Draft ETSI EN 304 619 V1.0.0 (2026-07)

HARMONISED EUROPEAN STANDARD

Cyber Security (CYBER);
CRA;
Cybersecurity requirements for software that searches for, removes, or quarantines malicious software

Reference
DEN/CYBER-EUS-0018

Keywords
antimalware, antivirus, CRA, cybersecurity,
security analysis, security profiles,
security requirements, threat analysis, use cases

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables (European Standard (EN), Technical Specification (TS), Group Specification (GS) or ETSI Standard (ES)) may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI IPR online database.

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™**, **LTE™** and **5G™** logo are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This draft Harmonised European Standard (EN) has been produced by ETSI Technical Committee Cyber Security (CYBER), and is now submitted for the combined Public Enquiry and Vote phase of the ETSI Standardisation Request deliverable Approval Procedure (SRdAP).

The present document has been prepared under the Commission's standardisation request C(2025) 618 final [i.3] to provide one voluntary means of conforming to the requirements of Regulation (EU) 2024/2847 of the European Parliament and of the Council of 23 October 2024 on horizontal cybersecurity requirements for products with digital elements and amending Regulations (EU) 168/2013 and (EU) 2019/1020 and Directive (EU) 2020/1828 (Cyber Resilience Act) (CRA) [i.1].

Once the present document is cited in the Official Journal of the European Union under that Regulation, compliance with the normative clauses of the present document given in Table A.1 confers, within the limits of the scope of the present document, a presumption of conformity with the corresponding requirements of that Regulation and associated EFTA regulations.

Proposed national transposition dates	
Date of latest announcement of this EN (doa):	3 months after ETSI publication
Date of latest publication of new National Standard or endorsement of this EN (dop/e):	6 months after doa
Date of withdrawal of any conflicting National Standard (dow):	18 months after doa

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Introduction

The present document defines cybersecurity requirements for products with digital elements (product) whose intended purpose is to search for, remove, or quarantine malicious software or code. These products are intended to protect a protected system against malicious software or code that can compromise its confidentiality, integrity, or availability.

The present document is part of a broader group of vertical standards developed to support the Cyber Resilience Act (CRA) [i.1] and should not be considered a standalone resource. Users are expected to consult it in conjunction with the CRA itself [i.1], the associated legal guidance [i.13], as well as relevant horizontal and neighbouring vertical standards cited in clause 2, such as prEN 40000-1-1 [i.4], prEN 40000-1-2 [i.5], prEN 40000-1-3 [i.6], and prEN 40000-1-4 [i.7]. Material already addressed in those sources is not repeated in the present document.

The present document provides the technical cybersecurity requirements for the products in scope, following a risk-based approach in support of the Cyber Resilience Act (CRA) [i.1]. The technical cybersecurity requirements are thereby proportionate to the intended purpose, reasonably foreseeable use, deployment context, and threat exposure of the products.

Clause 4 does not contain technical requirements; it describes the product context that is considered for the application of the present document.

Clause 4 also defines Use Cases (UCs) that represent the main deployment scenarios reflecting the intended purpose and reasonably foreseeable use of the product, which serve as the basis for identifying relevant cybersecurity risks.

Clause 5 specifies technical cybersecurity requirements for the product to mitigate the identified risks, including their applicability conditions.

Clause 6 specifies the assessment criteria and compliance verification procedures with the requirements of clause 5.

Annex A maps the technical requirements of the present document with the essential requirements of the CRA [i.1] regulation.

Annex B informs about the methodology used to assess the security risks of the products in their context.

Annex C informs the possible relationship between the product in the present document and other possible vertical products and their corresponding CRA-related harmonised standards.

Annex G presents further information on guidance for the application of the present document.

Annex K supports the definition of the cryptographic requirements and assessment criteria used by the present document.

Annex R provides supplementary requirements and assessment provisions where a product relies on Remote Data Processing Solutions (RDPS) for the provision or support of one or more product functions.

1 Scope

The present document specifies technical requirements and corresponding assessment criteria for software products with digital elements that detect or search for malicious software or code on a device, or remove or quarantine such software or code to prevent or mitigate system infection related to cybersecurity. The products with digital elements in scope, thereafter "the product":

- are specified within the "technical description" of the "category of product" number "4" by the Commission Implementing Regulation (EU) 2025/2392 [i.2] as:
  - "Software products with digital elements, typically referred to as antivirus or antimalware, that detect or search for malicious software or code on devices, or remove or quarantine such software or code, in order to maintain the integrity, confidentiality, or availability of such devices.

    In the context of this category of products, malicious software means software containing malicious features or capabilities that can cause harm directly or indirectly to the user and/or the computer system, such as viruses, worms, ransomware, spyware and trojans.

    This category includes but is not limited to software that detects or searches for malicious software in real-time or manually, rootkit detection and rescue disks with the core functionality of searching, removing or quarantining malicious software."
- are only covered within the product context described in clause 4.

The present document covers those products to demonstrate compliance with essential cybersecurity requirements in the Regulation (EU) 2024/2847 [i.1], Annex I, Part I under the conditions identified in Annex A.

The present document specifies technical characteristics and methods of assessment for Antivirus/Antimalware products.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found in the ETSI docbox.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents are necessary for the application of the present document.

[1] ENISA Report 1747792503 (version 2 - April 2025): "European Cybersecurity Certification Group Sub-group on Cryptography Agreed Cryptographic Mechanisms".

[2] IETF-IRTF RFC 7748 (Version 11 - January 2016): "Elliptic Curves for Security".

[3] IETF-IRTF RFC 8032 (Version 08 - January 2017): "Edwards-Curve Digital Signature Algorithm (EdDSA)".

[4] NIST FIPS 186-5 (Version 5 - February 2023): "Digital Signature Standard (DSS)".

[5] IETF-IRTF RFC 9106 (Version 14 - September 2021): "Argon2 Memory-Hard Function for Password Hashing and Proof-of-Work Applications".

[6] IETF RFC 7914 (Version 5 - August 2016): "The scrypt Password-Based Key Derivation Function".

[7] IETF-IRTF RFC 8439 (Version 4 - June 2018): "ChaCha20 and Poly1305 for IETF Protocols".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents may be useful in implementing an ETSI deliverable or add to the reader's understanding, but are not required for conformance to the present document.

[i.1] Regulation (EU) 2024/2847 of the European Parliament and of the Council of 23 October 2024 on horizontal cybersecurity requirements for products with digital elements and amending Regulations (EU) No 168/2013 and (EU) 2019/1020 and Directive (EU) 2020/1828 (Cyber Resilience Act).

[i.2] Commission Implementing Regulation (EU) 2025/2392 of 28 November 2025 on the technical description of the categories of important and critical products with digital elements pursuant to Regulation (EU) 2024/2847 of the European Parliament and of the Council.

[i.3] Standardisation request M/606 - C(2025)618: "Commission Implementing decision of 3.2.2025 on a standardisation request to the European Committee for Standardisation (CEN), the European Committee for Electrotechnical Standardisation (Cenelec) and the European Telecommunications Standards Institute (ETSI) as regards products with digital elements in support of Regulation (EU) 2024/2847 of the European Parliament and of the Council of 23 October 2024 on horizontal cybersecurity requirements for products with digital elements and amending Regulations (EU) No 168/2013 and (EU) 2019/1020and Directive (EU) 2020/1828 (Cyber Resilience Act)".

[i.4] prEN 40000-1-1: "Cybersecurity requirements for products with digital elements - Vocabulary" (produced by CEN CENELEC).

NOTE: Version and date to be added upon its publication by CEN CENELEC.

[i.5] prEN 40000-1-2: "Cybersecurity requirements for products with digital elements - Part 1-2: Principles, product risk management, and lifecycle activities" (produced by CEN CENELEC)".

NOTE: Version and date to be added upon its publication by CEN CENELEC.

[i.6] prEN 40000-1-3: "Cybersecurity requirements for products with digital elements - Part 1-3: Vulnerability handling" (produced by CEN CENELEC).

NOTE: Version and date to be added upon its publication by CEN CENELEC.

[i.7] prEN 40000-1-4: "Cybersecurity requirements for products with digital elements - Part 1-4: Security controls - Generic security requirements" (produced by CEN CENELEC).

NOTE: Version and date to be added upon its publication by CEN CENELEC.

[i.8] ETSI EN 304 626: "CYBER; CRA; Cybersecurity requirements for operating systems".

NOTE: Version and date to be added upon its publication by ETSI.

[i.9] ETSI EN 304 636: "Cyber Security (CYBER); CRA; Cybersecurity requirements for firewalls, intrusion detection and/or prevention systems".

NOTE: Version and date to be added upon its publication by ETSI.

[i.10] ETSI EN 304 622: "Cyber Security (CYBER); CRA; Cybersecurity requirements for Security information and event management (SIEM) systems".

NOTE: Version and date to be added upon its publication by ETSI.

[i.11] ETSI EN 304 623: "Cyber Security (CYBER); CRA; Cybersecurity requirements for boot managers".

NOTE: Version and date to be added upon its publication by ETSI.

[i.12] Directive (EU) 2022/2555 of the European Parliament and of the Council of 14 December 2022 on measures for a high common level of cybersecurity across the Union, amending Regulation (EU) No 910/2014 and Directive (EU) 2018/1972, and repealing Directive (EU) 2016/1148 (NIS 2 Directive). .

[i.13] EC: "Commission CRA draft legal guidance".

[i.14] ISO/IEC 27005: "Information cybersecurity, cybersecurity and privacy protection — Guidance on managing information cybersecurity risks".

[i.15] BSI TR-02102-1 (Version: 2026-01 - January 2026): "Cryptographic Mechanisms: Recommendations and Key Lengths".

[i.16] IETF-IRTF draft-irtf-cfrg-aegis-aead-18 (Version 18 - October 2025): "The AEGIS Family of Authenticated Encryption Algorithms".

[i.17] Conference Paper (Version 1.1 - August 2013): "AEGIS: A Fast Authenticate Encryption Algorithm".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in Regulation (EU) 2024/2847 [i.1], prEN 40000-1-1 [i.4] and the following apply:

antivirus/antimalware: software product whose essential function(s) is searching, detecting, removing, or quarantining malicious software or code

cryptographic item: cryptographic algorithm, primitive, scheme, protocol, protocol profile, cipher suite, mode of operation, construction or parameter set used to provide or support a cryptographic property

NOTE: A cryptographic item can be specified at different levels of abstraction. For example, AES is a cryptographic primitive, AES-GCM is a mode of operation / authenticated encryption construction, TLS 1.3 is a protocol, a TLS 1.3 restricted cipher-suite profile is a protocol profile, and TLS_AES_128_GCM_SHA256 is a cipher suite.

cryptographic mechanism: security-related procedure using cryptography

NOTE 1: The term "cryptographic mechanism" is used in Annex K as an umbrella term covering the categories used in the ECCG Agreed Cryptographic Mechanisms (ACM) catalogue [1], including cryptographic algorithms, primitives, schemes, protocols, protocol profiles, cipher suites, modes of operation, constructions and parameter sets.

NOTE 2: The terms "cryptographic primitive", "cryptographic construction", "cryptographic scheme" and "cryptographic protocol" are used consistently with the ACM catalogue [1].

NOTE 3: A cryptographic mechanism can be specified at different levels of abstraction. For example, AES is a cryptographic primitive, AES-GCM is a mode of operation / authenticated encryption construction, TLS 1.3 is a protocol, a TLS 1.3 restricted cipher-suite profile is a protocol profile, and TLS_AES_128_GCM_SHA256 is a cipher suite.

cryptographic property: property provided or supported by a cryptographic mechanism

NOTE 1: Cryptographic properties include, for example, confidentiality, integrity, authenticity, entity authentication, message authentication, key establishment, key confirmation, replay protection, collision resistance, second-preimage resistance, pre-image resistance, signature unforgeability, non-repudiation, forward secrecy and password-verifier protection. Where relevant, a cryptographic property can be expressed using a formal cryptographic notion, for example IND-CPA, IND-CCA, IND-CCA2, EUF-CMA, SUF-CMA or authenticated key exchange security.

NOTE 2: A cryptographic property is distinct from a product-level technical requirement, although it can support such a requirement. For example, a secure communication requirement can rely on cryptographic properties such as confidentiality, integrity and authentication.

detection rules update: malware signatures, heuristic rules, detection database

heuristic analysis: detection method based on rule- or behaviour-based analysis used to identify potentially malicious software or code not matching known signatures

interoperability: need to support interoperability with identified systems, deployments or operational environments that require use of specific cryptographic items which may not comply with the state-of-the-art in the domain

known exploitable vulnerabilities: Exploitable vulnerability as defined in [i.1] Article 3 point (41) that it is known.

malware software (malware): software designed with malicious intent that can disrupt, damage, or gain unauthorised access

multi-factor authentication: cybersecurity process requiring users to provide two or more verification factors to gain access to an account or application

personal data: As defined in [i.1] Article 3 point (47).

principle of least privilege: every user, process, service, and component gets only the minimum access (permissions, data, time, and network reach) needed to do its job with the goal to shrink attack surface and limit blast radius if something is compromised

NOTE: This definition is given in prEN 40000-1-1 [i.4].

protected system: PwDE, system or environment on which the antivirus/antimalware

quarantine: isolation of suspected malicious software or code to prevent execution or interaction with the protected system

security update: software patch addressing a product vulnerability

signature-based detection: detection method based on comparison with known patterns or signatures of malicious software or code

update mechanism: process used to retrieve, verify, and apply updates to signatures, detection logic, or related cybersecurity components

virus: type of malware specifically designed to self-replicate by injecting its own code into other programs code files

3.2 Symbols

Void.

3.3 Abbreviations

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

3GPP	3rd Generation Partnership Project
AAC	Authentication and Access Control
ACM	Agreed Cryptographic Mechanisms

ACN Agenzia per la Cybersicurezza Nazionale

NOTE: In English: National Cybersecurity Agency of Italy.

AES Advanced Encryption Standard
AES-GCM Advanced Encryption Standard - Galois/Counter Mode
AES-XTS Advanced Encryption Standard in XEX-based Tweaked Codebook mode with Ciphertext Stealing
API Application Programming Interface
ASLR Address Space Layout Randomization
ASM Attack Surface Minimisation
BSI Bundesamt für Sicherheit in der Informationstechnik

NOTE: In English: Federal Office for Information Security.

CAVP Cryptographic Algorithm Validation Program
CEN Comité Européen de Normalisation

NOTE: In English: European Committee for Standardization.

CENELEC Comité Européen de Normalisation Électrotechnique

NOTE: In English: European Committee for Electrotechnical Standardization.

CERT Computer Emergency Response Team
CFI Control Flow Integrity
CI/CD Continuous Integration and Continuous Delivery/Deployment
CIA Confidentiality, Integrity, Availability
CISA Cybersecurity and Infrastructure Security Agency
CLI Command Line Interface
CMVP Cryptographic Module Validation Program
CON CONfidential
CPU Central Processing Unit
CRA Cyber Resilience Act
CSA CyberSecurity Act
CSRC Computer Security Resource Center
CVE Common Vulnerabilities and Exposures
DA Data Asset
DB Database
DEP/NX Data Execution Prevention/No-Execute
DNS Domain Name Server
DoS Denial of Service
DPIA Data Privacy Impact Assessments
EDR Endpoint Detection and Response
ENISA European Union Agency for Cybersecurity

NOTE: Originally established as European Network and Information Security Agency.

