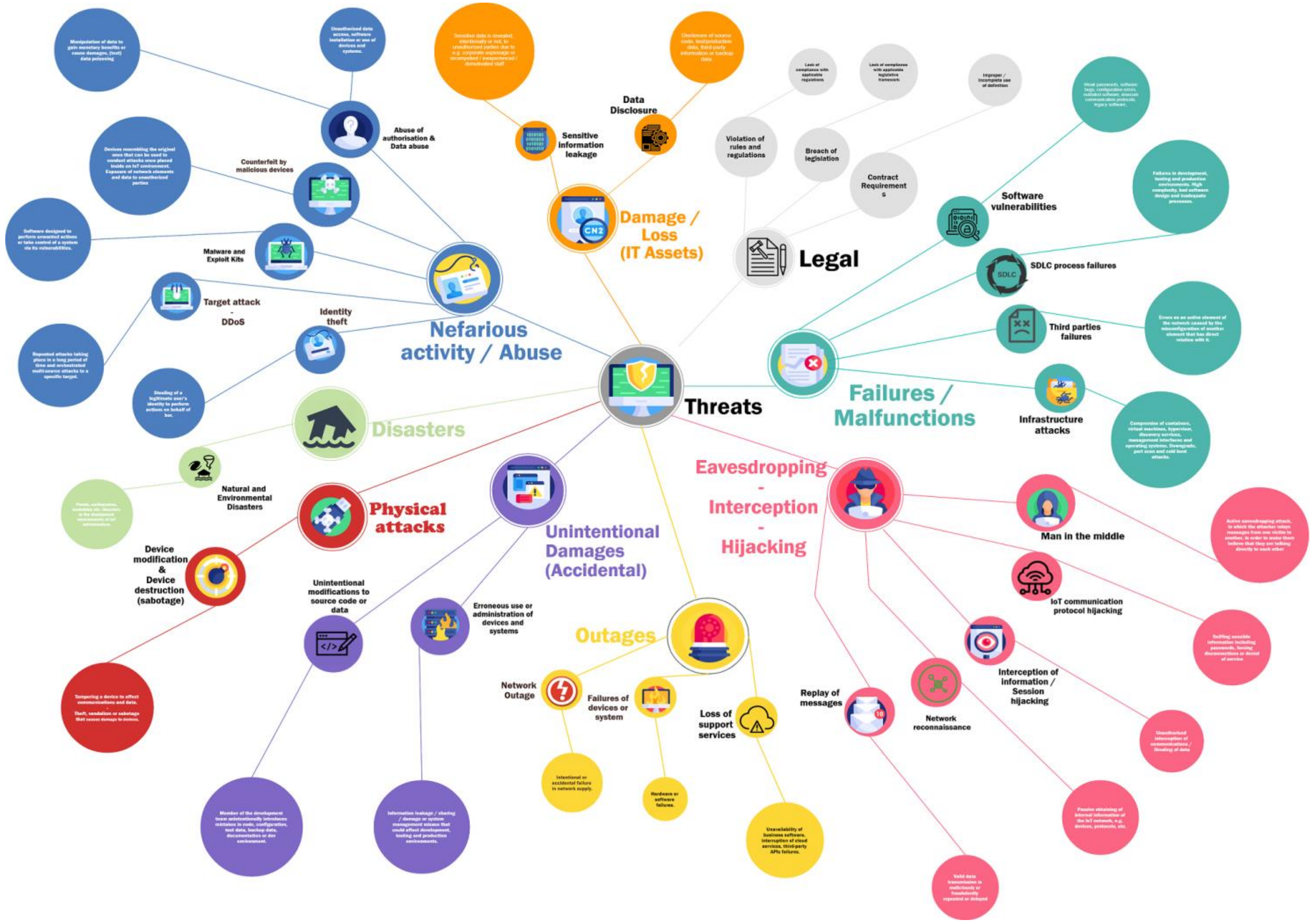

Threats
Nefarious activity / Abuse
Abuse of authorisation & Data abuse
Unauthorised data access, software installation or use of devices and systems.
Manipulation of data to gain monetary benefits or cause damages, (test) data poisoning
Counterfeit by malicious devices
Devices resembling the original ones that can be used to conduct attacks once placed inside an IoT environment. Exposure of network elements and data to unauthorized parties
Malware and Exploit Kits
Software designed to perform unwanted actions or take control of a system via its vulnerabilities.
Target attack - DDoS
Repeated attacks taking place in a long period of time and orchestrated multi-source attacks to a specific target.
Identity theft
Stealing of a legitimate user's identity to perform actions on behalf of her.
Damage / Loss (IT Assets)
Sensitive information leakage
Sensitive data is revealed, intentionally or not, to unauthorised parties due to e.g. corporate espionage or incompetent / inexperienced / demotivated staff
Data Disclosure
Disclosure of source code, test/production data, third-party information or backup data
Legal
Violation of rules and regulations
Lack of compliance with applicable regulations
Breach of legislation
Lack of compliance with applicable legislative framework
Contract Requirement s
Improper / incomplete use of definition
Failures / Malfunctions
Software vulnerabilities
Weak passwords, software bugs, configuration errors, outdated software, insecure communication protocols, legacy software.
SDLC process failures
SDLC
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 & Device destruction (sabotage)
Tampering a device to affect communications and data. - Theft, vandalism or sabotage that causes damage to devices.
Unintentional Damages (Accidental)
Unintentional modifications to source code or data
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.
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.
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
10
Valid data transmission is maliciously or fraudulently repeated or delayed

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

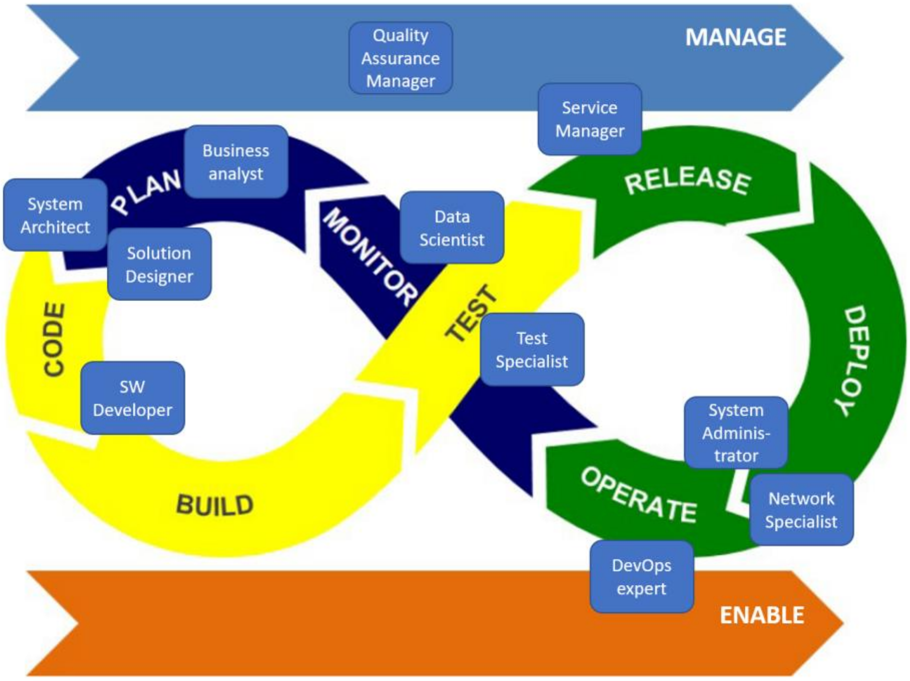


Figure 2 ICT Roles in the Development Environment[9]

- ○ **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:

Table 3.1 List of security and privacy requirements for DUET Digital Twins

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)	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

Co14: Protect sensitive management information of DUET development environment	
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	

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.

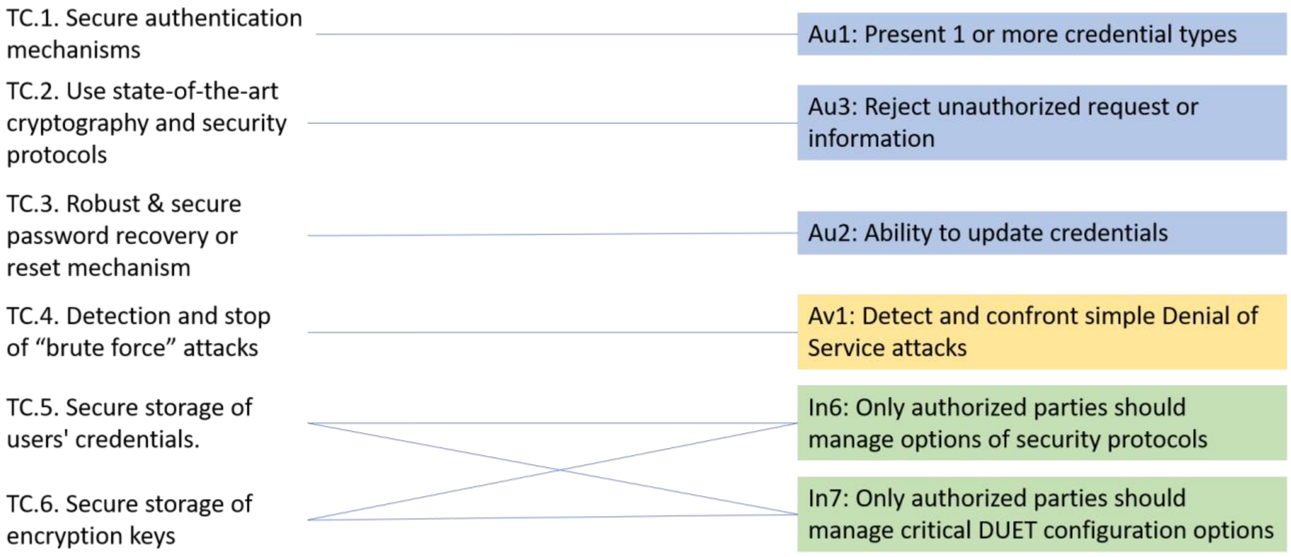
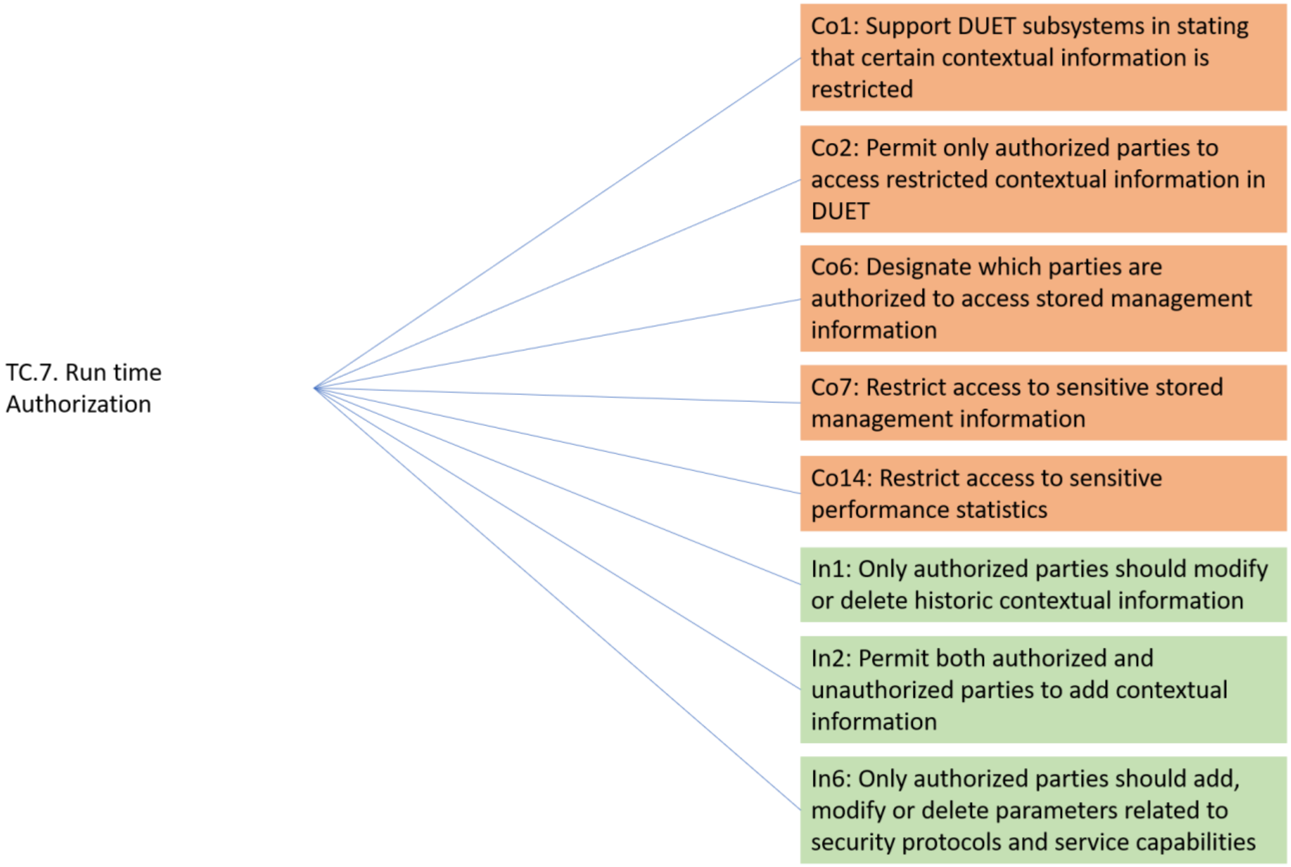

TC.1. Secure authentication mechanisms
TC.2. Use state-of-the-art cryptography and security protocols
TC.3. Robust & secure password recovery or reset mechanism
TC.4. Detection and stop of "brute force" attacks
TC.5. Secure storage of users' credentials.
TC.6. Secure storage of encryption keys
Au1: Present 1 or more credential types
Au3: Reject unauthorized request or information
Au2: Ability to update credentials
Av1: Detect and confront simple Denial of Service attacks
In6: Only authorized parties should manage options of security protocols
In7: Only authorized parties should manage critical DUET configuration options

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.

[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.



TC.7. Run time Authorization

Co1: Support DUET subsystems in stating that certain contextual information is restricted

Co2: Permit only authorized parties to access restricted contextual information in DUET

Co6: Designate which parties are authorized to access stored management information

Co7: Restrict access to sensitive stored management information

Co14: Restrict access to sensitive performance statistics

In1: Only authorized parties should modify or delete historic contextual information

In2: Permit both authorized and unauthorized parties to add contextual information

In6: Only authorized parties should add, modify or delete parameters related to security protocols and service capabilities

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.