EPC Energy Performance Certificate
ETSI European Telecommunications Standards Institute
EUF-CMA Existential Unforgeability under Adaptive Chosen Message Attack
FRD Factory Reset and Data portability
HKDF HMAC-based Key Derivation Function
HMAC Hash-based Message Authentication Code
HTTP Hypertext Transfer Protocol
HTTPS Hypertext Transfer Protocol Secure
I/O Input/Output
IAM Identity and Access Management
ID IDentifier
IDS Intrusion Detection System
IEC International Electrotechnical Commission
IEEE Institute of Electrical and Electronics Engineers
IETF Internet Engineering Task Force
IND-CCA Indistinguishability under adaptive Chosen-Ciphertext Attack
IND-CPA Indistinguishability under Chosen-Plaintext Attack

INT	INTegrity
IP	Internet Protocol
IPC	Inter-Process Communication
IPS	Intrusion Prevention System
ISO	International Organization for Standardization
IT	Information Technology
ITU-T	International Telecommunication Union - Telecommunication Standardization Sector
JTC	Joint Technical Commitee
KEV	Known Exploitble Vulneravility
LAN	Local Area Network
MAC	Media Access Control
MFA	Multi-Factor Authentication
MITM	Machine-In-The-Middle
MON	MONitoring
MTTD	Mean Time To Detect
MTTR	Mean Time To Respond
NIC	Network Interface Card
NIS2	Network and Information Security Directive 2
NIST	National Institute of Standards and Technology
NVD	National Vulnerability Database
OoA	Object of Assessment
O-RAN	Open Radio Access Network
OS	Operating System
PoLP	Principle of Least Privilege
PwDE	Product with Digital Elements
RBAC	Role-Based Access Control
RDPS	Remote Data Processing Solution
RF	Risk Factor
RFL	Risk Factor on Likelihood
RFM	Risk Factor on Magnitud
RTO	Recovery Time Objective
SaaS	Security as a Services
SBD	Secure By DefaultSBOM Software Bill Of Materials
SDK	Software Development Kit
SIEM	Security Information and Event Management
SLA	Service Level Agreement
SOAR	Security Orchestration Automation and Response
SOG-IS	Senior Officials Group - Information Systems Security
SP	Security Profile
SPDX	Software Package Data eXchange
SU	Security Updates
SUF-CMA	Strong Existential Unforgeability under Chosen-Message Attack
TC	Technical Committee
TCP	Transmission Control Protocol
THx.y	THreat, where x and y represent numbers
TLS	Transport Layer Security
TPM	Trusted Platform Module
UC	Use Case
UEFI	Unified Extensible Firmware Interface
UI	User Interface
UPS	Uninterruptible Power Supply
URL	Uniform Resource Locator
USB	Universal Serial Bus
VM	Virtual Machine
VPN	Virtual Private Network
XDR	eXtended Detection and Response
ZT	Zero Trust

4 Product context

4.0 Introduction

The cybersecurity requirements of the present document apply under the product context described herein, which is in accordance with its intended use. The equipment needs to comply with all the cybersecurity requirements of the present document at all times when operating in such product context.

4.1 Product Functions

For the purpose of the security analysis and associated cybersecurity requirements, this clause defines the functional context of the product in relation to its own cybersecurity and its interaction with other systems. The product implements the following high-level functional purposes:

- **Threat Identification Functions:** The product analyses files, processes, system activities, or data flows to determine whether they exhibit characteristics associated with malware. These functions rely on internal detection logic (e.g. signature-based detection or heuristic analysis) and may leverage local or remote data sources. The present document considers these functions only to the extent that they influence the product's attack surface, input validation requirements, privileged operations, or dependency on external data feeds.

- **Threat Handling Functions:** Upon identifying a potential threat, the product can initiate actions that alter system state or data, such as isolating, restricting, or deleting content. For the purpose of the present document, these functions are relevant insofar as they require elevated privileges, interact with critical system components, or modify user or system data, thereby affecting the confidentiality, integrity, or availability of the product or the host environment.

- **System Monitoring and Telemetry Functions:** The product may continuously observe system behaviour, file operations, or network activity to enable its detection capabilities. The present document assesses these functions with regard to their impact on data protection, authorised access boundaries, and the exposure of monitored data to internal or external interfaces.

- **Update and Maintenance Functions:** The product relies on periodic updates to its detection logic, data sources, or operational modules. Updates may be delivered locally or remotely. These functions are considered in terms of secure update mechanisms, protection of update channels, validation and authenticity of update content, and assurance that update processes do not introduce vulnerabilities or degrade the cybersecurity of the product.

- **User and System Interaction Functions:** The product may provide notifications, logs, or management interfaces that enable users or administrators to review identified threats or modify product behaviour. The present document covers these interactions as potential exposure points where authentication, authorisation, integrity protection, and availability should be ensured.

These functional elements define the operational boundaries of the product for the purposes of assessing cybersecurity risks and establishing cybersecurity requirements under the CRA [i.1]. The functions are described at a level necessary to analyse the product's role in a system, its privileged capabilities, and any interfaces or dependencies that could introduce cybersecurity risks.

4.2 Product Architecture

The present clause describes a generic product architecture used to identify the cybersecurity-relevant assets considered within the present document. The architecture serves solely as an analytical model to support the security analysis and associated cybersecurity requirements. It does not prescribe implementation methods, does not limit possible design approaches, and does not imply that all products shall implement all elements described herein.

The architecture reflects typical structural components found in antivirus/antimalware products as defined in clause 4.1 and performing the functional purposes described in clause 4.2. It enables manufacturers to determine whether the architectural assumptions underlying the present document correspond to their product design. Alternative architectures may exist and may be included in future revisions of the present document.

The generic architecture includes the following assets:

- **User Interface:** Provides the means through which users or administrators interact with the product, including access to configuration options, review of system status, and visibility of product actions. The UI constitutes an interaction surface and may expose cybersecurity-relevant data or controls. In some cases the UI can be remote and the product can be controlled remotely.
- **Detection Data Sources:** Includes local or remote sources of data used to support threat identification (e.g. signature data, reputation data, behavioural rules). These sources may influence the integrity and confidentiality of detection operations and may involve dependencies on external systems.
- **Behavioural or Heuristic Analysis Components:** Implements analysis mechanisms used to assess files, processes, or system activities against detection logic. These components typically operate on untrusted inputs and may execute within privileged or sensitive contexts, making them relevant to the product's attack surface.
- **Dynamic Analysis Environment:** Represents isolated or controlled execution environments used to analyse potentially harmful content. While such environments interact closely with the host system and may require strong isolation properties to prevent adverse effects in case of malicious behaviour during analysis, they may also be part of a remote component. For this reason, it is key that the product identifies where this component is within its architecture to properly identify the corresponding threats and risks.
- **Detection Decision Logic:** Aggregates inputs from different analysis components and determines the classification or handling of potentially harmful content. Decision logic may directly initiate actions affecting system state and therefore constitutes a critical cybersecurity-relevant asset.
- **Threat Handling Components:** Implements mechanisms for isolating, restricting, or removing content identified as potentially harmful. Such components may require elevated privileges and may manipulate system files or processes, making them essential to the cybersecurity model of the product and the host environment.
- **Update and Maintenance Mechanisms:** Provides the means through which the product obtains updated data, configuration, or software components. These mechanisms are cybersecurity-critical due to their role in maintaining product integrity, and due to potential exposure to remote sources.
- **Logging & Telemetry Components:** Record operational events, product actions, and error conditions. In some implementations, telemetry may be shared with backend services to support threat intelligence or operational improvement. These components may handle sensitive data and expose interfaces relevant to access control and data protection.
- **Backend or Remote Services:** Represents remote infrastructure that may support enhanced detection, data retrieval, update distribution, administrative, UI, RDPS capabilities or other operational functions. Backend services may include cloud-based environments or other remote systems, and may introduce communication-related or dependency-related risks. Depending on the setup, they may expose cybersecurity-relevant data or controls or need isolation properties, if corresponding functionality is implemented remotely. Within the scope of the present document the definition is restricted to the components directly needed for the product's functionality, thus explicitly excluding a general analysis and signature creation framework needed as supporting infrastructure.

The elements described above form the basis for the cybersecurity considerations and requirements defined in the present document. They represent the architectural scope considered during the security analysis and do not constitute an exhaustive or prescriptive model for all antivirus/antimalware products.

4.3 Operational Environment

4.3.1 General description

The present clause characterises the operational environment in which the product is deployed and operated, to frame the security analysis and the applicability of the cybersecurity requirements defined in clause 5. The content in this clause is descriptive and does not prescribe implementation choices.

4.3.2 Physical/Hardware environment

The present clause describes the physical operational environments considered for the product in the present document. The objective is to identify physical deployment contexts and access conditions that influence the product's CIA, inform the security analysis, and frame the applicability of the cybersecurity requirements in clause 5. This clause is descriptive and does not prescribe implementation choices.

- **Deployment Form Factors and Sites:** Multiple physical deployment contexts are considered, such as user endpoint (i.e. laptops, desktops, workstations, and thin/virtual desktop clients operated by end users), enterprise servers and appliances (i.e. general purpose servers or dedicated appliances hosted in controlled facilities such as equipment rooms or data centres), field and branch installations (i.e. devices operated in semi-controlled or remote sites such as branch offices, shared spaces, customer premises), and embedded or specialised devices deployed in constrained, unattended or physically exposed environments. Physical location affects exposure to theft, tampering, and uncontrolled shutdowns, with potential impact on product availability (e.g. interruption of protection services), integrity of stored data (e.g. quarantine, logs, configurations), and confidentiality of sensitive artefacts (e.g. keys, telemetry in buffers). Requirements related to secure storage, tamper resistance/evidence (where applicable), and safe recovery are addressed in clause 5.

- **Physical Access Models:** The following access conditions such as individually assigned devices (i.e. single user, managed), shared or kiosk devices (i.e. multiple users, reduced assurance of physical control). operator-restricted devices (i.e. locked enclosures, supervised access). untrusted physical environments (i.e. devices temporarily or persistently exposed to hostile actors) need to be considered. Requirements for local interface protection, hardening of privileged components, and audit/logging resilience are addressed in clause 5.

- **Removable Media and Peripheral Interfaces:** Deployments may involve removable or externally attached media and devices (e.g. USB storage, external drives, removable NICs) and service ports exposed on physically accessible hosts.

- **Risk linkage:** Physical introduction of untrusted content, interface abuse to disable or confuse protection mechanisms, and exfiltration opportunities via removable media. Requirements concerning scanning triggers, interface protection, rate limiting, and policy enforcement around removable media are addressed in clause 5.

- **Hardware Trust Anchors and Secure Storage:** Deployments may rely on platform trust anchors (e.g. TPM on endpoints, secure elements on appliances) and on hardware-assisted protections for keys, measurements and sealed storage. Physical access conditions influence the tamper resistance expected from these components and the assurances they provide. Requirements for binding of sensitive state to hardware trust anchors (where available), protection of local keys and credentials, verification of integrity measurements, and secure handling of device reprovisioning or decommissioning are addressed in clause 5.

- **Environmental Conditions Relevant to Cybersecurity:** Environmental factors are considered only insofar as they affect cybersecurity (i.e. CIA). Examples include conditions that materially increase error rates or instability (e.g. repeated thermal-induced reboots) with potential loss of protection or corrupted state, and storage/media reliability considerations affecting the durability of logs, quarantine and update artefacts. Requirements on integrity verification, redundancy or journaling for critical artefacts, and detection/reporting of degraded protection states are addressed in clause 5.

- **Power, Shutdown and Availability Conditions:** Operation may be subject to unexpected power loss, battery depletion, frequent suspend/resume cycles, or controlled shutdown policies (e.g. in data centres with UPS). The relevance to cybersecurity lies in how such events affect transactional integrity (e.g. updates, policy changes), logging completeness, and service continuity. Requirements for atomic/verified update, crash-safe storage of critical state, graceful degradation (e.g. read-only policy on boot if integrity is uncertain), and recovery procedures are addressed in clause 5.

4.3.3 Logical/Software environment

The present clause describes the logical operational environment considered for antivirus/antimalware products within the scope of the present document. The aim is to identify the principal external dependencies and interactions that influence the product's cybersecurity posture, inform the security analysis, and frame the applicability of the cybersecurity requirements specified in clause 5. This clause is descriptive and does not prescribe implementation choices. Where a product introduces additional external dependencies or fundamentally different interaction patterns, the associated risks may fall outside the scope of the present document until explicitly analysed in future revisions.

- **Platform and Execution Environment:** Antivirus/antimalware products operate on top of an operating system (OS) that provides process execution, memory management, file system services, inter-process communication, cryptographic services and networking stacks. Informative material on OS cybersecurity requirements can be found in [i.8]. The interaction of the product with the OS is relevant to the security analysis due to:
  - reliance on OS interfaces and privilege models (e.g. service accounts, kernel drivers, user-mode services);
  - exposure to untrusted inputs via OS facilities (e.g. file I/O, IPC, network APIs);
  - dependency on OS cybersecurity features (e.g. isolation models, access control, code-signing enforcement).
- **Boot and Trust Chain:** Certain product components may integrate with the boot sequence (e.g. early-start services, kernel-mode drivers) and rely on the platform trust chain (e.g. UEFI Secure Boot, measured boot, TPM-based attestation) to ensure integrity of the execution environment. Information material on boots and cybersecurity requirements can be found in [i.11].
- **Virtualisation and Containment:** Products may execute on virtualised hosts or interact with sandboxed/contained workloads (e.g. VMs, containers). In such cases, cybersecurity-relevant factors include visibility into guest resources, isolation boundaries between guest and host, and broker components used for inspection.
- **Backend and Remote Services:** Products may interact with backend or remote services to obtain updates, reputation or classification data, advanced analysis, license validation, management orchestration, or telemetry aggregation. These services may be cloud-based or on-premises.
- **Interoperation with other Cybersecurity Components:** The following interactions are considered for the purposes of the security analysis. They are framed as architectural contexts rather than functional overlap:
  - **Firewall / IDS / IPS**: Network-centric controls that complement host-centric malware protection. Coordination points include policy consistency, exception handling and event correlation. Informative references: [i.9].
  - **SIEM / SOAR**: Consumption of product telemetry and alerts to support detection engineering and automated response workflows. Coordination points include event schemas, authenticity/integrity of logs, and avoidance of conflicting response actions. Informative references: [i.10].
  - **EDR / XDR**: Endpoint and cross-domain detection/response platforms that may use product classifications as signals and orchestrate responses. Coordination points include data exchange, action de-duplication and policy precedence.
- **Identity, Trust and Update Infrastructure:** Operation may rely on public-key infrastructures (code-signing certificates, trust anchors), time synchronisation, revocation information and licensing/entitlement systems.
- **Administrative and Management Environment:** Products may be locally administered or managed centrally (e.g. enterprise consoles, directory-integrated policies, etc.). Management channels and role-based access controls form part of the logical environment.

4.3.4 Connectivity aspects

The product in the present document is not affected or does not manage connectivity aspects as its main functionalities are the identification, removal and/or isolation of malware software that may affect the system hosting the product. Any connectivity aspect that the product in the present document may have, relies on the host system (e.g. OS) where the product is hosted.

4.4 Distribution of Security Functions

4.4.1 Cybersecurity Functionalities Offered to Other Products

Possible cybersecurity functionalities that the product in the present document may offer to other products are comprehensive security services to applications and user data. For example:

- Detection and prevention to:

  a) scan files (e.g. configurations, logs, events, executables, etc.) using signature or behavioural models (or other possible mechanisms) and identify if they are/contain malicious software;

  b) to validate if active programs aim to apply suspicious actions;

  c) to protect user applications and data from ransomware; and/or

  d) to detect possible known vulnerabilities being exploited and propose their corresponding patches (if available).

  Depending on the product, these actions may be done in real-time or not.

- Isolation to provide safe environments to run suspicious or unknown applications, preventing them from accessing sensitive system files or other user apps.

- Antivirus security extensions in browsers (to block malicious sites/ads), password managers, and VPNs to protect user credentials and data.

- Antivirus may share threat intelligence databases and identified cybersecurity events (e.g. alerts, incidents, compliance and audit logs, etc.) with other products (e.g. Firewalls, IDS/IPS, SIEM, etc.) to enforce the security in a system or domain.

4.4.2 Cybersecurity Functionalities Required from Other Products

To properly do its essential functionalities, the product may require deep system integration and support from the host and the environment it protects:

- Antivirus solutions need elevated/privileged system access to access key parts of the host/environments (e.g. low-level system drivers, kernel-level operations, etc.) and read/write access to all files to properly offer the detection and prevention functionality previously described.

- Antivirus have a strong dependency on OSs as they need: a) to offer APIs and hooks to inspect scripts and behaviour in real-time without causing the OS or the environment to crash, b) to allow automatic, timely security updates to patch vulnerabilities and receive new threat definitions, c) to deliver secure methods for scanning data at rest, including files stored in encrypted formats or containers and/or d) to give the necessary memory and CPU.

- Antivirus may also require trusted execution and integrity mechanisms to: a) interact with hardware with hardware-based security elements to ensure the integrity of the boot process and/or b) code signing and signature verification.

- Like any software-based product, antivirus require frequent updates (e.g. software related or security related), to this end they require, in addition to the possible security updates requirements in the present document, that other elements such as the OS or the network products (e.g. routers, switches, etc.) ensure a secure transmission of any data sent or received by the antivirus.

4.5 Users

The product can be operated by different types of users across various governance and administrative contexts. These can range from individual device owners to centrally managed administrative teams operating within structured organizational environments.

The way in which the product is administered and managed can influence configuration practices, operational discipline and the likelihood of misconfiguration or misuse. However, the applicable UCs and corresponding SPs are determined by the deployment context and associated risk characteristics, rather than by the skill level or organizational type of the user.

Accordingly, while cybersecurity expectations and requirements defined in the present document focus on the robustness of the product itself and do not assume a particular level of user expertise, when necessary, a specific user will be referenced for clarity and avoid vague interpretations.