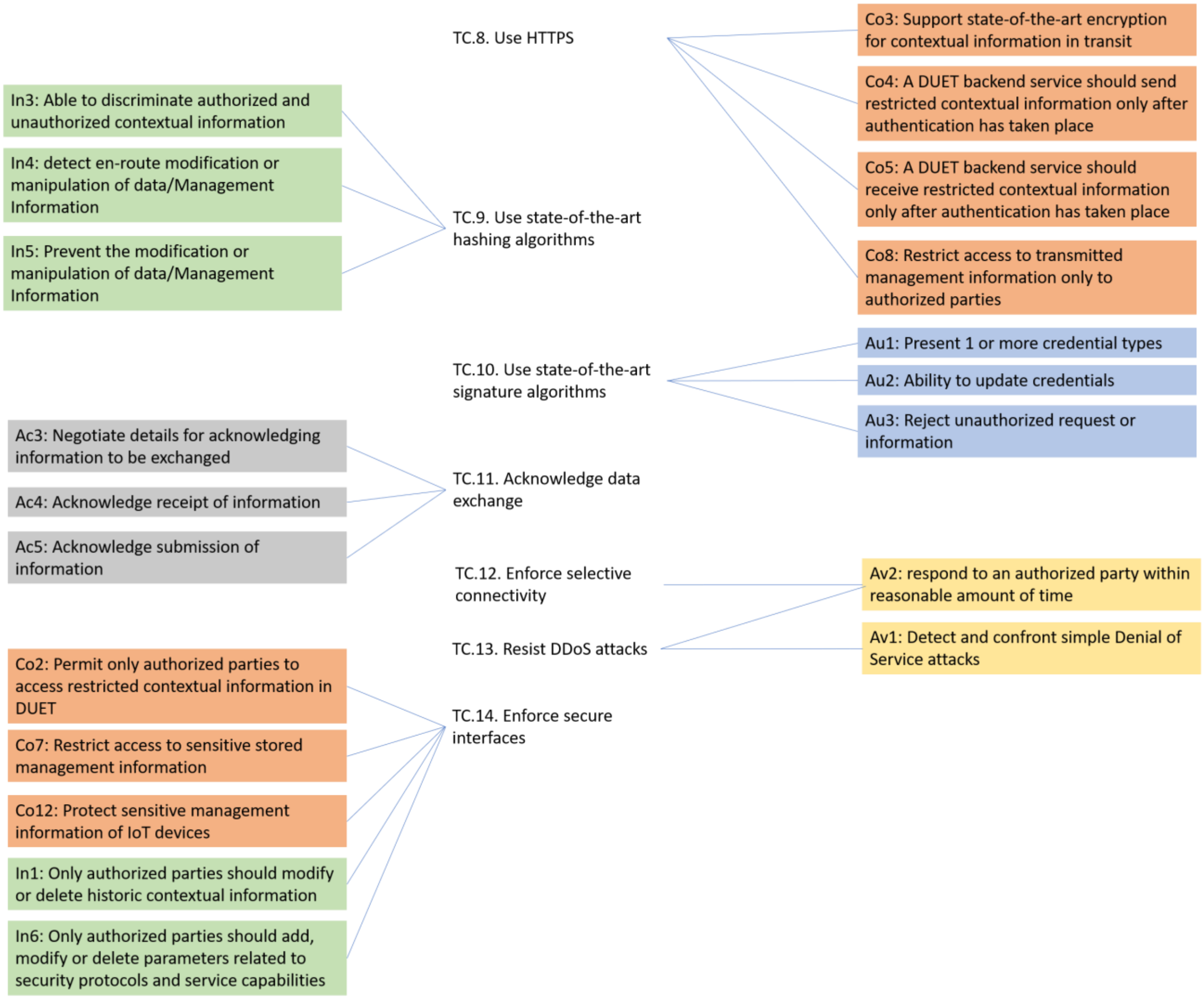

TC.8. Use HTTPS
Co3: Support state-of-the-art encryption for contextual information in transit
Co4: A DUET backend service should send restricted contextual information only after authentication has taken place
Co5: A DUET backend service should receive restricted contextual information only after authentication has taken place
Co8: Restrict access to transmitted management information only to authorized parties
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
TC.9. Use state-of-the-art hashing algorithms
TC.10. Use state-of-the-art signature algorithms
Au1: Present 1 or more credential types
Au2: Ability to update credentials
Au3: Reject unauthorized request or information
Ac3: Negotiate details for acknowledging information to be exchanged
Ac4: Acknowledge receipt of information
Ac5: Acknowledge submission of information
TC.11. Acknowledge data exchange
TC.12. Enforce selective connectivity
Av2: respond to an authorized party within reasonable amount of time
TC.13. Resist DDoS attacks
Av1: Detect and confront simple Denial of Service attacks
Co2: Permit only authorized parties to access restricted contextual information in DUET
Co7: Restrict access to sensitive stored management information
Co12: Protect sensitive management information of IoT devices
In1: Only authorized parties should modify or delete historic contextual information
In6: Only authorized parties should add, modify or delete parameters related to security protocols and service capabilities
TC.14. Enforce secure interfaces

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.

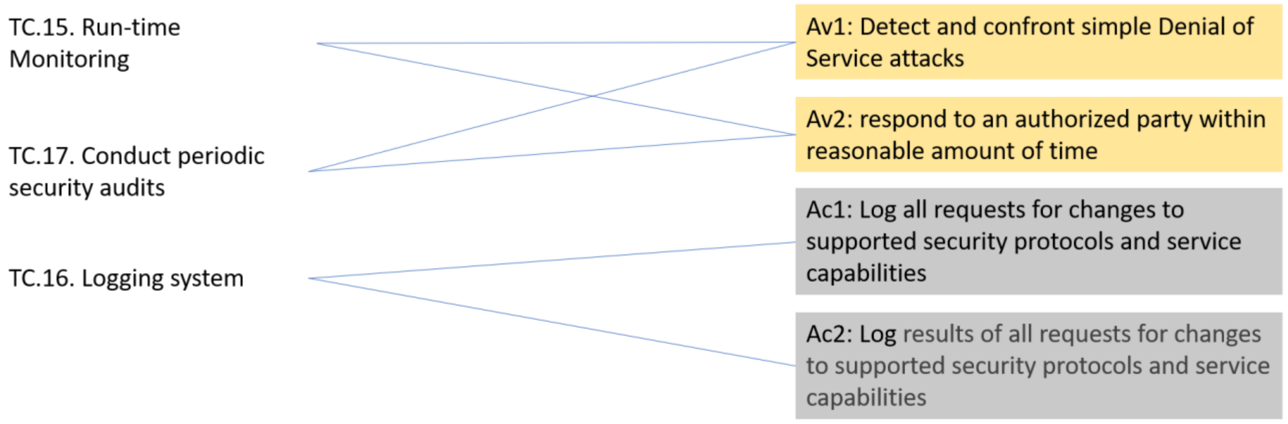

TC.15. Run-time Monitoring
TC.17. Conduct periodic security audits
TC.16. Logging system
Av1: Detect and confront simple Denial of Service attacks
Av2: respond to an authorized party within reasonable amount of time
Ac1: Log all requests for changes to supported security protocols and service capabilities
Ac2: Log results of all requests for changes to supported security protocols and service capabilities

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.

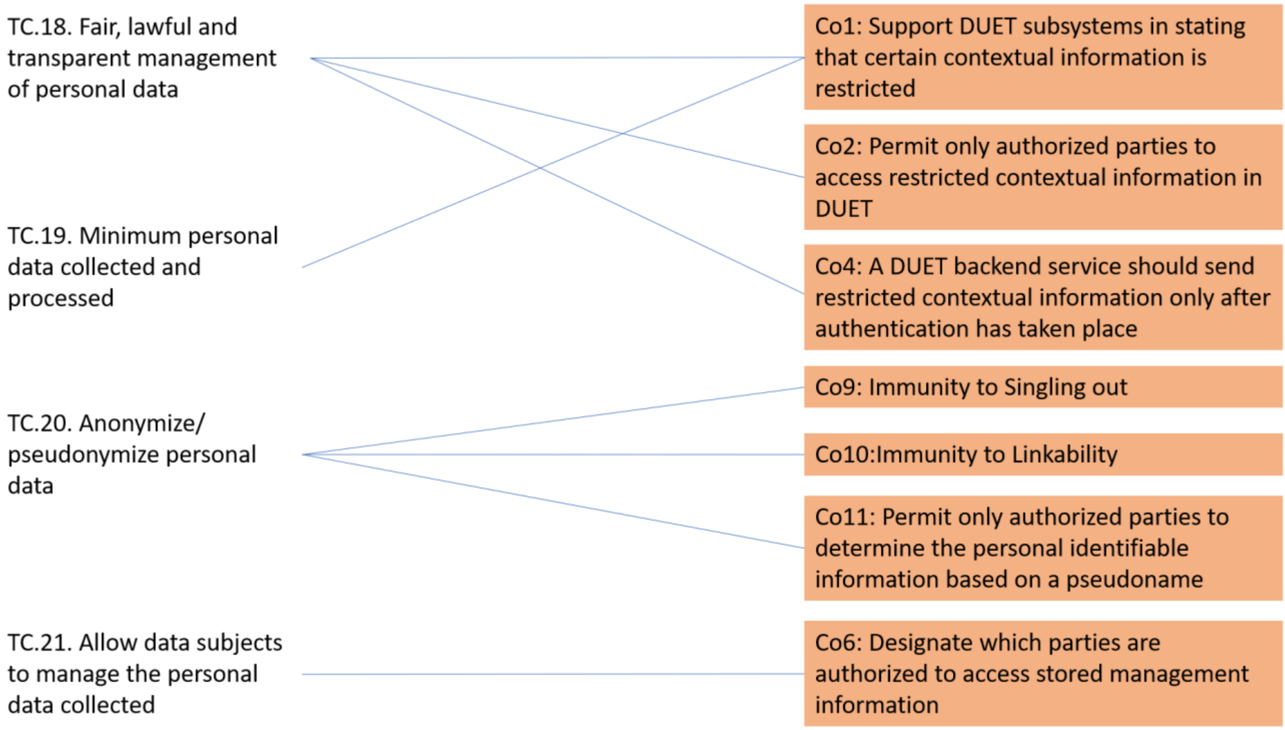

TC.18. Fair, lawful and transparent management of personal data
TC.19. Minimum personal data collected and processed
TC.20. Anonymize/ pseudonymize personal data
TC.21. Allow data subjects to manage the personal data collected
Co1: Support DUET subsystems in stating that certain contextual information is restricted
Co2: Permit only authorized parties to access restricted contextual information in DUET
Co4: A DUET backend service should send restricted contextual information only after authentication has taken place
Co9: Immunity to Singling out
Co10:Immunity to Linkability
Co11: Permit only authorized parties to determine the personal identifiable information based on a pseudoname
Co6: Designate which parties are authorized to access stored management information

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.

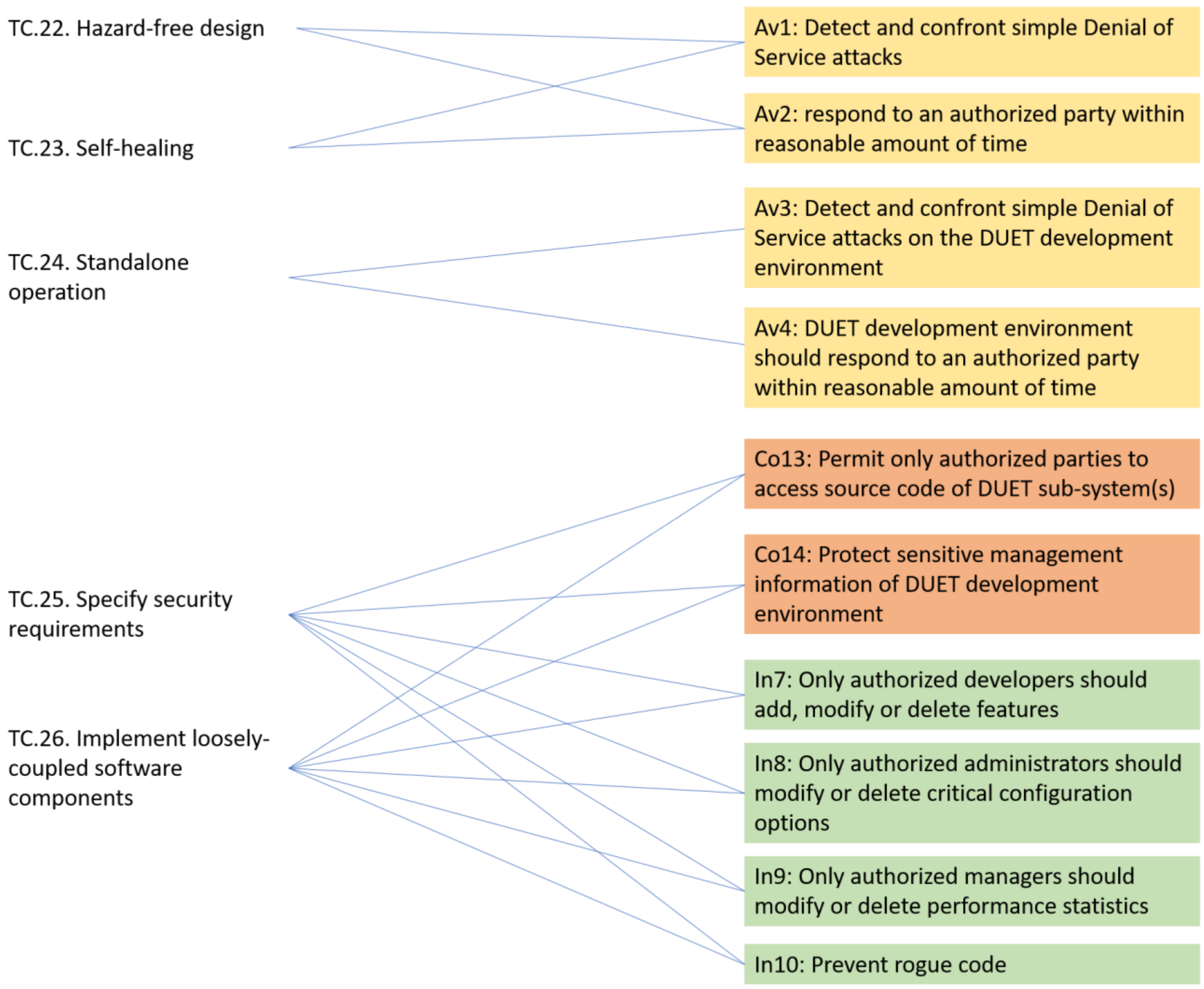

TC.22. Hazard-free design
TC.23. Self-healing
TC.24. Standalone operation
TC.25. Specify security requirements
TC.26. Implement loosely-coupled software components
Av1: Detect and confront simple Denial of Service attacks
Av2: respond to an authorized party within reasonable amount of time
Av3: Detect 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
Co13: Permit only authorized parties to access source code of DUET sub-system(s)
Co14: Protect sensitive management information of DUET development environment
In7: Only authorized developers should add, modify or delete features
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

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.