To this end, the following users in the context of the product in the present document were identified:

- Security administrative user: A user of the product who handles security-relevant settings and manages threats. This user has an extensive knowledge and cybersecurity usage rights over the product deployed in the protected system and it is the one to respond to the cybersecurity aspects.
- IT administrative user: A user of the product who deploys and maintains the product on a protected system or removes it from a protected system.
- Privileged user: A user of the product who has elevated informative access to the product, but does not have administrative access, and thus cannot access security-relevant settings, manage threats or deploy / remove the product.
- End user: A user of the product who works on a protected system. May not exist on all kinds of protected system.

NOTE: The term user, if used in the present document outside this clause and without further specification, then, the different users defined within this clause are not necessarily separate, depending on the product and environment.

4.6 Use Cases

4.6.1 Introduction

This clause identifies UCs, where the intended product is the software that searches for, removes, or quarantines malicious software or code. For each UC, a set of minimum capabilities that the product vendor can implement are presented, together with the associated risk that can appear. In addition, a set of possible and/or potential supply chain interactions with other standardised products is proposed as an informative aspect for the product manufacturers to take into consideration when possible, at the time of their product development and possible requirements needed to interact with these potential products.

This clause defines representative UC (i.e. deployment contexts) in which the product may be used. Each UC describes the operational environment, integration characteristics and working domain.

The UCs are intended to capture differences in exposure, privilege levels, dependency relationships and the operational context of the antimalware functions provided by the product.

The UCs do not prescribe product capabilities or performance characteristics. Instead, they provide the contextual basis for assessing the level of cybersecurity robustness expected of the product, as reflected in the SP mapping defined in Table 1.

A single organization may operate multiple deployment contexts simultaneously. The applicable UC is therefore determined per deployment environment, in relation to the intended purpose of the product and its reasonably foreseeable use.

4.6.2 UC1: Limited Operational Context

This use case describes deployment scenarios where antimalware capabilities are delivered as an addon, component or standalone extension to an application whose primary function is different than the antimalware protection.

The protection provided by this component is narrowly scoped to the target application's specific operational context, such as scanning files exchanged within a chat application, inspecting specific data streams processed by a VPN client, or verifying content retrieved by a web browser.

The general operational environment is presumed to be trusted, with this component offering an additional, targeted layer of defence against threats introduced via the application's specific interactions. Configuration and updates for the anti-malware functions are typically managed through the host or standalone application interface.

4.6.3 UC2: Baseline Operational Context

Baseline Operational Context refers to deployment scenarios in which the product is deployed on individual endpoints with public internet connectivity or lightly protected networks, moderate physical cybersecurity (e.g. home, small office, mobile environments), and limited integration with centralised management or cybersecurity infrastructure. Administrative control is primarily local (i.e. endpoint administrator is the product administrator), with configuration and updates managed at the device level.

In this context, the product operates primarily at the endpoint level, with restricted interconnection to other cybersecurity systems or central orchestration platforms.

4.6.4 UC3: Elevated Operational Context

Elevated Operational Context refers to deployment scenarios in which the product is typically deployed within corporate networks featuring standard cybersecurity controls [i.7] (e.g. firewalls, access controls, network segmentation), physical access controls in office or business environments, and centralised administrative structures. Configuration and update processes are coordinated across multiple devices or system instances through management infrastructure.

The product in this UC can be integrated with the external dependencies, including monitoring or management systems.

The deployment environment for this UC includes both individual endpoints and administrative systems.

4.6.5 UC4: High Value Operational Context

High Value Operational Context refers to deployment scenarios in which the product is typically deployed within environments characterised by hardened network infrastructure with perimeter controls and segmentation, physical access controls in data centres or secure facilities, and extensive integration with other cybersecurity or operational systems.

Administrative control structures are formalised, and the product operates within coordinated cybersecurity or monitoring ecosystems.

The features of the deployed product are expected to have the same level of security performance than its environment.

4.6.6 UC5: Critical Operational Context

Critical Operational Context refers to deployment scenarios in which the product is deployed within environments characterised by highly controlled and monitored network infrastructures (potentially air-gapped or isolated), stringent physical cybersecurity measures, and tightly governed operational processes. The product may operate as part of safety-critical systems, essential service infrastructures, dissidents or people targeted by state actors, operations subject to national cybersecurity or critical infrastructure regulations (e.g. NIS2 Directive [i.12]).

The systems protected by the product can be directly bound to public safety insurance, essential services availability (e.g. energy, water, healthcare, transportation), regulated critical infrastructure operations (including national cybersecurity operations), or processes where failure could result in physical harm or widespread social disruption.

4.6.7 Use cases to risk factors and risk levels mapping

Table 1 presents the mapping of the UCs with the SPs (clause B.2) based on the security analysis done in clause B.4 using the risk analysis methodology (clause B.3) and the threats and the RFs in clause B.1.

Table 1: Mapping of the use cases to the risk levels and the SP

UC	Users Involved (clause 4.5)	Computed Total SP (clause B.4)	SP (clause B.2)
UC1	End and Privileged Users	1,0	Minimal
UC2	End and Privileged Users	1,78 → 2	Medium
UC3	Privileged User and Security/IT Administrative Users	2,17 → 3	High
UC4	Privileged User and Security/IT Administrative Users	2,78 → 3	High
UC5	Privileged User and Security/IT Administrative Users	3,0	High

5 Technical requirements for products

5.1 Introduction - Applicability of the requirements

The technical requirements of the present document apply under the product context described in clause 4, which shall be in accordance with its intended use. The product shall comply with all applicable technical requirements of the present document at all times when operating in such a product context.

Not all requirements are universally applicable: The applicability of requirements may be based on use cases or specific capabilities of the product. To this end, at the end of each requirement family clause, there is a table where each of the identified cybersecurity requirements in this clause is mapped to the previous UCs (clause 4.6) and the SPs (clause 4.5) to determine the corresponding applicability.

5.2 Appropriate level of cybersecurity

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (1).

Based on the Commission CRA draft legal guidance [i.13] point 7.2, the present document has been developed with the objective that by using the requirements in the following sub-clauses, the corresponding requirement defined in the CRA [i.1], Annex I, Part I (1) is properly covered given the security analysis methodology described in clause B.3 and implemented in clause B.4 to the UCs (clause 4.6), which it is based on the threats and RFs identified in clause B.1 over the product assets defined in clause 4.1.

5.3 No known exploitable vulnerabilities

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (a) and their applicability is presented in Table 2.

- **REQ-KEV-01:** The product shall incorporate only components, including third party and open source elements, for which no known exploitable vulnerabilities exist at the time of release.

Table 2: Mapping of No known exploitable vulnerabilities requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-KEV-01	X	X	X	X	X	X	X	X

NOTE: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.4 Secure by default configuration

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (b) and their applicability is presented in Table 3.

NOTE 1: "The CRA essential requirement laid out in Annex I, Part I (2) (b) foresees an applicability exception to the secure configuration by default in case there is an agreement between manufacturer and business user in relation to a tailor-made PwDE."

- **REQ-SBD-01:** The product shall be shipped with cybersecurity-hardened default settings and with its documented essential cybersecurity functions enabled by default. It shall provide a mechanism to restore the secure default configuration by the security/IT administrative user. Where automatic cybersecurity updates are supported, they shall be enabled by default.
- **REQ-SBD-02:** The product shall implement separation of privileges between its processes, services, and components to prevent compromise of one from resulting in unauthorised control of others performing the product's documented essential cybersecurity functions.
- **REQ-SBD-03:** When handling potentially malicious content, the product shall ensure that such evaluation occurs within an isolated or otherwise constrained execution environment designed to identify the malicious content and to prevent such content from compromising the product or the host system.
- **REQ-SBD-04:** The product shall employ memory safety or memory hardening protections, or equivalent mechanisms, for components performing the product's documented essential cybersecurity functions.
- **REQ-SBD-05:** Where the product makes use of an RDPS feature, the product shall comply with the requirements specified in Annex R, applied in accordance with to the UC where the product belongs or the product's assigned Security Profile.

Table 3: Mapping of Secure by default configuration requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-SBD-01	X	X	X	X	X	X	X	X
REQ-SBD-02		X	X	X	X		X	X
REQ-SBD-03		X	X	X	X		X	X
REQ-SBD-04		X	X	X	X		X	X
REQ-SBD-05	NOTE: For the applicability of this requirement, check Annex R.							

NOTE 2: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.5 Security updates

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (c) and their applicability is presented in Table 4.

NOTE 1: For clarification, the reader of the present document should take into consideration the following: (a) severity-based remediation timeframes (e.g. Vulnerability Handling [i.6]) apply to security updates, (b) detection rules updates follow a separate, manufacturer-documented release schedule, and (c) the integrity and authentication requirements (e.g. REQ-SU-01) apply equally to both update types.

- **REQ-SU-01:** The product shall ensure the authenticity, integrity and provenance of all externally sourced cybersecurity-relevant data before use, including but not limited to detection signatures, threat intelligence feeds, heuristic rules, detection models, configuration data, and software updates, where applicable. Validation shall occur both during retrieval and before the data is applied or executed.
- **REQ-SU-02:** Where the product supports automatic cybersecurity updates, the product shall, where technically feasible, either (i) provide a mechanism to temporarily postpone installation of a cybersecurity update before it is applied, (ii) allow update scheduling through authorised management systems, or (iii) provide mechanisms to ensure the update does not negatively impact the operating environment during update (e.g. update without rebooting).

- **REQ-SU-03:** The product shall receive regular updates to the product code, configuration and detection databases during its support period (as defined in [i.1] Article 13, Section 8 Paragraph 3) since the product is acquired by the user.

- **REQ-SU-04:** Where the product supports automatic cybersecurity updates, the product shall either (i) provide a clear and easy-to-use mechanism to disable automatic cybersecurity updates, or (ii) provide mechanisms to ensure the update does not negatively impact the operating environment during update (e.g. update without rebooting).

- **REQ-SU-05:** The product shall implement version control mechanisms to prevent unauthorised installation of cybersecurity updates or cybersecurity data that are older than the currently installed version, except where a controlled rollback to a previously verified and trusted version is performed by the product.

- **REQ-SU-06:** The product shall ensure that, in case of failed or interrupted update, no partially applied or unverified update remains active and the product reverts to a previously verified and trusted state.

- **REQ-SU-07:** The product shall implement mechanisms to prevent the deployment of faulty updates by ensuring that cybersecurity updates are validated for operational correctness prior to full deployment by, where technically feasible, applying:

  - staged or phased deployment of updates;

  - validation of successful update application and functional integrity before broad rollout;

  - automated rollback when abnormal behaviour or degradation is detected; and

NOTE 2: This aspect would be on-machine rollback.

  - automated suspension of update propagation when abnormal behaviour or degradation is detected.

NOTE 3: This aspect would be a decision of the update infrastructure.

- **REQ-SU-08:** Where an OS update triggers a false positive detection, the product shall still be in a state which allows updating the signature database or rule set to enable the new OS update to work as intended.

- **REQ-SU-09:** Where the product uses any undocumented or deprecated OS interfaces, the product shall identify and mitigate the risks associated with using undocumented/deprecated OS interfaces so essential product functionality remains controlled when OS behavior changes and the usage of said interfaces does not increase the OS attack surface and where possible, does not restrict usage of those very interfaces by other security related software.

NOTE 4: The risks are considered mitigated if all of the following apply:

  - the product is part of a combined and integrated solution containing an OS and the product;

  - the vendor ensures that the product and the associated OS are updated in lock-step and any OS interface changes are handled by the matching product version;

  - the combined solution (e.g. antivirus and OS) either does not allow installing another security-related software (e.g. SIEM), or where possible the usage of the OS interfaces by other security-related software is not restricted.

Table 4: Mapping of Security updates requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-SU-01	X	X	X	X	X	X	X	X
REQ-SU-02	X	X	X	X	X	X	X	X
REQ-SU-03	X	X	X	X	X	X	X	X
REQ-SU-04	X	X	X	X	X	X	X	X
REQ-SU-05		X	X	X	X		X	X
REQ-SU-06	X	X	X	X	X	X	X	X
REQ-SU-07	X	X	X	X	X	X	X	X
REQ-SU-08		X	X	X	X		X	X
REQ-SU-09		X	X	X	X		X	X

NOTE 5: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.6 Authentication and access control

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (d) and their applicability is presented in Table 5.

- **REQ-AAC-01:** The product shall implement tamper- and self-protection wherever feasible/applicable.
- **REQ-AAC-02:** The product shall require authentication for administrative operations leveraging existing OS or platform authentication mechanisms where available, including but not limited to configuration changes, protection deactivation, and policy modification.
- **REQ-AAC-03:** The product shall support and encourage the use of MFA for administrative IAM where technically feasible.
- **REQ-AAC-04:** The product shall support role-based or equivalent privilege control mechanisms to ensure that administrative access rights are restricted to the minimum privileges necessary for the assigned role.
- **REQ-AAC-05:** Where the product offers a web management console that is remote or local in a multi-user system, the product shall generate unique, non-predictable, and resistant to brute force attacks authentication tokens for the web management console that is remote or local in a multi-user system.
- **REQ-AAC-06:** Where the product offers a web management console that is remote or local in a multi-user system, the product shall not implement authentication tokens that do not expire for the web management console that is remote or local in a multi-user system.
- **REQ-AAC-07:** Where the product offers a web management console that is remote or local in a multi-user system, the product shall protect authentication tokens for the web management console that is remote or local in a multi-user system by:
  - transmitting them only over secure channels that provide confidentiality and integrity protection; and
  - implementing appropriate token protection mechanisms for the interface type (e.g. APIs, websockets, or other equivalents interfaces); and
  - preventing authentication tokens from appearing in audit events, logs, URLs, or error messages.
- **REQ-AAC-08:** Where the product offers a web management console that is remote or local in a multi-user system, the product shall implement protection against authentication token fixation by generating new authentication tokens when the authentication status changes for the web management console that is remote or local in a multi-user system.
- **REQ-AAC-09:** Where the product offers a web management console that is remote or local in a multi-user system, the product shall implement authorisation mechanisms that enforce role-based access controls to prevent privilege escalation.
- **REQ-AAC-10:** Where the product offers a web management console that is remote or local in a multi-user system, the product shall implement brute force protection (e.g. progressive delay or account lockout after repeated failed authentication attempts).
- **REQ-AAC-11:** The product shall not use shared default credentials.

Table 5: Mapping of Authentication and access control requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-AAC-01		X	X	X	X		X	X
REQ-AAC-02		X	X	X	X		X	X
REQ-AAC-03			X	X	X			X
REQ-AAC-04			X	X	X			X
REQ-AAC-05		X	X	X	X		X	X
REQ-AAC-06		X	X	X	X		X	X
REQ-AAC-07		X	X	X	X		X	X
REQ-AAC-08		X	X	X	X		X	X
REQ-AAC-09			X	X	X			X
REQ-AAC-10		X	X	X	X		X	X
REQ-AAC-11	X	X	X	X	X	X	X	X

NOTE: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.7 Confidentiality protection

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (e) and their applicability is presented in Table 6.

- **REQ-CON-01**:

The product shall protect stored sensitive data including, but not limited to, authentication credentials, cryptographic private keys, sensitive configuration parameters, or quarantined content by facilitating encryption either by product or by external systems for data storage of such data at rest using state of the art cryptography as defined in Annex K in a manner proportional to the capabilities of the execution environment.

- **REQ-CON-02:** Where the product stores authentication credentials or cryptographic private keys, the product shall protect such data against unauthorised access using mechanisms available in the execution environment (e.g. platform credential storage, hardware-backed key storage, administrative user rights or equivalent protection).

- **REQ-CON-03:** Where the product supports local malware processing without an RDPS, the product shall allow the user to disable transmitting telemetry which would:

  a) allow the vendor or third parties to derive the presence of a file; or

  b) upload the file to the vendor or a third party.

- **REQ-CON-04:** Where the product relies on an RDPS for malware processing and does not support equivalent local malware processing, the product shall be accompanied by documentation that describes this dependency informing the user of the residual risks associated with the absence of local malware processing, including the residual risk related to TH5.4 as defined in clause B.1.1.6.

Table 6: Mapping of Confidentiality protection requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-CON-01		X	X	X	X		X	X
REQ-CON-02		X	X	X	X		X	X
REQ-CON-03	X	X	X	X	X	X	X	X
REQ-CON-04	X	X	X	X	X	X	X	X

NOTE: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.8 Integrity protection

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (f) and their applicability is presented in Table 7.

- **REQ-INT-01:** The product shall verify the integrity of its own executable files and updates using digital signatures in accordance with the cryptographic requirements specified in Annex K, in a manner proportionate to the capabilities of the execution environment. Executable files and updates that fail such verification shall not be executed, applied, or installed, and the integrity verification failure shall be recorded as a cybersecurity-relevant event.
- **REQ-INT-02:** Where the product transmits data, such transmission shall be protected against unauthorised access and modification using secure communication mechanisms in accordance with the cryptographic requirements specified in Annex K in a manner proportional to the capabilities of the execution environment.