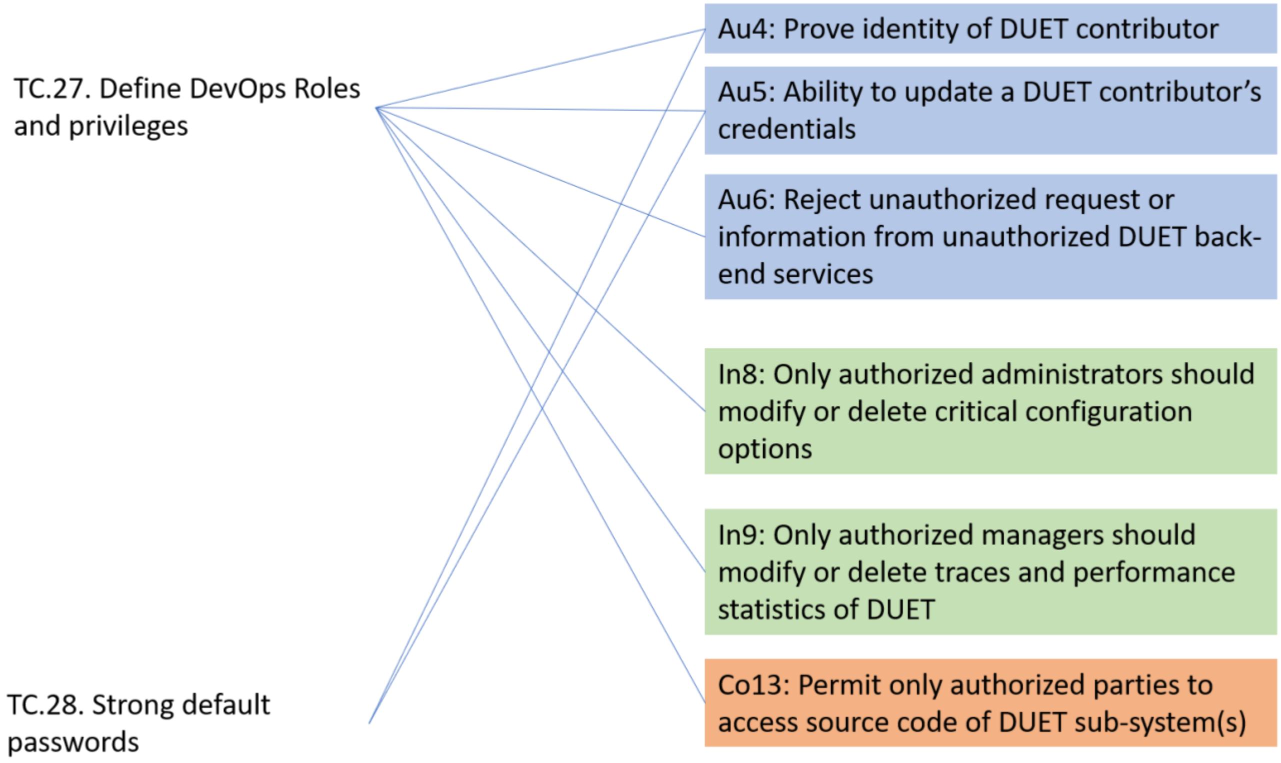

TC.27. Define DevOps Roles and privileges
TC.28. Strong default passwords
Au4: Prove identity of DUET contributor
Au5: Ability to update a DUET contributor's credentials
Au6: Reject unauthorized request or information from unauthorized DUET back-end services
In8: Only authorized administrators should modify or delete critical configuration options
In9: Only authorized managers should modify or delete traces and performance statistics of DUET
Co13: Permit only authorized parties to access source code of DUET sub-system(s)

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.

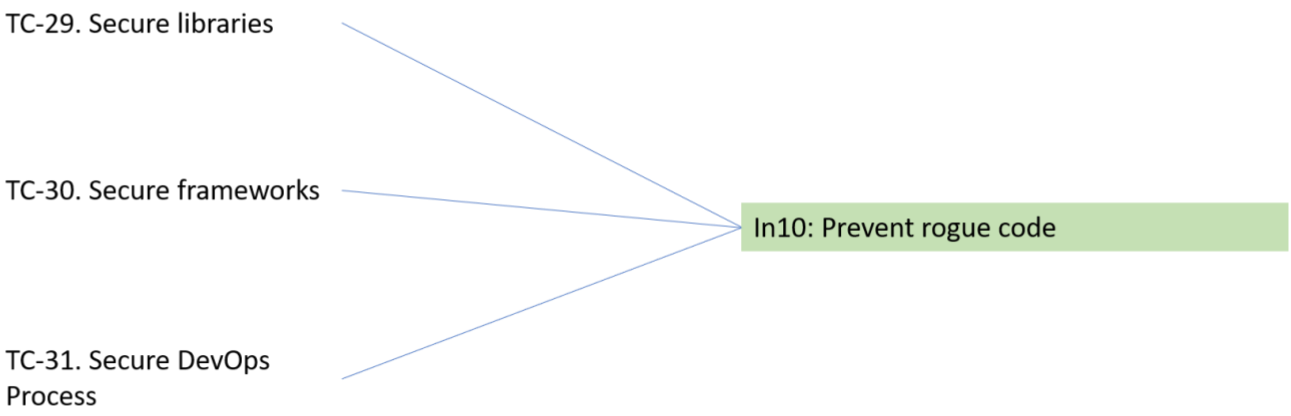


Figure 10 Mapping of SDLC development measures and security objectives

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.