Table 7: Mapping of Integrity protection requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-INT-01		X	X	X	X		X	X
REQ-INT-02	X	X	X	X	X	X	X	X

NOTE: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.9 Data minimisation

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (g) and their applicability is presented in Table 8.

- **REQ-DM-01:** The product shall, by default, process personal data only to the extent necessary for its intended purpose and shall ensure that such processing is adequate, relevant and limited to what is necessary in relation to that intended purpose.

Table 8: Mapping of Data minimisation requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-DM-01	X	X	X	X	X	X	X	X

NOTE: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.10 Availability protection

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (h) and their applicability is presented in Table 9.

- **REQ-AP-01:** The product shall identify and document all critical cybersecurity relevant components of the product, including the rationale for determining which components are considered critical, based on its design, intended use, and declared cybersecurity objectives.
- **REQ-AP-02:** The product shall protect the availability of its documented essential cybersecurity functions during and after an operational incident by implementing resilience measures such that, when a cybersecurity-critical component required for those functions terminates unexpectedly or an update operation is interrupted, the product shall either: a) automatically re-establish that service/function (e.g. restart), or b) enter a degraded protection state in which (i) no unsafe update/content is applied, and (ii) the product notifies the security administrative user that protection is degraded.

- **REQ-AP-03:** Where the product controls externally accessible interfaces (e.g. interfaces that are not controlled by the OS) used to support its documented essential cybersecurity functions, the product shall implement measures to mitigate denial-of-service conditions that could impair those functions, such as allow-listing, rate limiting, request throttling, auto-scaling, resource usage controls, or equivalent protective mechanisms.

- **REQ-AP-04:** The product shall implement mechanisms to detect and prevent incorrect or unauthorised denylisting of legitimate entities (e.g. files, processes, services) depending on the UC:

  a) for all the UCs, the ability for authorised users or automated safeguards to review, override, or revert denylisting decisions in a controlled and auditable manner;

  b) for all the UCs, mechanisms to ensure that essential product components and trusted update sources cannot be permanently or irreversibly denylisted without verified justification

  c) for UCs with high SP (i.e. UC3, UC4, UC5) and where technically feasible, the validation of detection decisions prior to enforcement using multiple independent indicators; and

  d) for UCs with high SP (i.e. UC3, UC4, UC5), mechanisms to detect abnormal or large-scale denylisting behaviour affecting legitimate system functionality.

Table 9: Mapping of Availability protection requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-AP-01	X	X	X	X	X	X	X	X
REQ-AP-02		X	X	X	X		X	X
REQ-AP-03			X	X	X			X
REQ-AP-04	X	X	X	X	X	X	X	X

NOTE: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.11 Non-interference

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (i) and their applicability is presented in Table 10.

- **REQ-NI-01:** The product shall minimise the consumption of system or network resources (e.g. CPU, memory, storage, or bandwidth) such as it does not block the availability of services on the host system or connected networks.

- **REQ-NI-02:** The product documentation shall identify and notify in a clear way the physical/hardware and logical/software environment where the product is to be deployed and used, the physical requirements that the product needs to apply all its functions, the supported OS types and the minimum version of them.

Table 10: Mapping of Non-interference requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-NI-01	X	X	X	X	X	X	X	X
REQ-NI-02	X	X	X	X	X	X	X	X

NOTE: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.12 Attack surface minimisation

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (j) and their applicability is presented in Table 11.

- **REQ-ASM-01:** The product shall ensure that its processes, services, and components under the control of the manufacturer operate only with the privileges necessary to perform their documented essential cybersecurity functions within the execution environment.
- **REQ-ASM-02:** Where the product exposes local or network services for integration with other components or the underlying OS, the product shall ensure that such services are protected with authentication (where technically applicable) and access-controls.
- **REQ-ASM-03:** Where the product provides local or network services for optional integration with other components or the underlying OS, such services shall be disabled by default unless required for the product's functionality, intended purpose or explicitly enabled by an authorised entity (e.g. administrative user or management system).

Table 11: Mapping of Attack surface minimisation requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-ASM-01		X	X	X	X		X	X
REQ-ASM-02		X	X	X	X		X	X
REQ-ASM-03		X	X	X	X		X	X

NOTE: As stated clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.13 Exploit mitigation

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (k) and their applicability is presented in Table 12.

- **REQ-EM-01:** When enabled by the administrator, the product shall disclose identified incidents and possible steps to mitigate them to the security administrative user.
- **REQ-EM-02:** The product shall leverage OS security functionality to implement runtime protection measures to prevent unauthorised modification, code injection, or tampering with its critical cybersecurity-relevant components within the execution environment, and where prevention is not technically feasible, implement feasible measures to detect such attempts and to preserve the integrity and continued secure operation of the product.

Table 12: Mapping of Exploit mitigation requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-EM-01		X	X	X	X		X	X
REQ-EM-02			X	X	X			X

NOTE: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.14 Monitoring

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (l) and their applicability is presented in Table 13.

- **REQ-MON-01:** The product shall support logging and monitoring mechanism, or leverage OS's logging and monitoring mechanism, to record and, where technically feasible, store cybersecurity-critical events that may include detections, security updates, detection rule updates, configuration changes, and access to or modification of cybersecurity-critical data, services, or functions and, where technically feasible, provide access to such records to authorised entities (e.g. administrative users or management systems).
- **REQ-MON-02:** Where the product creates and stores logs or passes the logs to be stored by other components, the product shall maintain tamper-resistant logs of detections, updates, configuration changes, access to quarantined files, and preventing modification or deletion of these files without authorised administrative action.
- **REQ-MON-03:** Where the product creates and stores logs, the product shall implement integrity protection to detect unauthorised modification or deletion of recorded cybersecurity-relevant logs. Integrity protection controls may include the following:
  - cryptographically signing or hashing in accordance with the cryptographic requirements specified in Annex K in a manner proportional to the capabilities of the execution environment;
  - storing files in an append-only access-controlled location, and/or;
  - preventing modification or deletion without authorised administrative action.
- **REQ-MON-04:** Where the product creates and stores logs, the product shall restrict access to such records to authorised entities (e.g. administrative users, privileged users or management systems.
- **REQ-MON-05:** The product shall automatically notify authorised entities (e.g. administrative users or management systems) when cybersecurity-relevant events result in a change to the product's protection status.
- **REQ-MON-06:** Where the product records or transmits monitoring data beyond what is strictly necessary for its intended purpose, the product shall provide a mechanism allowing the authorised user to disable such additional monitoring.

Table 13: Mapping of Monitoring requirements applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-MON-01	X	X	X	X	X	X	X	X
REQ-MON-02	X	X	X	X	X	X	X	X
REQ-MON-03		X	X	X	X		X	X
REQ-MON-04		X	X	X	X		X	X
REQ-MON-05			X	X	X			X
REQ-MON-06	X	X	X	X	X	X	X	X

NOTE: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

5.15 Factory reset and data portability

This clause addresses the requirements in the CRA [i.1], Annex I, Part I (2) (m) and their applicability is presented in Table 14.

- **REQ-FRD-01:** The product shall notify its security/IT administrative and privileged users when it reaches its end of support period. If technically feasible, it shall also inform its end users.
- **REQ-FRD-02:** The product shall provide a mechanism or instructions that allow authorised administrative users to manage the permanent removal of the product data, including configuration, files and log entries stored by the product.

Table 14: Mapping of Factory rest and data portability applicability to the UCs & SPs

Requirements	UCs					SPs		
	UC1	UC2	UC3	UC4	UC5	Minimal	Medium	High
REQ-FRD-01	X	X	X	X	X	X	X	X
REQ-FRD-02	X	X	X	X	X	X	X	X

NOTE: As stated in clause B.2, SPs are inclusive: Medium includes Minimal, High includes Medium and Minimal.

6 Assessment criteria for compliance with technical requirements

6.1 Introduction to the assessment and compliance criteria

6.1.0 Assessment guide

This clause provides objective and reproducible assessment criteria to determine whether a product complies with the technical security requirements of clause 5.

For each cybersecurity requirements defined in clause 5, the following clauses specify assessment criteria to determine if the technical requirement is met.

The assessment criteria for each security requirements are described in a structured manner, as follows:

- **Assessment reference:** Refers to the identifier of the concerned technical requirement.
- **Assessment objective:** Defines the security property or capability that shall be verified, ensuring that the assessment remains focused on the intent of the requirement.
- **Assessment preparation:** Describes the environment, setup, and preconditions required before executing the test with a view to guaranteeing consistent evaluation results. It includes the following elements as applicable:
  - Test environment: Describe the hardware, software, and network setup used for the assessment, including versions, topology, and any relevant dependencies.
  - Preconditions: Specify any configurations, credentials, or operational states that should be established before the test (e.g. product initialised, certificates loaded, user roles created).
  - Required tools: Identify the tools or software necessary to perform the assessment (e.g. vulnerability scanners, protocol fuzzers, traffic analysers, static code analysers, cryptographic test suites).
  - Required information/documentation for the assessment: Specify all information that is necessary to perform the assessment.
  - Reference any vendor-provided setup guides, configuration instructions, or operational manuals, as well as any relevant standards or technical notes, that define how the product shall be configured or operated for the assessment.
- **Assessment activities:** Provides execution steps to be performed such that the same assessment verdict would be reached when repeating these steps now or at a later point in time. Assessment activities may include, as applicable:
  - Review information/documentation for the assessment to confirm that the described implementation matches the requirement (e.g. verify that the security architecture document specifies TLS 1.2 or higher for all external interfaces, or that the password policy aligns with the defined threshold).
  - Perform security functional tests to verify the completeness and correctness of the information/documentation for the assessment.

- Perform security functional or penetration tests to verify that implemented controls are correctly implemented e.g. to prevent unauthorised access or data modification (e.g. via attempting to log in with invalid credentials to test lockout enforcement or trying to modify protected configuration files without administrative privileges).
- Analyse code or binaries to identify potential security weaknesses or misconfigurations (e.g. perform static analysis to detect hardcoded credentials or use dynamic analysis tools to identify buffer overflow or injection vulnerabilities).
- Inspect configurations to ensure that required security parameters are correctly applied (e.g. check that weak cipher suites are disabled, two-factor authentication is enabled, and least-privilege access controls are configured in the system).
- Observe runtime behaviour to confirm that protections such as encryption, authentication, and integrity verification operate as intended (e.g. monitor network traffic to ensure data in transit is encrypted or observe system logs to verify successful validation of digital signatures during startup).

- **Assessment verdict:** Defines the pass/fail criteria.
  - **Pass:** The assessment is considered passed if the product demonstrably fulfils the requirement and meets the defined security thresholds. Examples of such thresholds include:
    - Minimum cryptographic strength (e.g. AES-128 or higher).
    - Password policy limits (e.g. minimum of 12 characters).
    - Login protection mechanisms (e.g. account lockout after five consecutive failed attempts).
    - Resistance to a specified attack potential (e.g. equivalent to CSA High/AVA_VAN.3 or higher).
  - **Fail:** The assessment is considered failed if the requirement is not fulfilled, or if the defined security thresholds are not achieved (e.g. insufficient key length, missing authentication enforcement, or inadequate resistance to the required attack potential).
- **Assessment evidence**: Defines the artefacts and documentation collected to demonstrate that the requirement has been assessed and fulfilled. The evidence shall be sufficient to enable independent verification of the assessment results and to demonstrate compliance with the relevant CRA [i.1] essential requirements. The supporting evidence include, where applicable:
  - Test or assessment reports showing the steps performed and results obtained.
  - Logs, configuration files, or audit traces demonstrating the implementation of the requirement.
  - Screenshots, captures, or console outputs confirming the correct execution or protection behaviour.
  - Relevant vendor or design documentation describing the applied security measures.

6.1.1 Inability to comply with assessment methodology

If an assessment activity for an individual requirement cannot be performed as described due to technical limitations arising from the product's intended purpose or operational environment, an alternative assessment approach may be applied, provided that it preserves the intent and objective of the specified methodology.

In such cases, the technical documentation shall include:

- a clear justification explaining why compliance with the prescribed assessment methodology was not feasible; and
- a detailed description of the alternative assessment methodology applied.

6.2 Appropriate level of cybersecurity

As stated in clause 5.2 of the present document and based on the Commission CRA draft legal guidance [i.13] point 7.2, the present document has been developed with the objective that by using the requirements from clauses 5.3 to 5.15, the corresponding requirement defined in the CRA [i.1], Annex I, Part I (1) is properly covered. Therefore, the same applies with the assessments presented in the following clauses and associated to each of the cybersecurity requirements present in clause 5.

6.3 No known exploitable vulnerabilities

6.3.1 REQ-KEV-01

- **Assessment reference: REQ-KEV-01**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product and all its constituent components, including third-party and open-source elements such as libraries, packages, embedded engines, signature-processing modules, cloud-connector modules, and any supporting software, contain no known exploitable vulnerabilities that were publicly identified at the time of the product's release. This includes confirming that:

  - All incorporated components are inventoried and traceable (e.g. via an SBOM).
  - Component versions used in the released product were checked against recognised vulnerability databases.
  - No component included in the release has an associated known exploitable vulnerability as defined in the CRA [i.1], Article 3 point (41).
  - Product vendor-specific vulnerability monitoring processes exist and are implemented.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed instance of the product under assessment, including the final build intended for market release. Hardware and operating system representative of intended customer environments (e.g. Windows® endpoints, Linux® servers, macOS® clients). Access to the product's build artefacts documentation (e.g. SBOM, dependency manifests, and component inventory). Network access for querying vulnerability databases as needed.

NOTE: Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

  - **Preconditions:** Product installed and initialised in accordance with vendor documentation. All update mechanisms (e.g. signature engine updates, cloud-reputation modules, etc.) disabled unless needed for component verification. Availability of vendor-provided release documentation (e.g. SBOM, third-party licence list, component version list and build metadata and/or secure development and vulnerability monitoring process description, etc.).
  - **Required Tools:** Vulnerability database query tools (e.g. local or online scanners enabling access to CVE/NVD, CERT, CISA KEV or other recognised databases). Analysis tools (e.g. supporting SPDX or CycloneDX formats SBOMs and other similar options). Static analysis tools capable of identifying embedded version information in binaries. Digital signature and hash verification tools for checking component integrity.
  - **Reference Documentation:** Vendor configuration and installation guides. Vendor dependency documentation or others such as SBOM.

- **Assessment Activities:**

  - **Documentation Review:** Examine the vendor documentation (e.g. SBOM) to confirm completeness and accuracy of listed components. Review vendor documentation describing the vulnerability monitoring and validation process applied before release. Verify that each component has an identified version and an enabled traceability method is used (e.g. hash).

  - **Vulnerability Verification:** For each listed component in the vendor documentation, query authoritative vulnerability databases (e.g. NVD, CVE.org, CISA KEV Catalogue, Vendor security advisories among other well-known and recognised possibilities). Document whether any entry corresponds to known exploitable vulnerability as defined in the CRA [i.1], Article 3 point (41).

  - **Binary and Configuration Analysis:** Where version numbers are not explicitly stated, use methods such as binary inspection or static analysis to extract component version information. Compare extracted versions with the vendor documentation (e.g. SBOM) to ensure alignment and detect undocumented components.

  - **Penetration-Oriented Checks (as applicable):** Attempt to identify unlisted open-source or third-party components through dynamic analysis, dependency tracing, scanning for known library signatures or similar methodologies. For any discovered but undocumented components, repeat the steps defined in the previous "Vulnerability Verification" activity.

  - **Evaluation of Vendor Vulnerability Monitoring:** Review evidence of the vendor's vulnerability check performed prior to release (e.g. internal reports, automated scan results). Confirm that checks were conducted at a date consistent (e.g. less than a month) with the product release timeline.

- **Assessment Verdict:**

  The assessment is **passed if all of the following occur**:

  - All components listed in the vendor's documentation (e.g. SBOM) correspond to the versions present in the product.

  - No component has a known exploitable vulnerability as defined in the CRA [i.1], Article 3 point (41), at the time of release according to authoritative sources.

  - No undocumented components are discovered, or if discovered, they are shown to have no known exploitable vulnerabilities.

  - Vendor-provided vulnerability monitoring documentation shows that the checks were performed within an appropriate timeframe preceding the release.

  The assessment is **failed if any of the following occur**:

  - One or more components are found to have KEVs at the time of release.

  - Component versions in the product do not match the vendor's documentation provided (e.g. SBOM) or are incomplete or misleading.

  - Undocumented components are identified and cannot be verified as vulnerability-free.