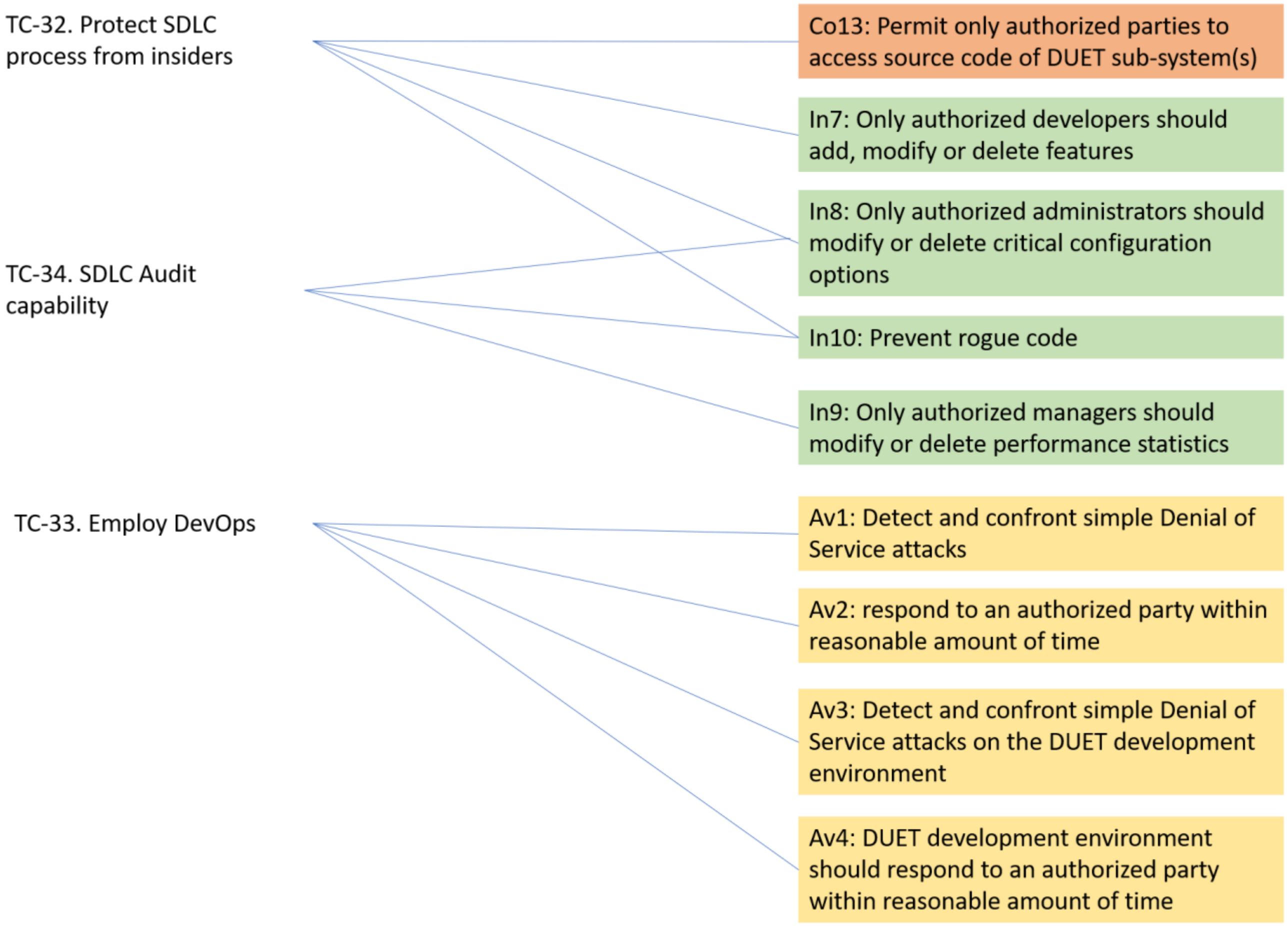

TC-32. Protect SDLC process from insiders
TC-34. SDLC Audit capability
TC-33. Employ DevOps
Co13: Permit only authorized parties to access source code of DUET sub-system(s)
In7: Only authorized developers should add, modify or delete features
In8: Only authorized administrators should modify or delete critical configuration options
In10: Prevent rogue code
In9: Only authorized managers should modify or delete performance statistics
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

Figure 11 Mapping of SDLC monitoring measures and security objectives

The following map depicts the complete tree of the DUET Security Measures and relevant technical controls (TCs):

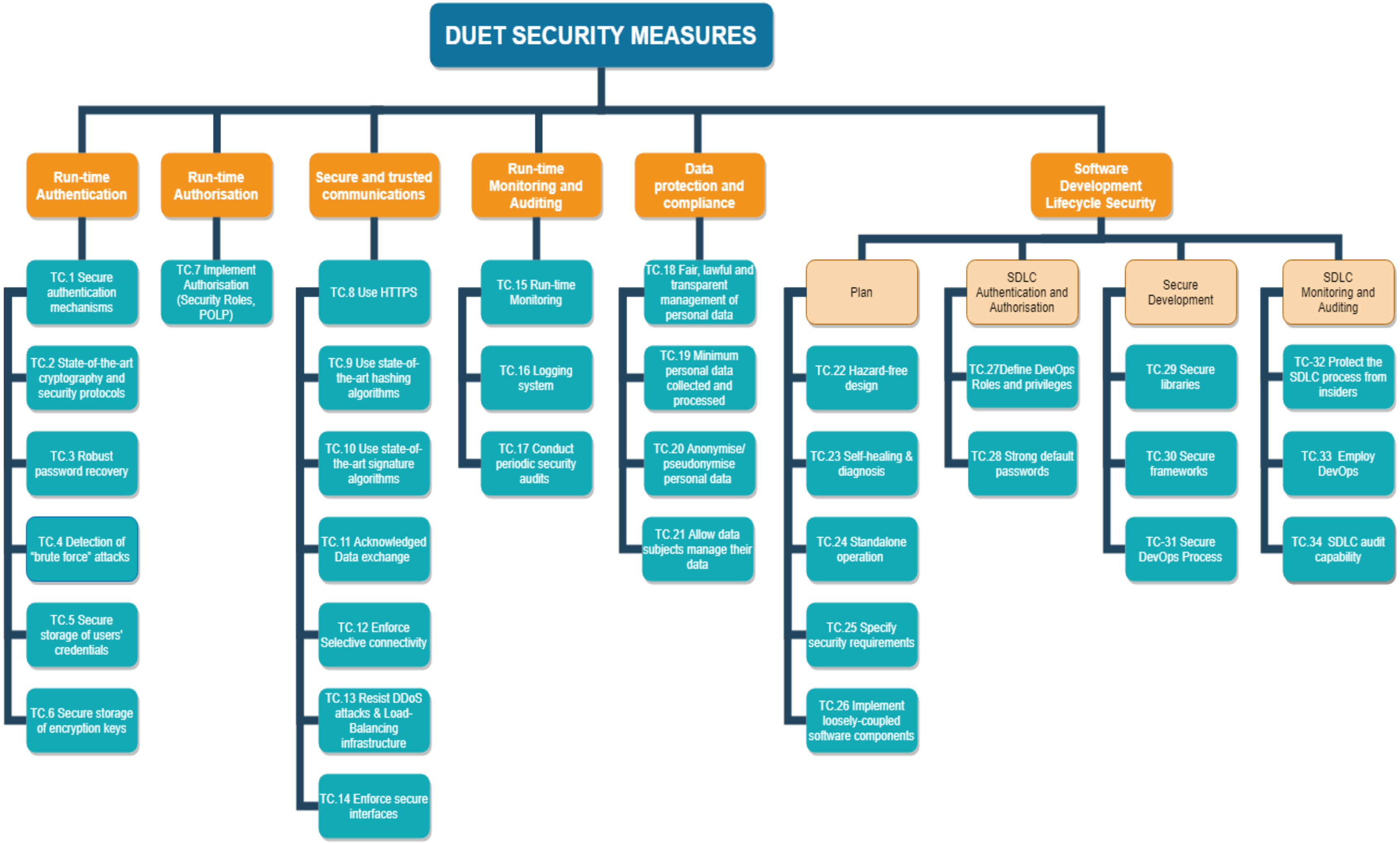

DUET SECURITY MEASURES
Run-time Authentication
Run-time Authorisation
Secure and trusted communications
Run-time Monitoring and Auditing
Data protection and compliance
Software Development Lifecycle Security
TC.1 Secure authentication mechanisms
TC.2 State-of-the-art cryptography and security protocols
TC.3 Robust password recovery
TC.4 Detection of "brute force" attacks
TC.5 Secure storage of users' credentials
TC.6 Secure storage of encryption keys
TC.7 Implement Authorisation (Security Roles, POLP)
TC.8 Use HTTPS
TC.9 Use state-of-the-art hashing algorithms
TC.10 Use state-of-the-art signature algorithms
TC.11 Acknowledged Data exchange
TC.12 Enforce Selective connectivity
TC.13 Resist DDoS attacks & Load-Balancing infrastructure
TC.14 Enforce secure interfaces
TC.15 Run-time Monitoring
TC.16 Logging system
TC.17 Conduct periodic security audits
TC.18 Fair, lawful and transparent management of personal data
TC.19 Minimum personal data collected and processed
TC.20 Anonymise/ pseudonymise personal data
TC.21 Allow data subjects manage their data
Plan
SDLC Authentication and Authorisation
Secure Development
SDLC Monitoring and Auditing
TC.22 Hazard-free design
TC.23 Self-healing & diagnosis
TC.24 Standalone operation
TC.25 Specify security requirements
TC.26 Implement loosely-coupled software components
TC.27Define DevOps Roles and privileges
TC.28 Strong default passwords
TC.29 Secure libraries
TC.30 Secure frameworks
TC-31 Secure DevOps Process
TC-32 Protect the SDLC process from insiders
TC.33 Employ DevOps
TC.34 SDLC audit capability

Figure 12 Taxonomy of DUET security measures and respective TCs

5. DUET Multi-layered Security

The Open Beta Release of DUET integrates most of the envisaged technologies and components supporting the Digital Twin concept, following the plan of implementation milestones presented in D5.1 [9]. Therefore, there is currently a clear approach with regards to security and privacy measures implementation in the platform. The following paragraphs present a detailed mapping of the components with the TCs identified in the previous section of the deliverable, the approach every component follows to address these TCs and how emerging needs will be safeguarded in view of the final release of the DUET platform.