  - The product vendor cannot demonstrate that vulnerability checks were conducted prior to release.

  - Evidence indicates that exploitable vulnerabilities were known but not addressed.

- **Assessment Evidence:**

  The following evidence shall be collected to support assessment results:

  - The full product vendors' documentation (e.g. SBOM) used for the assessment, including component identifiers and versions.

  - Logs or reports of vulnerability database queries, including timestamps.

  - Scan results from the product vendors' documentation (e.g. SBOM) analysers, static analysis tools, binary inspection tools or similar tools.

- Screenshots, command-line outputs, or captures showing component identification.
- Documentation from the vendor demonstrating pre-release vulnerability verification.
- A consolidated assessment report detailing the steps performed, databases queried, vulnerabilities found (if any) and the final verdict justification

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.4 Secure by default configuration

6.4.1 REQ-SBD-01

- **Assessment reference: REQ-SBD-01**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product is shipped with cybersecurity-hardened default settings, with its documented essential cybersecurity functions enabled by default, and that the required configuration restoration mechanism and automatic update defaults are implemented as specified. This includes confirming that:

  - The product's default configuration applies cybersecurity-hardened settings without requiring post-installation intervention by the user.
  - The product's documented essential cybersecurity functions are active in the default configuration.
  - A mechanism exists to restore the secure default configuration, accessible to the security/IT administrative user.
  - Where automatic cybersecurity updates are supported, they are enabled by default.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, in its default state as delivered, with no post-installation configuration changes applied. The environment shall include monitoring capabilities appropriate to the product's deployment model to observe the product's default operational behaviour.
  - **Preconditions:** The product is in its default state following a clean installation. No administrative configuration changes or cybersecurity customisations are applied prior to testing. Vendor documentation identifying the product's documented essential cybersecurity functions and their expected default state is available. Where the product supports automatic cybersecurity updates, network or repository access is available to allow observation of the automatic update mechanism.
  - **Required Tools:** Mechanisms capable of enumerating the product's configuration and active functions, appropriate to the product's deployment environment; monitoring mechanisms capable of observing the product's default operational behaviour, appropriate to the product's deployment model.
  - **References:** Vendor installation or setup documentation specifying default cybersecurity configurations. Vendor documentation identifying the product's documented essential cybersecurity functions. Vendor cybersecurity hardening guidelines, if provided.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to confirm which cybersecurity functions are documented as essential and verify that these are stated to be enabled by default. Verify that the documentation describes the mechanism for restoring the secure default configuration and identifies the security/IT administrative user as the entity authorised to invoke it. Where the product supports automatic cybersecurity updates, confirm that the documentation states these are enabled by default.

- **Configuration Inspection:** Inspect the product's configuration following a clean installation in its default state and verify that all documented essential cybersecurity functions are active. Confirm that the default configuration reflects the documented cybersecurity-hardened settings. Where the product supports automatic cybersecurity updates, verify that this function is active in the default configuration.
- **Functional Testing:** Verify that the mechanism for restoring the secure default configuration is accessible to the security/IT administrative user and produces a configuration that matches the documented secure default state. Attempt to invoke the restoration mechanism using a non-administrative account and verify that such access is prevented. Where the product supports automatic cybersecurity updates, verify that the update mechanism initiates as expected in the default configuration without requiring manual activation.
- **Runtime Observation:** Observe the product's behaviour during normal operation in its default state to confirm that documented essential cybersecurity functions remain active during routine operation in the default configuration.

- **Assessment Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product is delivered in a cybersecurity-hardened configuration with all documented essential cybersecurity functions active by default, as confirmed through the assessment activities.
  - The product provides a mechanism to restore the secure default configuration that is accessible to the security/IT administrative user and not accessible to non-administrative accounts.
  - Where automatic cybersecurity updates are supported, the product has such updates enabled by default.

  The assessment is **failed if any of the following occur**:

  - The product's default configuration does not enable one or more documented essential cybersecurity functions without post-installation intervention.
  - No mechanism exists to restore the secure default configuration, or the mechanism is not accessible to the security/IT administrative user.
  - The restoration mechanism is accessible to non-administrative accounts.
  - Where automatic cybersecurity updates are supported, such updates are not enabled by default.
  - Documentation does not identify the product's essential cybersecurity functions or their expected default state, making independent verification impossible.

- **Assessment Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation identifying the product's documented essential cybersecurity functions and their expected default state, including the mechanism for restoring the secure default configuration.
  - Configuration inspection outputs confirming that documented essential cybersecurity functions are active in the default state and that the default configuration reflects the documented cybersecurity-hardened settings.
  - Test reports documenting the steps taken to verify the restoration mechanism, the account access controls applied to it, and the results obtained.
  - Monitoring records confirming the active state of essential cybersecurity functions during default operation and, where applicable, the activation of the automatic cybersecurity update mechanism.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.4.2 REQ-SBD-02

- **Assessment reference: REQ-SBD-02**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product implements privilege separation between its internal processes, services, and components such that a compromise of one does not result in unauthorised control over others performing the product's documented essential cybersecurity functions. This includes confirming that:

  - The product's internal architecture enforces distinct privilege boundaries between components.
  - Components performing essential cybersecurity functions operate under privilege levels that are not shared with or accessible by unrelated or lower-privilege components.
  - The product's documented architecture accurately reflects the implemented privilege separation model.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment (e.g. standalone workstation, managed endpoint, or embedded system as applicable).
  - **Preconditions:** The product is installed and running in its default operational configuration. Vendor architecture documentation describing the product's internal component structure and privilege model is available.
  - **Required Tools:** Mechanisms capable of enumerating product processes and their assigned privilege levels, appropriate to the product's deployment environment; mechanisms capable of monitoring inter-component communication or equivalent runtime activity during product operation, appropriate to the product's deployment environment; mechanisms capable of simulating inter-component access or privilege escalation attempts, where available in the product's deployment environment.
  - **References:** Vendor architecture or design documentation describing internal component separation and privilege assignment. Vendor security hardening guide, if provided.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor architecture documentation to confirm that the product's design implements privilege separation between its processes, services, and components. Verify that the documentation identifies which components perform essential cybersecurity functions and specifies the privilege level at which each operates.
  - **Configuration Inspection:** Enumerate the product's running processes and services and inspect their assigned privilege levels and inter-process communication mechanisms using available mechanisms appropriate to the product's deployment environment. Where direct enumeration is not feasible, verify through vendor-provided architecture documentation, diagnostic interfaces, or equivalent evidence that components performing essential cybersecurity functions operate at distinct privilege levels from unrelated or lower-privilege components, and document the reason direct enumeration was not performed.
  - **Functional Testing:** For each component that directly communicates with or shares an interface with a component performing essential cybersecurity functions, simulate a compromise using available mechanisms appropriate to the product's deployment environment and verify that such a compromise cannot be used to gain unauthorised access to or control over components performing essential cybersecurity functions. Verify that no unintended or unauthorised privilege escalation path exists between components beyond those explicitly documented and functionally justified. Where simulation mechanisms are not available in the product's deployment environment, the assessor shall verify through architecture documentation review and runtime observation alone that no unintended or unauthorised privilege escalation path exists between components, and shall document the reason simulation testing was not performed.

- **Runtime Observation:** Monitor inter-process communication or equivalent runtime activity during product operation, using available monitoring mechanisms in the deployment environment, to confirm that access between components is controlled and that interactions between components are constrained to documented and necessary communication channels.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Demonstrated privilege separation between the internal processes, services, and components of the product.
  - The product does not permit a compromise of one component to result in unauthorised access to or control over components performing essential cybersecurity functions, as confirmed through the assessment activities.
  - Product documentation is delivered and it accurately reflects the implemented privilege separation model.

  The assessment is **failed if any of the following occur**:

  - Components share privilege contexts in a manner not justified by their documented function.
  - A simulated compromise of one component can be demonstrated to result in unauthorised access to or control over components performing essential cybersecurity functions, or, where simulation was not performed due to the unavailability of suitable simulation mechanisms in the product's deployment environment, architecture documentation or runtime observation reveals the existence of privilege escalation paths between components that are not documented or functionally justified.
  - Documentation does not describe privilege separation or contradicts the observed implementation.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Process and privilege enumeration reports showing the privilege levels assigned to product components, or where direct enumeration is not feasible, vendor-provided architecture documentation, diagnostic interface outputs, or equivalent evidence demonstrating distinct privilege levels between components, together with a documented justification for why direct enumeration could not performed.
  - Architecture documentation describing the internal component separation model.
  - Test reports documenting the steps taken to simulate lateral access between components and the results obtained, or where simulation was not feasible, a documented justification and the results of architecture documentation review and runtime observation performed in its place.
  - Runtime monitoring traces confirming that inter-component communication is restricted to documented channels.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.4.3 REQ-SBD-03

- **Assessment reference: REQ-SBD-03**

- **Assessment Objective:**

  The objective of this assessment is to verify that the product ensures that the execution of potentially malicious content occurs within an isolated or otherwise constrained execution environment designed to prevent such content from compromising the product or the host system. This includes confirming that:

  - The product employs an isolation or containment mechanism when executing content through the execution path intended for potentially malicious content handling.

- The constrained environment prevents executed content from accessing or affecting host system resources, product components, or data outside the scope of the execution environment.
- The isolation mechanism operates as intended across the product's supported deployment configurations.

- **Assessment Preparation:**
  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment. The environment shall include monitoring capabilities at the available observation boundary. For products where isolation occurs on the host system, this includes host-level file system access monitoring, process creation monitoring, and network traffic monitoring; for products where isolation occurs outside the host system (e.g. cloud-delivered or SaaS-based products), monitoring shall be performed through vendor-provided monitoring interfaces, API responses, or equivalent mechanisms.
  - **Preconditions:** The product is installed and running in its default operational configuration. Inert or non-malicious test content items that trigger the product's execution mechanisms through their content, format, or other characteristics recognised by the product are prepared prior to testing. Monitoring and logging are active before testing begins. No restrictions are applied that would prevent observation of the product's execution behaviour.
  - **Required Tools:** Monitoring mechanisms appropriate to the product's deployment model (e.g. host-level monitoring utilities for products where isolation occurs on the host system, or vendor-provided monitoring interfaces for products where isolation occurs outside the host system); network monitoring tools capable of detecting outbound communication from the execution environment, where applicable; inert test content items as described in the "Preconditions" preparation.
  - **References:** Vendor documentation describing the isolation or containment mechanism used during content execution. Vendor operational manual describing how content execution is handled and the associated execution environments.
- **Assessment Activities:**
  - **Documentation Review:** Review vendor documentation to confirm that the product's potentially malicious-content execution workflow is designed to operate within an isolated or constrained execution environment. Verify that the documentation describes the nature of the isolation mechanism, its scope of protection, and the specific resources or components it is designed to protect against unauthorised access by executed content.
  - **Configuration Inspection:** Inspect the product's configuration to verify that isolation or containment controls are active during content execution through the malicious-content execution workflow. Confirm that the constrained environment is not disabled or bypassable through default configuration settings.
  - **Functional Testing:** Present the prepared test content to the product in a manner appropriate to its execution model and monitor for unintended effects outside the constrained execution environment at the available observation boundary (e.g. unauthorised file system modification, process creation outside the isolated context, network communication not attributable to documented product behaviour or other possible effects).
  - **Runtime Observation:** Observe activity at the available observation boundary during content execution and related product operation to confirm that no interactions outside the constrained environment occur.
- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:
  - The product's execution path for potentially malicious content handling operates within an isolated or constrained execution environment, as confirmed through the assessment activities.
  - The product demonstrates no unintended effects outside the constrained execution environment at the available observation boundary during content execution.
  - The product provides documentation that accurately describes the isolation mechanism and its protective scope.

The assessment is **failed if any of the following occur**:

- No isolation or containment mechanism is in place for content execution through the malicious-content execution workflow.
- Unintended effects outside the constrained execution environment are observed at the available observation boundary during content execution.
- The isolation mechanism can be bypassed through default configuration or normal product operation.
- Documentation does not describe the isolation mechanism, or it contradicts the observed isolation behaviour.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Product vendor documentation describing the isolation or constrained execution mechanism and its protective scope.
  - Monitoring traces recorded during the assessment, covering deliberate content execution and related product operation, confirming no interactions outside the constrained environment at the available observation boundary; where host-level monitoring was not available, vendor-provided logs, API responses, or equivalent evidence shall be provided in its place.
  - Test reports documenting the test content used, the execution steps performed, and the results obtained.
  - Configuration inspection outputs confirming that isolation controls are active during content execution.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.4.4 REQ-SBD-04

- **Assessment reference: REQ-SBD-04**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product employs memory safety or memory hardening protections, or equivalent mechanisms, for components performing the product's documented essential cybersecurity functions. This includes confirming that:

  - Components performing essential cybersecurity functions are protected against common memory-based exploitation techniques (e.g. buffer overflows, heap corruption, or code injection).
  - Memory safety or hardening mechanisms or equivalent mechanisms are applied to those components and are not disabled in the deployed configuration.
  - The vendor has documented which components perform essential cybersecurity functions and the corresponding memory protections applied to each.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment. Where binary or code analysis is performed as part of the assessment, a static and/or dynamic analysis environment shall be available. For products where binaries are not directly accessible to the assessor (e.g. cloud-delivered or SaaS-based products), assessment shall rely on vendor-provided documentation, build attestations, or equivalent evidence as described in the actions under the "Assessment Activities".
  - **Preconditions:** The product is installed in its default operational configuration. Vendor documentation identifying the components performing essential cybersecurity functions and the memory safety or hardening mechanisms applied to each is available prior to testing. Where applicable, product binaries or components are accessible for analysis.

- **Required Tools:** Analysis mechanisms capable of verifying the presence of memory protection features (e.g. ASLR, DEP/NX, stack canaries, CFI, or equivalent platform or language-level protections); dynamic analysis or fuzzing tools, where the product's interface model and deployment environment permit their application.
- **References:** Vendor documentation identifying components performing essential cybersecurity functions and specifying the memory safety or hardening mechanisms or equivalents mechanisms applied. Vendor build and deployment configuration documentation.

- **Assessment Activities:**
  - **Documentation Review:** Review vendor documentation to confirm which components perform essential cybersecurity functions and what memory safety or memory hardening or equivalent mechanisms are applied to those components. Where the vendor employs memory-safe programming languages or runtime environments as the primary protection mechanism, verify this claim through documentation and, where feasible, code or binary review.
  - **Binary and Code Analysis:** Where product binaries or components are accessible, analyse those identified as performing essential cybersecurity functions to verify the presence of applicable memory safety or memory hardening mechanisms (e.g. stack canaries, ASLR, DEP/NX, CFI, or equivalent platform or language-level protections). Confirm that these protections are present and not disabled in the deployed build. Where product binaries are not directly accessible, the assessor shall request and evaluate vendor-provided build attestations, third-party audit reports, or equivalent documented evidence confirming the presence of the required protections, and shall document the reason direct binary analysis was not performed.
  - **Configuration Inspection:** Verify that memory protection features are not disabled in the product's build configuration, deployment settings, or runtime environment. Confirm that no configuration option available to the end user or privileged user disables these protections for components performing essential cybersecurity functions.
  - **Functional Testing:** Where the product's interface model and deployment environment allows it, perform controlled testing against the interfaces of components performing essential cybersecurity functions, and verify that the product does not exhibit behaviour indicative of exploitable memory safety weaknesses (e.g. uncontrolled crashes, memory corruption symptoms, or unintended code execution). Where controlled testing is not technically feasible given the product's deployment model, the assessor shall document the reason and rely on binary analysis and documentation review to confirm the presence of memory protections.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - It is demonstrated that the product employs memory safety or hardening protections, or equivalent mechanisms, for all components performing essential cybersecurity functions, as confirmed through the assessment activities.
  - The product has such protections enabled in the deployed configuration.
  - Product vendor provides documentation that accurately identifies the relevant components and the protections applied.

  The assessment is **failed if any of the following occur**:

  - Components performing essential cybersecurity functions lack any memory safety or hardening mechanisms or equivalent protections, as confirmed through binary analysis, documentation review, or vendor-provided evidence.
  - Memory protection features are absent or disabled in the deployed configuration.
  - Documentation does not identify the relevant components or the protections applied, and no equivalent vendor-provided evidence is available, making independent verification impossible.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation identifying components performing essential cybersecurity functions and the memory safety or memory hardening or equivalent mechanisms applied to each.
  - Binary analysis reports confirming the presence of memory protection features for the relevant components, or where direct binary analysis was not feasible, vendor-provided build attestations, third-party audit reports, or equivalent documented evidence provided in its place, together with a documented justification for why direct analysis was not performed.
  - Build and deployment configuration documentation confirming that memory protections are enabled in the deployed configuration.
  - Functional test or controlled testing reports documenting the steps performed and results obtained, or where controlled testing was not technically feasible, a documented justification for why it was not performed.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.4.5 REQ-SBD-05

- **Assessment reference: REQ-SBD-05**

NOTE: The assessment for this requirement depends on the UC and or SP where the product under assessment belongs to. For this reason, in order to know the assessments needed, proceed to Annex R where the specific applicability of the RDPS-related requirements is presented in clause R.3.2.