5.1 Implementation of TCs by DUET components

Using the DUET Security Measures Taxonomy depicted in Figure 12, a matrix which maps components and TCs was compiled in order to clearly present how the multi-layer security and privacy is achieved. For readability reasons, the matrix has been split in two parts, the first one displaying the TCs referring to Authentication/Authorisation, Communications, Monitoring, and the second one referring to TCs addressing Data protection and SDLC best practices.

Table 5.1 Security measures mapping (part1)

Sec Measure / Component	TC1	TC2	TC3	TC4	TC5	TC6	TC7	TC8	TC9	TC10	TC11	TC12	TC13	TC14	TC15	TC16	TC17
API Gateway	X	X		X			X	X			X		X	X			
Message Broker	X	X		X			X	X			X		X	X			
Data Sources Gateway		X						X			X		X		X	X	
Data Catalog		X						X			X		X		X	X	
Users Manager	X	X	X		X	X	X	X	X	X				X			
Cases Manager	X	X		X			X	X					X	X			
Map Visualiser																X	
KUL Traffic model																X	
P4A Traffic model																X	
Noise model																X	
Air pollution model																X	

Message Streaming Platform (Kafka)		X					X	X									
DUET Landing pages	X	X	X	X				X									
DUET Platform pages	X	X	X	X				X									
DUET IAM (Keycloak)	X	X	X	X	X		X	X	X								
Monitoring Tool								X			X	X		X	X	X	X

Table 5.2 Security measures mapping (part2)

Sec Measure / Component	TC18	TC19	TC20	TC21	TC22	TC23	TC24	TC25	TC26	TC27	TC28	TC29	TC30	TC31	TC32	TC33	TC34
API Gateway						X	X		X							X	
Message Broker						X	X		X					X			
Data Sources Gateway	x					X	X		X					X			
Data Catalog	x					X	X		X					X			
Users Manager	x	x		x		X	X		X					X			
Cases Manager						X	X		X					X			
Map Visualiser		X							X								
KUL Traffic model		X				X			X								
P4A Traffic model		x				X			X					X			
Noise model		X							X								
Air pollution model		X							X								
Message Streaming Platform (Kafka)						X			X								
DUET Landing pages	x	x		x		X			X								
DUET Platform pages	x	x		x		X								X			
DUET IAM (Keycloak)				X		X			X		X			X			
Monitoring Tool							X	X	X	X	X	X	X	X	X		X

The way in which the components satisfy the TCs is described in the following paragraphs.

The *API Gateway* is the single point of access for all the APIs deployed in the platform. It employs access control through secure protocols and hides the internal connection points of the APIs. It is implemented with the help of the Kong Gateway and Kubernetes Ingress, so in case of a failure Kubernetes tries to restore it. Azure where all components are deployed, protects the Gateway through its firewall and security measures it applies.

The *Message Broker* is a component that acts as an abstract layer in front of the Message Bus of the platform which is implemented on Kafka. It assists the communication between the different components within or outside the platform. It also allows secure communication and receives the same security measures as in the case of the API Gateway, since it follows a similar deployment strategy.

The *Data Sources Gateway* is a collection of components. There is one component that faces the client and serves responses to data requests. It uses multiple backends which fetch and transform the data appropriately. The configuration of these backends is done in the Data Catalog.

The *Data Catalog* allows for the registration of data sources. Some aspects of the configuration are picked up automatically by other components, so that for instance a query for data will be picked up by the right component and fulfilled. It is built as an API on top of a mongodb database.

The *Users Manager* provides a common interface for the addition, update and retrieval of users and roles in the system and acts as an abstract layer in front of Keycloak. It stores a minimum set of data for the user (email, password and intentions) in a secure database using up-to-date encryption and hashing algorithms, that can be accessed only by administrators. The passwords of the users are hashed and recovery is only possible via a reset mechanism available only to users themselves. Users can also delete their account and any associated information. All communications with external clients are performed through the API Gateway using https.

The *Cases Manager* allows the users of the system to manage pilot cases, that is, a set of datasources and models that can be used to solve a problem in the city. It is deployed in Kubernetes and all communication with external components is done through the API Gateway. All data are stored in a secure database and cases can be only accessed by their creators, unless they are marked as public.

The Map Visualiser is based on VC Map, a commercial product following strict security policies. Since the VC MAP is a web application deployed on a web server, all of the security things are controlled by the web server itself. Authentication and user management is provided by the Duet platform. Communication with DUET components takes place via the API Gateway, thus ensuring secure data exchange and privacy maintenance since visualised data are retrieved via the Data Sources Gateway which is responsible to enforce access control.

The *KUL traffic model* is integrated as a docker container running inside of the platform. It restarts upon failure and communicates with other components only indirectly via Kafka. Data is passed on by reference using a blob storage service and progress of the computation, including possible errors, is logged. There's no real personal data being processed since all the data that the model receives from the platform are infrastructure changes and a reference to the relevant user scenario, each of which eventually gets overwritten since the model only stores the results for a while until they can be processed by the platform (and visualised in the Map Visualiser).

P4A's Traffic Modeller (TraMod) runs on P4A's servers with connection to DUET via docker container inside Kafka. Base traffic model (network, OD matrix, traffic generators) is located alongside the TraMod itself on the P4A' s server. This base traffic model for DUET runs independent from the model used for other P4A clients. There is no personal data processed. Base models used to select roads to close in Map Visualiser have been passed to VCS as JSON file before the model could be run.

P4A´s Noise model uses the same approach as TraMod with notable difference that core of the technology is based on an pre-existing open source project NoiseModelling (developed by the DECIDE team from the Lab-STICC (CNRS) and by the Mixt Research Unit in Environmental Acoustics)

TNO's Air pollution model is run at an, external to DUET, infrastructure which only exposes a single endpoint to receive requests for model running. This endpoint only receives requests by the Message Broker and likewise sends the results back to it.

The Message Streaming Platform is implemented based on Kafka and facilitates the messaging infrastructure for the communication of the components of the system. It implements authorisation and supports secure protocols.

The *Message Streaming Platform* is implemented based on Kafka and facilitates the messaging infrastructure for the communication of the components of the system. It implements authorisation and supports secure protocols.

The *DUET Landing pages* form a set of webpages implemented in Wordpress CMS that act as an entry point to DUET platform for the users and implements a number of security measures which are constantly reviewed and updated[11]. The pages are served via https and the only requirement for a user to register is an email and a password that are stored securely. The host of the webpages also provides security measures like a firewall. Plugins that prevent attacks and enable additional authorisation constraints are also installed (e.g. WordFence[12]). Frequent updates of Wordpress and all installed plugins are performed in order to receive all new security fixes and maintain an up-to-date installation.

The *DUET Platform pages* allow the users to access the main functionalities of the platform. They run under https and also require a minimal set of data for the user registration which are stored in a secure database. The platform pages are also deployed in Kubernetes so as the rest of the components have a self-healing process.

The *DUET IAM* component is based on Keycloak (see Section 5.3 below).

The *Monitoring Tool* allows real-time monitoring of the DUET services. It is introduced in the architecture to cover the needs of platform administrators and component owners who wish to have a central dashboard

[11] https://wordpress.org/about/security/

[12] https://www.wordfence.com/

offering monitoring capabilities for DUET components and services. It can be also used as the initial point for the detection of potentially harmful/malicious incidents concerning security or privacy of the platform. It is based on the Advanced Visualisation Toolkit[13]. The tool securely connects to the Message Broker and retrieves information on the execution results of various services, e.g. a model's output status. The tool is built using opensource libraries and frameworks with active community involvement, i.e. Angular, Node.js and has a clearly defined update path following the major releases of the used frameworks. Moreover, code is organised in private git repos with access permissions based on the dev team's roles. SonarQube is used for code quality checking.