6.5 Security updates

6.5.1 REQ-SU-01

- **Assessment reference: REQ-SU-01**
- **Assessment Objective:**

  The objective of this assessment is to ensure that the product validates the authenticity, integrity, and provenance of all externally sourced cybersecurity-relevant data both during retrieval and before the data is applied or executed. This includes confirming that:

  - The product applies validation mechanisms to all categories of externally sourced cybersecurity-relevant data applicable to its architecture and functionality, including but not limited to detection signatures, threat intelligence feeds, heuristic rules, detection models, configuration data, and software updates (e.g. detection rules, security or others).
  - Validation covers authenticity, integrity, and provenance of the data.
  - Validation is performed at two points: during retrieval and before application or execution.
  - Data that fails validation is not applied or executed by the product.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, with the ability to observe data retrieval processes at the available observation boundary. For products where data retrieval occurs through vendor-managed infrastructure (e.g. cloud-delivered or SaaS-based products), monitoring shall be performed through vendor-provided interfaces, logs, or equivalent mechanisms.

- **Preconditions:** The product is installed and running in its default operational configuration. Vendor documentation describing the validation mechanisms applied to each category of externally sourced cybersecurity-relevant data is available prior to testing. The product is connected to or configured to retrieve externally sourced data as it would in normal operation. Test data items with deliberately invalidated authenticity, integrity, or provenance indicators are prepared prior to testing, appropriate to the data categories applicable to the product.
- **Required Tools:** Mechanisms capable of observing data retrieval processes at the available observation boundary; mechanisms capable of presenting tampered, unsigned, or otherwise invalidated data to the product for validation testing; logging or audit mechanisms to capture the product's response to validation failures.
- **References:** Vendor documentation describing the validation mechanisms applied to externally sourced cybersecurity-relevant data, including the cryptographic or other methods used.

- **Assessment Activities:**
  - **Documentation Review:** Review vendor documentation to confirm that the product is designed to validate the authenticity, integrity, and provenance of all categories of externally sourced cybersecurity-relevant data applicable to its architecture. Verify that the documentation specifies which validation mechanisms are applied to each data category and confirms that validation occurs both during retrieval and before application or execution. Where a data category listed in the requirement is not applicable to the product, verify that the vendor documentation explicitly identifies this and provides justification.
  - **Configuration Inspection:** Inspect the product's configuration to verify that validation mechanisms are active for all applicable categories of externally sourced cybersecurity-relevant data. Confirm that validation cannot be disabled through default or readily accessible configuration settings.
  - **Functional Testing:** For representative applicable categories of externally sourced cybersecurity-relevant data, present the product with test data items in which authenticity, integrity, or provenance indicators have been deliberately invalidated (e.g. tampered content, absent or invalid signatures, or data from an unrecognised source). Verify that the product rejects such data and does not apply or execute it. Additionally, verify that validation is enforced at both stages, during retrieval and before application or execution. Where the product's architecture permits observation of both validation stages at the available observation boundary, verify through monitoring or logging that validation is performed during retrieval and again before application or execution. Where only one stage is observable, the assessor shall verify the two-stage validation through vendor architecture documentation and product logs, and shall document which stage was directly verified and which was verified through documentation.
  - **Runtime Observation:** During normal product operation, observe the data retrieval and update process at the available observation boundary to confirm that validation steps are consistently executed as part of the product's routine operation and are not bypassed during normal update or data ingestion cycles.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product validates the authenticity, integrity, and provenance of all applicable categories of externally sourced cybersecurity-relevant data, as confirmed through the "Assessment Activities".
  - The product rejects data that fails validation and does not apply or execute such data.
  - The product performs validation at both the retrieval stage and before application or execution.
  - The product provides documentation that accurately describes the validation mechanisms applied to each applicable data category.

  The assessment is **failed if any of the following occur**:

  - The product applies or executes externally sourced cybersecurity-relevant data without validating its authenticity, integrity, or provenance.
  - The product accepts and applies data in which authenticity, integrity, or provenance indicators have been deliberately invalidated during testing.

- Validation is performed at only one of the two required stages, during retrieval or before application or execution.
- Documentation does not describe the validation mechanisms applied to applicable data categories, or contradicts the observed product behaviour.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation describing the validation mechanisms applied to each applicable category of externally sourced cybersecurity-relevant data.
  - Monitoring traces or equivalent records captured at the available observation boundary, confirming that data retrieval processes are subject to validation.
  - Test reports documenting the invalidated data items used, the steps performed, and the product's observed response to each.
  - Logs or audit records produced by the product confirming rejection of data that failed validation.
  - Configuration inspection outputs confirming that validation mechanisms are active and not disabled.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.5.2 REQ-SU-02

- **Assessment reference: REQ-SU-02**

NOTE 1: The assessment is not applicable if the product does not support automatic cybersecurity updates, provided that vendor documentation confirms this and the assessor has documented the outcome accordingly. In this case, only the necessary Supporting Evidence actions need to be carried out to justify this outcome.

NOTE 2: For the purposes of this assessment, automatic cybersecurity updates include both security updates (software patches addressing product vulnerabilities) and detection rules updates (malware signatures, heuristic rules, detection database), as defined in clause 3.1.

- **Assessment Objective:**

  The objective of this assessment is to verify that, where the product supports automatic cybersecurity updates, the product implements at least one of the following mechanisms: (i) a mechanism to temporarily postpone installation of a cybersecurity update before it is applied, where technically feasible, (ii) a mechanism to allow update scheduling through authorised management systems, where technically feasible, or (iii) mechanisms to ensure the update does not negatively impact the operating environment during update. This includes confirming that:

  - The product implements at least one of the three required mechanisms where automatic cybersecurity updates are supported.
  - Where a postponement mechanism is provided, it allows temporary delay of a specific update before it is applied, and is available where technically feasible.
  - Where scheduling through authorised management systems is provided, it allows authorised management systems to control when updates are applied, and is available where technically feasible.
  - Where a non-disruptive update mechanism is provided, it demonstrably prevents negative impact on the operating environment during update installation.
  - The implemented mechanism operates as documented across the product's supported deployment configurations.

NOTE 3: For the purposes of this assessment, negative impact refers to unintended or avoidable interruption, degradation, or loss of functionality in the operating environment during update installation, such as forced reboots, service outages, or loss of protection.

- **Assessment Preparation:**
  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, configured to support automatic cybersecurity updates. The environment shall include mechanisms to observe the update process and its effects on the operating environment at the available observation boundary.
  - **Preconditions:** The product is installed and running in its default operational configuration. Confirm whether the product supports automatic cybersecurity updates:
    - Where automatic updates are not supported, this assessment is not applicable and no verdict is assigned; the assessor shall document this outcome and retain vendor documentation confirming that automatic updates are not supported.
    - Where automatic updates are supported, vendor documentation describing the implemented mechanism, whether postponement, scheduling, or non-disruptive update, is available prior to testing. The product is configured to perform automatic cybersecurity updates in its default or normal operational state.
  - **Required Tools:** Mechanisms capable of observing the update process and its effects on the operating environment at the available observation boundary; mechanisms capable of triggering or simulating an automatic update cycle where technically feasible; logging or audit mechanisms to capture update process behaviour and any effects on the operating environment.
  - **References:** Vendor documentation describing the automatic update mechanism and the implemented postponement, scheduling, or non-disruptive update controls. Vendor installation or operational guide, if provided.
- **Assessment Activities:**
  - **Documentation Review:** Review vendor documentation to confirm that the product supports automatic cybersecurity updates and that at least one of the three required mechanisms is implemented. Verify that the documentation accurately describes the implemented mechanism and its intended behaviour. Where a postponement mechanism is provided, confirm that the documentation describes how it is accessed and operated and under what conditions postponement is supported. Where scheduling through authorised management systems is provided, confirm that the documentation describes how scheduling is configured and which management systems are supported. Where a non-disruptive update mechanism is provided, confirm that the documentation describes how updates are applied without negatively impacting the operating environment.
  - **Configuration Inspection:** Inspect the product's configuration to verify that the implemented mechanism is active and accessible in the default operational configuration. Where a postponement mechanism is provided, confirm that it is accessible without requiring specialised technical knowledge. Where scheduling is provided, confirm that it is accessible to authorised management systems as described in vendor documentation. Where a non-disruptive update mechanism is provided, confirm that it is enabled by default and applies to automatic cybersecurity updates as described in vendor documentation.
  - **Functional Testing:** Where a postponement mechanism is provided, trigger or simulate an automatic update cycle and verify that the postponement mechanism successfully delays installation of the update before it is applied. Where scheduling through authorised management systems is provided, verify that the scheduling mechanism correctly controls the timing of update installation as configured through an authorised management system. Where a non-disruptive update mechanism is provided, trigger or observe an automatic update cycle and verify that the update is applied without causing negative impact on the operating environment (e.g. without requiring a reboot, without interrupting active services, or without causing observable degradation of product functions during the update). Where triggering an update cycle is not technically feasible in the test environment, the assessor shall verify the implemented mechanism's behaviour through product logs, vendor-provided evidence, or equivalent documentation, and shall document the reason direct observation was not possible.

- **Runtime Observation:** Where a non-disruptive update mechanism is provided, observe the product's behaviour during normal operation to confirm that automatic update cycles behave consistently with the documented non-disruptive behaviour. Where postponement or scheduling mechanisms are provided, observe the update process at the available observation boundary to confirm that the implemented mechanism operates as documented during routine update cycles.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product supports automatic cybersecurity updates and implements at least one of the three required mechanisms, as confirmed through the assessment activities.
  - Where a postponement mechanism is provided, the product demonstrably delays installation of a specific update before it is applied, where technically feasible.
  - Where scheduling through authorised management systems is provided, the product demonstrably allows authorised management systems to control the timing of update installation, where technically feasible.
  - Where a non-disruptive update mechanism is provided, the product demonstrably applies automatic updates without causing negative impact on the operating environment.
  - The product provides documentation that accurately describes the implemented mechanism and its behaviour.

  The assessment is **failed if any of the following occur**:

  - The product supports automatic cybersecurity updates but implements none of the three required mechanisms.
  - Where implemented, the postponement mechanism does not successfully delay installation of the update before it is applied.
  - Where implemented, the scheduling mechanism does not allow authorised management systems to control the timing of update installation.
  - Where implemented, the non-disruptive update mechanism causes observable negative impact on the operating environment during update installation, inconsistent with the vendor's documentation.
  - The implemented mechanism does not operate as described in vendor documentation or does not meet the applicable requirement conditions.
  - Documentation does not describe the implemented mechanism or contradicts the observed product behaviour.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation confirming whether the product supports automatic cybersecurity updates and describing the implemented mechanism.
  - Where the product does not support automatic updates, vendor documentation confirming this, retained as the basis for the not applicable outcome.
  - Configuration inspection outputs confirming that the implemented mechanism is active and accessible in the default operational configuration.
  - Test reports documenting the steps performed to verify the postponement, scheduling, or non-disruptive update behaviour, and the results obtained.
  - Evidence demonstrating the activation and effect of the implemented mechanism, confirming that it operates as required.

- Logs or audit records capturing update process behaviour and any observed effects on the operating environment, or where direct observation was not feasible, vendor-provided evidence provided in its place together with a documented justification.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.5.3 REQ-SU-03

- **Assessment reference: REQ-SU-03**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product receive regular updates to its code, configuration and detection databases during its support period as defined in the CRA [i.1]. This includes confirming that:

  - The product ensures regular, secure, and reliable delivery of updates to the product code, configuration data and detection artefacts (e.g. signatures, heuristics, behavioural models, machine-learning models).
  - The product maintains this update capability for the full declared support period after acquisition by the user.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, configured to support automatic cybersecurity updates. Network connectivity representative of a typical user environment (e.g. internet-connected, optionally via proxy or firewall). The environment shall include mechanisms to observe the update process and its effects on the operating environment at the available observation boundary.
  - **Preconditions:** The product is installed and running in its default operational configuration. The product's support period status is verifiable (e.g. valid subscription, licence, or entitlement). Automatic update mechanisms are enabled using default settings, unless the product explicitly relies on manual updates. Administrative access credentials are available for inspecting update configurations and logs. Mechanism to modify date information in the product to simulate end of period of support.
  - **Required Tools:** Mechanisms capable of observing the update process and its effects on the operating environment at the available observation boundary; mechanisms capable of triggering or simulating an automatic update cycle where technically feasible; logging or audit mechanisms to capture update process behaviour and any effects on the operating environment.
  - **Reference Documentation:** Vendor-provided update and lifecycle documentation, including update policy and cadence description, supported update channels and mechanisms, End-of-support statements. Product security architecture or design documentation describing update trust and integrity mechanisms. CRA [i.1] content related to support period obligations.

- **Assessment Activities:**

  - **Documentation Review:** Verify that the vendor documentation explicitly defines the update scope (code, configuration, detection databases), the declared support period after acquisition, confirm that update mechanisms are described as operational for the full support period and when possible, the update frequency or triggering conditions and procedures for exceptional updates (e.g. emergency threat response) are documented.
  - **Configuration and Capability Inspection:** Inspect product configuration to confirm that update functionality is enabled by default or can be enabled by the user, it cannot be disabled in a manner that contradicts the product's intended security function without explicit user action. Verify that update sources (servers, repositories) are predefined and, if possible, securely configurable.
  - **Update Execution Testing:** Trigger or observe at least one update cycle for each of the following, where applicable: a) detection database update, b) configuration update and c) product code or engine update. Verify that updates are successfully retrieved and applied and logged with date/time, version identifiers, and status. Where automatic updates are supported, confirm that updates occur without requiring manual intervention.

- **Support Period Validation:** Verify that the product explicitly reports update eligibility status (e.g. active, expired). Confirm that update attempts during the declared support period are not artificially restricted. Validate that update mechanisms are not blocked until the end of the product's support period.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product demonstrably supports regular updates to product code, configuration data and detection databases.
  - Update mechanisms are operational and functional during the declared support period after acquisition.
  - Update activity is observable and auditable through logs or version information.
  - No evidence is found that undermines the availability, security, or continuity of updates during the support period.

  The assessment is **failed if any of the following occur**:

  - One or more required update categories (i.e. product code, configuration data, detection databases) is not updated.
  - Updates are not available or cease during the declared support period.
  - Updates require undocumented manual procedures for standard operation.
  - Evidence indicates that the update process is non-operational during the support period.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting update scenarios executed, results obtained and dates observed.
  - Logs or system records showing successful update checks, installations and version changes over time.
  - Configuration files or screenshots demonstrating update enablement and parameters.
  - Relevant vendor documentation stating update policies, mechanisms, and support period commitments.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.5.4 REQ-SU-04

- **Assessment reference: REQ-SU-04**

NOTE 1: The assessment is not applicable if the product does not support automatic cybersecurity updates, provided that vendor documentation confirms this and the assessor has documented the outcome accordingly. In this case, only the necessary Supporting Evidence actions need to be carried out to justify this outcome.

NOTE 2: For the purposes of this assessment, automatic cybersecurity updates include both security updates (software patches addressing product vulnerabilities) and detection rules updates (malware signatures, heuristic rules, detection database), as defined in clause 3.1.

- **Assessment Objective:**

  The objective of this assessment is to verify that, where the product supports automatic cybersecurity updates, the product either provides a clear and easy-to-use mechanism to disable automatic cybersecurity updates, or provides mechanisms to ensure that updates do not negatively impact the operating environment during installation. This includes confirming that:

  - The product implements at least one of the two required mechanisms where automatic cybersecurity updates are supported.

- Where a disable mechanism is provided, it is accessible and operable by the end user without requiring specialised technical knowledge.

- Where non-disruptive update mechanisms are provided, they demonstrably prevent negative impact on the operating environment during update installation.

NOTE 3: For the purposes of this assessment, negative impact refers to unintended or avoidable interruption, degradation, or loss of functionality in the operating environment during update installation, such as forced reboots, service outages, or loss of protection.

- The implemented mechanism operates as documented across the product's supported deployment configurations.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, configured to support automatic cybersecurity updates. The environment shall include mechanisms to observe the update process and its effects on the operating environment at the available observation boundary.

  - **Preconditions:** The product is installed and running in its default operational configuration. Confirm whether the product supports automatic cybersecurity updates or not:

    - Where automatic updates are NOT supported, this assessment is not applicable and no verdict is assigned; the assessor shall document this outcome and retain vendor documentation confirming that automatic updates are not supported.

    - Where automatic updates are supported, vendor documentation describing the implemented mechanism, either the disable mechanism or the non-disruptive update mechanism, is available prior to testing. The product is configured to perform automatic cybersecurity updates in its default or normal operational state.

  - **Required Tools:** Mechanisms capable of observing the update process and its effects on the operating environment at the available observation boundary; mechanisms capable of triggering or simulating an automatic update cycle where technically feasible; logging or audit mechanisms to capture update process behaviour and any effects on the operating environment.

  - **References:** Vendor documentation describing the automatic update mechanism and the implemented opt-out or non-disruptive update controls. Vendor installation or operational guide, if provided.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to confirm that the product supports automatic cybersecurity updates and that at least one of the two required mechanisms is implemented. Verify that the documentation accurately describes the implemented mechanism and its intended behaviour. Where a disable mechanism is provided, confirm that the documentation describes how it is accessed and operated. Where a non-disruptive update mechanism is provided, confirm that the documentation describes how updates are applied without negatively impacting the operating environment.

  - **Configuration Inspection:** Inspect the product's configuration to verify that the implemented mechanism is active and accessible in the default operational configuration. Where a disable mechanism is provided, confirm that it is accessible to the end user without requiring specialised technical knowledge. Where a non-disruptive update mechanism is provided, confirm that it is enabled by default and applies to automatic cybersecurity updates as described in vendor documentation.

  - **Functional Testing:** Where a disable mechanism is provided, activate it and verify that automatic cybersecurity updates are disabled following activation. Verify that the mechanism is accessible and operable by the end user without requiring specialised technical knowledge. Where a non-disruptive update mechanism is provided, trigger or observe an automatic update cycle and verify that the update is applied without causing a negative impact on the operating environment (e.g. without requiring a reboot, without interrupting active services, or without causing observable degradation of product functions during the update). Where triggering an update cycle is not technically feasible in the test environment, the assessor shall verify the non-disruptive behaviour through product logs, vendor-provided evidence, or equivalent documentation, and shall document the reason direct observation was not possible.

- **Runtime Observation:** Where a non-disruptive update mechanism is provided, observe the product's behaviour during normal operation to confirm that automatic update cycles behave consistently with the documented non-disruptive behaviour.

- **Assignment of Verdict:**

  The assessment is passed if all of the following occur:

  - The product supports automatic cybersecurity updates and implements at least one of the two required mechanisms, as confirmed through the assessment activities.
  - Where a disable mechanism is provided, the product disables automatic cybersecurity updates upon activation and is accessible to the end user without requiring specialised technical knowledge.
  - Where a non-disruptive update mechanism is provided, the product applies automatic updates without causing negative impact on the operating environment.
  - The product provides documentation that accurately describes the implemented mechanism and its behaviour.

  The assessment is **failed if any of the following occur:**

  - The product supports automatic cybersecurity updates but implements neither a disable mechanism nor a non-disruptive update mechanism.
  - Where implemented, the disable mechanism does not disable automatic cybersecurity updates upon activation.
  - Where implemented, the disable mechanism requires specialised technical knowledge or is not accessible to the user.
  - Where implemented, the non-disruptive update mechanism causes observable negative impact on the operating environment during update installation, inconsistent with the vendor's documentation.
  - The implemented mechanism does not operate as described in vendor documentation or does not meet the applicable requirement conditions.
  - Documentation does not describe the implemented mechanism or contradicts the observed product behaviour.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation confirming whether the product supports automatic cybersecurity updates and describing the implemented mechanism.
  - Where the product does not support automatic updates, vendor documentation confirming this, retained as the basis for the not applicable outcome.
  - Configuration inspection outputs confirming that the implemented mechanism is active and accessible in the default operational configuration.
  - Test reports documenting the steps performed to verify the disable mechanism or the non-disruptive update behaviour, and the results obtained.
  - Evidence demonstrating the activation and effect of the implemented mechanism, whether a disable mechanism or a non-disruptive update mechanism, confirming that it operates as required.
  - Logs or audit records capturing update process behaviour and any observed effects on the operating environment, or where direct observation was not feasible, vendor-provided evidence provided in its place together with a documented justification.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.5.5 REQ-SU-05

- **Assessment reference: REQ-SU-05**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product implements version control mechanisms to prevent unauthorised installation of cybersecurity updates or cybersecurity data that are older than the currently installed version, and that such mechanisms permit controlled rollback only where the rollback is performed to a previously verified and trusted version. This includes confirming that:

  - The product maintains version awareness of its currently installed cybersecurity updates and cybersecurity data.
  - The product rejects attempts to install cybersecurity updates or cybersecurity data that are older than the currently installed version, unless a controlled rollback condition is met.
  - Where rollback is supported, it is restricted to previously verified and trusted versions and is performed by the product under controlled conditions, including enforcement of defined access controls and integrity verification of the rollback target.
  - Version control mechanisms are enforced and cannot be bypassed through normal product operation or accessible configuration settings.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, with the ability to observe the product's response to update installation attempts at the available observation boundary.
  - **Preconditions:** The product is installed and running in its default operational configuration with a known current version of cybersecurity updates and cybersecurity data. Test update packages or data items representing versions older than the currently installed version are prepared prior to testing. Where the product supports controlled rollback, vendor documentation describing the rollback mechanism and the conditions under which it is permitted is available prior to testing.
  - **Required Tools:** Mechanisms capable of presenting older versioned update packages or cybersecurity data to the product for installation. logging or audit mechanisms to capture the product's response to installation attempts; mechanisms capable of observing the update installation process at the available observation boundary.
  - **References:** Vendor documentation describing the version control mechanisms implemented, including how version comparison is performed and how rollback is controlled. Vendor documentation describing the conditions and process for controlled rollback, where supported.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to confirm that the product implements version control mechanisms for cybersecurity updates and cybersecurity data. Verify that the documentation describes how the product determines whether an incoming update or data item is older than the currently installed version, and how it handles such cases. Where rollback is supported, verify that the documentation describes the conditions under which rollback is permitted and how the product ensures that rollback targets are previously verified and trusted versions.
  - **Configuration Inspection:** Inspect the product's configuration to verify that version control mechanisms are active and cannot be bypassed through default or readily accessible configuration settings. Where rollback is supported, confirm that the rollback mechanism is subject to access controls that prevent unauthorised initiation.

- **Functional Testing:** Attempt to install cybersecurity updates or cybersecurity data representing versions older than the currently installed version and verify that the product rejects such attempts. Where rollback is supported by the product, verify that rollback can only be performed to previously verified and trusted versions and that the rollback process is controlled by the product. Additionally attempt to initiate rollback to a version that is not identified as previously verified and trusted, and verify that the product prevents such rollback. Verify that rollback cannot be initiated without appropriate authorisation as described in vendor documentation.

- **Runtime Observation:** During normal product operation, observe the update process at the available observation boundary to confirm that version control checks are consistently applied during routine update or data ingestion cycles and are not bypassed during normal operation.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Demonstrably rejects attempts to install cybersecurity updates or cybersecurity data older than the currently installed version, as confirmed through functional testing.

  - Where rollback is supported, restricts rollback to previously verified and trusted versions and performs it under controlled conditions.

  - Maintains version control mechanisms that cannot be bypassed through normal product operation or accessible configuration settings.

  - Provides documentation that accurately describes the version control and rollback mechanisms implemented.

  The assessment is **failed if any of the following occur**:

  - The product accepts and installs cybersecurity updates or cybersecurity data older than the currently installed version without a controlled rollback condition being met.

  - Where rollback is supported, rollback can be performed to versions that are not previously verified and trusted, or can be initiated without appropriate authorisation.

  - Version control mechanisms can be bypassed through normal product operation or accessible configuration settings.

  - Documentation does not describe the version control mechanisms or contradicts the observed product behaviour.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation describing the version control mechanisms and, where applicable, the controlled rollback mechanism and its conditions.

  - Test reports documenting the older versioned update packages or data items used, the steps performed, and the product's observed response to each installation attempt.

  - Logs or audit records produced by the product confirming rejection of older versioned updates or data, and where applicable, confirming that rollback was performed only to previously verified and trusted versions.

  - Evidence demonstrating enforcement of rollback restrictions, including logs or test reports confirming failed rollback attempts to versions not identified as previously verified and trusted, where such testing was performed.

  - Configuration inspection outputs confirming that version control mechanisms are active and not bypassable through accessible configuration settings.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.5.6 REQ-SU-06

- **Assessment reference: REQ-SU-06**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product ensures that, in case of a failed or interrupted update, no partially applied or unverified update remains present or active and the product reverts to a previously verified and trusted state. This includes confirming that:

  - The product detects failed or interrupted update conditions.
  - Upon detection of a failed or interrupted update, the product removes or deactivates any partially applied or unverified update content.
  - The product reverts to a previously verified and trusted state following a failed or interrupted update.
  - The reversion mechanism operates reliably across the product's supported deployment configurations.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, with a known current verified and trusted state established prior to testing. The environment shall include mechanisms to observe the product's behaviour and state at the available observation boundary during and after update operations.
  - **Preconditions:** The product is installed and running in its default operational configuration. A baseline of the product's current verified and trusted state is documented prior to testing, including the version of installed cybersecurity updates and cybersecurity data. Mechanisms to deliberately interrupt or cause failure of an update process are identified and prepared prior to testing, appropriate to the product's update architecture. Vendor documentation describing the product's update failure handling and reversion behaviour is available prior to testing.
  - **Required Tools:** Mechanisms capable of triggering or simulating an update process appropriate to the product's deployment model; mechanisms capable of deliberately interrupting or causing failure of an update process at various stages, where technically feasible; logging or audit mechanisms to capture the product's state and behaviour before, during, and after the interrupted or failed update.
  - **References:** Vendor documentation describing the update failure handling mechanism and the reversion process. Vendor documentation describing what constitutes a previously verified and trusted state for the product.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to confirm that the product is designed to detect failed or interrupted update conditions and to revert to a previously verified and trusted state in such cases. Verify that the documentation describes how the product identifies a failed or interrupted update, how it removes or deactivates partially applied or unverified update content, and what constitutes the previously verified and trusted state to which the product reverts.
  - **Configuration Inspection:** Inspect the product's configuration to verify that update failure handling and reversion mechanisms are active in the default operational configuration and cannot be disabled through accessible configuration settings.
  - **Functional Testing:** Initiate an update process and deliberately interrupt or cause it to fail at representative stages of the update process, using mechanisms appropriate to the product's deployment model. Ensure that interruption or failure scenarios include cases occurring before update validation completes, during application, and after partial application, where applicable to the product's update architecture. After each interruption or failure, verify that: (i) no partially applied or unverified update content remains present or active in the product, and (ii) the product has reverted to the previously documented verified and trusted baseline state established prior to testing. Where deliberately interrupting an update is not technically feasible given the product's deployment model, the assessor shall verify the reversion behaviour through vendor-provided evidence, product logs from prior update failure events, or equivalent documentation, and shall document the reason direct testing was not performed.

- **Runtime Observation:** Observe the product's behaviour following a failed or interrupted update at the available observation boundary to confirm that the reversion process completes fully and that no residual partially applied or unverified update content remains present or active after reversion.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Demonstrably detects failed or interrupted update conditions and initiates reversion, as confirmed through the assessment activities.
  - Ensures that no partially applied or unverified update content remains present or active following a failed or interrupted update.
  - Reverts to a previously verified and trusted state following a failed or interrupted update.
  - Provides documentation that accurately describes the update failure handling and reversion mechanism.

  The assessment is **failed if any of the following occur**:

  - Partially applied or unverified update content remains present or active following a failed or interrupted update.
  - The product does not revert to a previously verified and trusted state following a failed or interrupted update.
  - The product reverts following a failed or interrupted update, but the resulting state is not previously verified and trusted.
  - The reversion mechanism can be disabled through accessible configuration settings.
  - Documentation does not describe the update failure handling and reversion mechanism, or contradicts the observed product behaviour.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation describing the update failure handling mechanism and the reversion process, including the definition of the previously verified and trusted state.
  - Test reports documenting the update interruption or failure scenarios used, the stages at which interruption was performed, and the product's observed state and behaviour after each.
  - Evidence demonstrating that no residual partially applied or unverified update content remains present or active in the product after reversion.
  - Logs or audit records capturing the product's state before, during, and after the failed or interrupted update, confirming reversion to the previously verified and trusted state, or where direct testing was not feasible, vendor-provided evidence or prior update failure logs provided in its place together with a documented justification.
  - Configuration inspection outputs confirming that update failure handling and reversion mechanisms are active in the default operational configuration.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.5.7 REQ-SU-07

- **Assessment reference: REQ-SU-07**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product and its associated update ecosystem implement robust safeguards against faulty update deployment, ensuring that:

  - Updates are introduced in a controlled and progressive manner (staged/phased deployment).
  - Update correctness and product functional integrity are validated prior to full-scale rollout.
  - The product can autonomously recover from faulty updates through rollback mechanisms.
  - The update infrastructure can detect anomalous behaviours and suspend propagation to prevent systemic impact.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, with a known current verified and trusted state established prior to testing, endpoint agents (clients) and update distribution infrastructure (e.g. cloud update servers, local mirrors, or management consoles). Representative environment including multiple endpoints to simulate phased rollout (e.g. test group, pilot group, production group). Controlled laboratory environment permitting the deployment of intentionally faulty updates. Network setup enabling monitoring of update propagation and control signalling between infrastructure and endpoints.
  - **Preconditions:** The product is installed and running in its default operational configuration. A baseline of the product's current verified and trusted state is documented prior to testing, including the version of installed cybersecurity updates and cybersecurity data, and connected to the update infrastructure. Update mechanisms enabled and configured according to vendor guidance. Access to update policies and rollout configuration settings, update logs and telemetry and incident detection or anomaly monitoring modules. Availability of test updates (including intentionally faulty or degraded updates where feasible in a controlled environment).
  - **Required Tools:** Network monitoring and traffic analysis tools. System monitoring tools (e.g. CPU, memory, process health, detection engine behaviour). Log collection and analysis tools. Test harness or sandbox environment for controlled update deployment scenarios. Fault injection tools (where feasible) to simulate degraded or faulty updates.
  - **References:** Vendor documentation describing the update deployment model (e.g. staged rollout, canary releases, etc.), validation mechanisms (e.g. health checks, integrity validation, functional testing), rollback procedures and update propagation control mechanisms. Product operational and security guides related to update management.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to confirm that staged or phased deployment mechanisms are defined and configurable, update validation procedures are specified prior to mass rollout, rollback mechanisms (local/on-device) are described and update suspension or "kill switch" capabilities exist within the update infrastructure. Verify that roles, thresholds, and decision criteria for anomaly detection are clearly defined.
  - **Functional Testing - Staged/Phased Deployment:** Configure groups staged deployment (e.g. test to pilot to production). Deploy an update and observe whether initial rollout is limited to a subset of endpoints and whether promotion to wider deployment requires successful validation.
  - **Functional Testing - Validation of Update Application:** Deploy an update to a controlled subset of systems. Observe and verify that the product performs integrity verification (e.g. signature verification, checksum validation) and functional checks (e.g. scanning engine operation, detection capability, absence of abnormal resource usage). Confirm that rollout to broader groups is conditional upon successful validation outcomes.

- **Functional Testing - Automated Rollback (Endpoint-Level):** Introduce or simulate a faulty update (e.g. degraded detection engine, resource exhaustion, false positives). Monitor endpoint behaviour to verify the detection of abnormal behaviour (e.g. crashes, excessive CPU usage, functional failure) and the automatic rollback to a previously stable version without manual intervention. Confirm that rollback restores normal operation.
- **Functional Testing - Automated Suspension of Update Propagation:** Trigger abnormal behaviour at scale or simulate anomaly signals reaching the update infrastructure. Verify that the update propagation is halted or suspended centrally and no additional endpoints receive the faulty update after suspension. Confirm existence of coordinated control (e.g. central policy enforcement, cloud-based kill switch).
- **Configuration and Integrity Inspection:** Inspect configuration settings to ensure that rollout policies are enforceable and not bypassable, rollback thresholds and triggers are defined and monitoring and telemetry mechanisms actively collect update health data.
- **Runtime Observation:** Collect logs and telemetry to confirm detection of anomalies is logged, rollback events are recorded and propagation suspension decisions are traceable and auditable.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur:**

  - Staged or phased deployment is implemented and demonstrably enforced.
  - Updates are validated for integrity and functional correctness prior to broader rollout.
  - The product automatically performs rollback upon detection of abnormal behaviour or degradation on endpoints.
  - The update infrastructure can automatically suspend update propagation in response to detected anomalies.
  - Detection, rollback, and suspension mechanisms operate without manual intervention and within acceptable timeframes to limit the UC impact.