5.2 Implementation of TCs by DUET as a platform

Apart from the way that individual components address them, a number of TCs is realised via the general setup and processed followed during the development of the DUET platform. The following table is an updated version of the one provided in D3.10 and aims to list the general approach of DUET to some of the TCs referring to broader security measures.

Table 5.3 DUET overall security & privacy implementation

Security Measure	Description of Implemented Solution
TC2	HTTPS and other secure protocols are applied to all communications and APIs (internal and external) within the platform.
TC4, TC12, TC13, TC17, TC23, TC24	DUET plans to have a portable solution easily deployable to cloud infrastructures. Azure and Kubernetes offer respectively the resources and orchestration of the platform. Therefore, protection from DdoS attacks, load balancing and scaling to support increased traffic are inherent offerings of the selected cloud deployment.
TC14	The DUET APIs follow the Open API specification to guarantee quality, collaborations and development effectiveness. Code quality controls are partially in place or planned for individual components and the respective results will be reported (e.g. code vulnerabilities, technical debt, etc)
TC18, TC19, TC20	Work in WP1 (D1.1 [4], D1.2 [5], D1.3 [6] and D1.5 [7]) lays the foundation of lawful and ethical compliance with respect to data management and the principles that pilot cities and in parallel DUET as a platform must follow. A solid privacy policy is in place so that usage of any data is transparent to the users and personal data usage is kept at a minimum level. Moreover, the data management plan of the project has been delivered D8.3 **Error! Reference source not found.** and is continuously updated to reflect latest s tatus of DUET datasets in use. The live version of this document is available here. Paragraph 5.4 below elaborates on the implemented mechanisms for data privacy in DUET.
TC21	Personal data that are required to use the platform, i.e registration details, are fully managed by end-users who have the right to delete them at any time (Section 5.3).

[13] https://aegisresearch.eu/solutions/advanced-visualization-toolkit/

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 **Error! Reference source not found.**and D3.11 (the current document) d escribe the security specifications and the relevant implemented measures.
TC26	The DUET platform has been designed following a microservices-based architecture.
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 have been employed by DUET to ensure secure development of software components.
TC33	Automation in deployment has been partially setup during the integration process for the Open Beta Release and further automation will be pursued towards the final release.
TC32	A configuration service will allow the controlling of a DUET platform instance.
TC34	Real-time monitoring of system services in conjunction with the capability for analysis of historical malfunction information enhance audit capabilities and enable situational awareness.

5.3 DUET Identity and Access Management

One of the important developments towards securing the DUET platform and processes is a mechanism that provides Identity management and Access Controls based on user roles and security policies. The DUET Identity and Access Management (DUET IAM) handles 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[14] 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.
- Fine-grained authorization policies to combine different access control mechanisms

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

- Password policies
- 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 was selected as the best fit for a platform such as DUET. The User Manager and Data Sources Gateway take advantage of Keycloak' s connectors to provide its functionalities to the rest of the components.

5.4 DUET Privacy Mechanisms

Technical Controls TC18 (Fair, lawful and transparent management of personal data), TC19 (Minimum personal data collected and processed), TC20 (Anonymization/pseudonymization of personal data) and TC21 (Allow data subjects to manage the personal data collected) 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. It must be noted that the majority of the measures mentioned below were also described in the previous version of the deliverable (D3.10), so the description in this version provides the current implementation status together with updates based on the evolvement of the platform components and processes.

Moreover, as presented in Table 5.3 above, several deliverables in other work packages describe measures and processes pertaining to personal data usage and privacy handling during the onboarding of DUET users. A significant detail is that personal data refer to information having to do with testing and providing feedback for the DUET platform, whereas personal data used within DUET (by the components) only use datasets with no personal information, i.e. open data, anonymised datasets, etc. as analytically listed in the continuously updated document with DUET datasets in use which is available here. The effort given to maintain a live version of this document is also important due to the involved assessment of third parties providing the used datasets which helps in determining if any privacy implications are present. It must be also mentioned that all pilot sites of DUET have a designated Data Protection Officer to handle any potential data privacy issues and matters that lie within GDPR and any other legislative restrictions.

As seen in Table 5.2, the Users Manager and Data Sources Gateway together with the DUET IAM component take care of the necessary mechanisms to preserve privacy in data handling by the platform components, especially the ones dealing with model execution and the relevant interactions with DUET Core as analysed in D3.9 [10]. The models used to perform the requested simulations of the city digital twins operate on a 'run and forget' mode where results are sent to the users (via the visualisations) without storing any information on the user or the datasources used to execute the simulation. This approach adds up to the data loss prevention capabilities of DUET, since unauthorised third parties have no way of retrieving simulation results of scenarios which they do not have access to in the first place. Furthermore, DUET landing and platform pages which serve as the entry point to DUET functionalities also foster privacy of registered users via solid user management capabilities (based on underlying frameworks like Wordpress) and minimum personal data storing. Basic measures enforced by the above-described mechanisms and components also include:

- Personal data dissociation, since no personal data is required to register a new data source. A unique system identifier is the only information kept by the system to allow the connection of the data source with the rest of the components of the platform.
- Implementation of the Principle of least privilege (POLP) for the role authorisation scheme which permits operation of DUET users at the lowest privilege level possible.
- Anonymity is maintained through DUET since users don’t need to provide their real name in order to register to the system and are 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 is guaranteed via the implemented User Manager and Data Source Gateway components which provide users the capability to add, edit, change any parameter and also completely delete their data sources, cases and scenarios.
- Apart from connected data sources, user profiles can be also created, managed and deleted by their owners without any intervention by DUET. The same applies to cases and scenarios created by users within the platform. 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 is an update of D3.10 and describes the implementation of the security and privacy mechanisms presented in the previous version. D3.10 has identified security and privacy challenges in a Digital Twin Environment that have been used to form the DUET Threat Taxonomy. In sequence, security requirements and the respective measures to address them were also defined. In this new version of the document, the Technical Controls that verify the application of security and privacy measures have been clearly mapped with DUET components so as to point out how the multi-layered security model has been realised. The current implementation is based on the Open Beta Release of the DUET platform but provides a clear approach to accommodate emerging user needs towards the final version of the platform.

Future work in ensuring privacy and security of the final DUET platform includes the establishment of a regular security and privacy assessment process that will ensure that current measures are consistently implemented by the platform components and will identify new measures that should be taken to address new threats. Additional privacy-preserving actions could include the definition of incident response plan to cater for responses against security incidents or privacy violations (e.g. data breaches). Moreover, identification and inclusion of new data sources to the DUET ecosystem will have to follow a standardised approach so as not to jeopardise the security and privacy layer of DUET. These proposed future steps will be shaped after taking into account feedback received by users and experts during the upcoming pilot testing cycles (Cycle 2 and 3).

7. References

[1] DUET project, Deliverable D3.10: “Multi Layered security model specification”, November 2020

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

[3] 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

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

[5] DUET project, Deliverable D1.2: “Cities Guide to Legal Compliance for Data-Driven Decision Making It. 1”, September 2020

[6] DUET project, Deliverable D1.3: “Cities Guide to Legal Compliance for Data-Driven Decision Making It. 2”, July 2021

[7] DUET project, Deliverable D1.5: “Ethical Principles for using Data-Driven Decision in the Cloud (It 1)”, May 2021

[8] DUET project, Deliverable D8.3: “Data Management and Modeling Plan”, November 2021

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

[10] DUET project, Deliverable D3.9: “Digital Twin data broker specifications and tools v1”, November 2021