  The assessment is **failed if any of the following occur:**

  - Updates are deployed simultaneously without staging or progressive rollout.
  - No effective validation mechanism exists prior to broad deployment.
  - Faulty updates cause persistent malfunction without automatic rollback.
  - The update infrastructure lacks the ability to suspend propagation of faulty updates.
  - Detection of abnormal behaviour is absent, unreliable, or delayed to the extent that widespread impact occurs.
  - Critical safeguards can be bypassed or disabled without appropriate control.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test reports describing staged deployment scenarios and outcomes.
  - Logs demonstrating update validation steps, detection of abnormal behaviour, rollback execution and suspension of update propagation.
  - Screenshots or captures showing rollout stages, policy enforcement and system recovery after rollback.
  - Configuration files and policy settings governing update behaviour.
  - Vendor documentation describing update control and resilience mechanisms.
  - Telemetry or monitoring outputs verifying anomaly detection and system response.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.5.8 REQ-SU-08

- **Assessment reference: REQ-SU-08**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product maintains operational recoverability and update capability when a false positive is triggered by a legitimate OS update, confirming that:

  - False positive detections do not render the product in an unusable or locked state.
  - The product continues to allow reception and application of signature or rule updates.
  - The update mechanism remains functional and is not blocked by the false positive condition.
  - The system can be restored to a correct operational state through signature or rule updates without requiring unsafe workarounds.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment of supported OS platforms (e.g. Windows, Linux, macOS), with a known current verified and trusted state established prior to testing. Controlled update infrastructure or test update server capable of delivering signature/rule updates. Capability to deploy OS updates (real or simulated) known to trigger false positives in a controlled environment.
  - **Preconditions:** The product is installed and running in its default operational configuration. A baseline of the product's current verified and trusted state is documented prior to testing. Update mechanisms enabled and properly connected to the update infrastructure. Availability of logs and alerting interfaces and administrative access to modify policies and observe system state. A test scenario prepared where an OS update (or simulated artefact) will trigger a false positive detection.
  - **Required Tools:** System and process monitoring tools (e.g. OS logs, performance monitors). Update delivery and management tools (local mirror, test update server). Traffic analysis tools to observe update communication. Test artefacts or simulation tools to reproduce false positive behaviour.
  - **References:** Vendor documentation describing false positive handling mechanisms, quarantine and remediation logic and update system operation under degraded conditions. Product operational manuals and update procedures.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to confirm that mechanisms exist to handle false positives triggered by trusted system components (e.g. OS files), update mechanisms are designed to remain operational during detection events and there are safeguards preventing critical system components from being irreversibly blocked.
  - **Functional Testing - False Positive Scenario:** Deploy or simulate an OS update that triggers a false positive detection in the product. Observe the product behaviour and verify detection and classification of the OS component as malicious (i.e. false positive condition) and actions taken by the product (e.g. quarantine, blocking and/or alerting).
  - **System State Verification:** Assess whether the system remains in a recoverable and functional state by verifying that the OS continues to operate sufficiently to allow product processes to run and confirming that user or administrative access is not irreversibly blocked.
  - **Update Mechanism Verification:** Attempt to deliver an updated signature database or rule set that resolves the false positive. Verify that the product can connect to update sources despite the false positive condition, the update, download and installation processes execute successfully and no blocking rules or self-protection mechanisms prevent update application.

- **Recovery Validation:** After applying the update, verify that the false positive is no longer triggered, any incorrectly quarantined or blocked OS components are restored or allowed to function and the system returns to normal operational state without manual intervention beyond update application.
- **Configuration and protection Inspection:** Inspect configuration settings to confirm that self-protection features do not inadvertently block update mechanisms, fail-safe or exception mechanisms exist for critical system components and update channels have sufficient priority or protection to operate during degraded states.
- **Runtime Observation:** Collect and analyse logs to confirm the detection event and classification of the false positive, the continued operation of update services during the event and the application of corrective update and subsequent resolution.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur:**

  - The product detects the false positive without rendering the system irrecoverable.
  - The update mechanism remains operational during the false positive condition.
  - A signature or rule update can be successfully applied to resolve the issue.
  - The system returns to normal operation after update application.
  - No unsafe or manual bypass (e.g. disabling protection entirely) is required to recover functionality.

  The assessment is **failed if any of the following occur:**

  - The false positive results in a system state that prevents update delivery or application.
  - The product blocks its own update mechanism.
  - Recovery requires disabling core security protections or manual intervention outside defined procedures.
  - The system remains in a degraded or unusable state after applying updates.
  - Critical OS functionality is irreversibly disrupted by the false positive.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test reports documenting the false positive scenario and recovery process.
  - Logs demonstrating the detection of the false positive, the continued operation of update mechanisms and the successful update application and resolution.
  - Screenshots or console outputs showing the system state before and after update and the alerts and recovery actions
  - Configuration files demonstrating update and protection settings.
  - Vendor documentation describing resilience to false positives.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.5.9 REQ-SU-09

- **Assessment reference: REQ-SU-09**
- **Assessment Objective:**

  Verify that the vendor identifies and manages risks from undocumented/deprecated OS interfaces so essential product functionality remains controlled when OS behaviour changes and that the vendor identifies and manages risks from undocumented/deprecated OS interfaces so the usage of said interfaces does not increase the OS attack surface and where possible, does not restrict usage of those very interfaces by other security related software. The assessment confirms that:

  - Relevant interface-related risks are identified and evaluated;
  - Mitigations are defined and maintained;
  - Compatibility is proactively validated against upcoming OS versions (including pre-release programs where available);
  - Detected issues are handled through corrective software updates and/or controlled feature limitation;
  - The vendor maintains sufficient traceability of tested product/OS versions and outcomes.

- **Assessment Preparation:**

  - **Test Environment:** Representative supported OS environments, including available pre-release OS versions used in the vendor's compatibility process.
  - **Preconditions:** Product operational, relevant technical documentation available, and records for risk handling and compatibility testing available.
  - **Required Tools:** Appropriate analysis and test tools as used by the vendor.
  - **References:** Applicable OS documentation/lifecycle information and vendor internal process documentation.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor records for identification of undocumented/deprecated interface dependencies and associated risks. Verify that the documentation explicitly states that mitigations exist and are applied when needed. Review compatibility testing evidence for upcoming OS releases, including tested OS/product versions and outcomes.
  - **Runtime Behaviour Analysis:** Verify that issues found are resolved via product updates or controlled feature deactivation/limitation. Confirm ongoing monitoring and reassessment as OS versions evolve.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Risks from undocumented/deprecated interfaces are managed with documented mitigations;
  - Compatibility testing for upcoming OS releases is performed and traceable;
  - Identified issues are addressed in a controlled and timely manner;
  - Essential functionality is not left with unmanaged risk, proven by a test of essential functionality of the product for tested OS versions.

  The assessment is **failed if any of the following occur**:

  - Such risks are not identified or not mitigated;
  - Compatibility testing evidence is missing or not traceable;

- Known issues are not handled appropriately;

NOTE 1: An absence of known issues is not an indication of failure by itself, if appropriate testing can be shown.

- Unmanaged loss/degradation of essential functionality is evident.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Risk and mitigation records;
  - Compatibility test records (OS/product version traceability and results);
  - Issue/remediation records (including updates or controlled feature limitation decisions);
  - Final assessment report with rationale for verdict.

NOTE 2: Confidentiality / NDA - Evidence may be provided in redacted, abstracted, or controlled-access form. Redaction, abstraction and access-control shall only be used for the parts of the evidence where the NDA with the OS vendor strictly requires such action.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.6 Authentication and access control

6.6.1 REQ-AAC-01

- **Assessment reference: REQ-AAC-01**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product implements effective tamper-protection and self-protection mechanisms. This includes confirming that:

  - The product prevents or significantly mitigates unauthorised attempts to, for example, disable protection functions, modify security-critical configuration, terminate or interfere with security-relevant processes or services, alter or delete detection databases, engines, or policy data.
  - The product maintains the integrity and availability of its security functions even when executed in a potentially hostile environment, including environments where malware may have local user-level or elevated privileges.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, with administrative functions accessible for testing. Supported operating systems and execution environments, including typical privilege separation models (e.g. user, administrator, system/root). The environment shall include mechanisms to observe authentication behaviour and administrative operation attempts at the available observation boundary.
  - **Preconditions:** The product is installed and running in its default operational configuration, including tamper/self-protection mechanisms. The system is not already compromised by unrelated malware that could invalidate test results. Vendor documentation describing the authentication mechanisms implemented for administrative operations is available prior to testing. Test accounts or credentials with and without administrative privileges are established prior to testing where the product and environment permit. Logs, alerts, and audit capabilities are enabled and accessible.
  - **Required Tools:** Mechanisms capable of attempting administrative operations with and without prior authentication; logging or audit mechanisms to capture authentication events and administrative operation attempts and their outcomes.

- **Reference Documentation:** Vendor documentation describing tamper-protection and self-protection features, supported configuration options and limitations. Product security architecture or threat model documentation. Installation and administration guides defining expected protection boundaries.

- **Assessment Activities:**

  - **Documentation Review:** Verify that the vendor documentation describes implemented tamper- and self-protection mechanisms. Confirm which tamper- and self-protected components are identified (e.g. processes, services, files, configuration, databases). Identify any documented exceptions, platform-specific limitations, or dependency on operating system protections.

  - **Configuration Inspection:** Inspect product configuration to confirm that disabling or weakening the components identified in the "Documentation Review" requires explicit, authenticated administrative action. Verify, when possible, that protection applies to both runtime components (e.g. processes, services) and persistent artefacts (e.g. files, registry entries, databases). If this was not possible, justify the reason why it was not possible and which component.

  - **Functional Tampering Attempts:** Attempt actions without appropriate authorisation and observe product behaviour such as terminate or suspend security-critical processes or services, disable real-time protection or detection modules, modify or delete detection databases, configuration files, or policy settings and prevent the product from starting at system boot. Repeat selected attempts with different privilege levels (e.g. standard user, administrator) where applicable.

  - **Runtime Behaviour Observation:** Observe whether the product prevents the tampering attempt, restores protection (self-healing) and raises alerts, logs events, or notifies the user or management system. Verify that protection remains active and functional after failed tampering attempts or system restarts.

  - **Advanced Scenario Evaluation:** Where applicable, simulate more advanced interference attempts consistent with the expected threat model (e.g. malware attempting to inject code or unload drivers). Verify that critical security functions are resistant to such interference within the feasibility constraints of the platform.

- **Assignment of Verdict:**

  The assessment is passed if all of the following occur:

  - The product demonstrably implements tamper- and self-protection mechanisms appropriate to its intended deployment and threat environment.

  - Unauthorised attempts to disable, modify, or interfere with security-critical components are prevented or mitigated.

  - Any permitted administrative override requires authenticated and documented procedures.

  - The product maintains or restores its protective functionality following tampering attempts.

  - Evidence shows consistent enforcement of protection across applicable components and privilege levels.

  The assessment is **failed if any of the following occur:**

  - Security-critical components can be disabled, modified, or removed without proper authorisation.

  - Self-protection mechanisms are absent, trivially bypassed or they do not protect the product.

  - Tampering causes permanent loss or prolonged degradation of core protection functions.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting tampering scenarios executed, results and observed behaviour.

  - Logs, alerts, or audit records demonstrating detection or prevention of tampering attempts.

  - Screenshots, console outputs, or equivalent methods showing blocked or mitigated actions.

- Configuration extracts confirming enablement of tamper/self-protection.
- Vendor documentation describing the implemented protection mechanisms and their intended scope.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.6.2 REQ-AAC-02

- **Assessment reference: REQ-AAC-02**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product requires authentication for administrative operations, leveraging existing OS or platform authentication mechanisms, including configuration changes, protection deactivation, and policy modification. This includes confirming that:

  - The product leverages existing OS or platform authentication mechanisms for administrative authentication.
  - The product requires authentication before permitting any administrative operation, applied consistently across all administrative operations including configuration changes, protection deactivation, and policy modification.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, with administrative functions accessible for testing. The environment shall include mechanisms to observe authentication behaviour and administrative operation attempts at the available observation boundary.
  - **Preconditions:** The product is installed and running in its default operational configuration. The OS or platform authentication mechanisms available in the deployment environment are identified prior to testing. Vendor documentation describing the authentication mechanisms implemented for administrative operations is available prior to testing. Test accounts or credentials with and without administrative privileges are established prior to testing where the product and environment permit.
  - **Required Tools:** Mechanisms capable of attempting administrative operations with and without prior authentication; logging or audit mechanisms to capture authentication events and administrative operation attempts and their outcomes.
  - **References:** Vendor documentation describing the authentication mechanisms applied to administrative operations and how they leverage OS or platform mechanisms.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to confirm that the product requires authentication for administrative operations. Verify that the documentation identifies which OS or platform authentication mechanisms are leveraged. Verify that the documentation covers all categories of administrative operations listed in the requirement, including configuration changes, protection deactivation, and policy modification.
  - **Configuration Inspection:** Inspect the product's configuration to verify that authentication is required for administrative operations in the default operational configuration and cannot be disabled through accessible configuration settings. Confirm that the authentication mechanism in use aligns with what is described in vendor documentation.

- **Functional Testing:** Attempt to perform representative administrative operations, including configuration changes, protection deactivation, and policy modification, without prior authentication and verify that the product prevents such attempts. Attempt the same operations following successful authentication and verify that they are permitted. Verify that the product leverages OS or platform authentication mechanisms as described in vendor documentation. Where test accounts or credentials cannot be established due to product or environment constraints, the assessor shall verify authentication enforcement through available observation mechanisms and vendor-provided evidence, and shall document the reason direct testing was not performed.

- **Runtime Observation:** During normal product operation, observe authentication events at the available observation boundary to confirm that authentication is consistently required before administrative operations are permitted and that no administrative operation can be performed without prior successful authentication.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Requires authentication before permitting any administrative operation, as confirmed through the assessment activities.

  - Leverages OS or platform authentication mechanisms for administrative authentication.

  - Prevents administrative operations from being performed without prior successful authentication.

  - Provides documentation that accurately describes the authentication mechanisms applied to administrative operations.

  The assessment is **failed if any of the following occur**:

  - Administrative operations can be performed without prior successful authentication.

  - The product does not leverage OS or platform authentication mechanisms for administrative authentication.

  - Authentication can be bypassed or disabled through accessible configuration settings.

  - Documentation does not describe the authentication mechanisms applied to administrative operations, or contradicts the observed product behaviour.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation describing the authentication mechanisms applied to administrative operations and how OS or platform mechanisms are leveraged.

  - Test reports documenting the administrative operations attempted, the authentication states under which they were attempted, and the product's observed response to each, or where direct testing was not performed, vendor-provided evidence and observation outputs used in its place, together with a documented justification.

  - Logs or audit records capturing authentication events and administrative operation attempts, confirming that unauthenticated attempts were prevented.

  - Configuration inspection outputs confirming that authentication is required for administrative operations in the default operational configuration.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.6.3 REQ-AAC-03

- **Assessment reference: REQ-AAC-03**
- **Assessment Objective:**

  The objective of this assessment is to verify that, where technically feasible, the product supports and encourages MFA for administrative access and enables it by default. This includes confirming that:

  - Where MFA is supported, it is enabled by default.
  - Where MFA is not technically feasible, the product's documentation provides a clear justification for why MFA cannot be implemented in the given deployment context.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, with administrative access functions accessible for testing.
  - **Preconditions:** The product is installed and running in its default operational configuration. Vendor documentation describing the MFA implementation or, where MFA is not supported, the justification for its absence is available prior to testing.
  - **Required Tools:** Mechanisms capable of attempting administrative access with single-factor and multi-factor authentication; logging or audit mechanisms to capture authentication events and their outcomes.
  - **References:** Vendor documentation describing the MFA mechanism implemented and how it is enabled by default, or the technical justification for why MFA is not feasible.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to confirm whether MFA for administrative access is technically feasible in the product's deployment model. Where MFA is supported, verify that the documentation describes the MFA mechanism implemented and confirms that it is enabled by default. Where MFA is not supported, the determination of technical feasibility shall be based on documented characteristics of the product's architecture, deployment model, and execution environment. Verify that the justification identifies specific technical constraints related to the product's architecture, deployment model, or execution environment that prevent MFA implementation.
  - **Configuration Inspection:** Where MFA is supported, inspect the product's default configuration to verify that MFA is enabled for administrative access.
  - **Functional Testing:** Where MFA is supported, attempt administrative access using only a single authentication factor and verify that the product denies such access. Attempt administrative access providing all required authentication factors and verify that access is granted. Confirm that the MFA mechanism operates as described in vendor documentation.
  - **Runtime Observation:** Where MFA is supported, observe authentication events during normal product operation at the available observation boundary to confirm that MFA is consistently enforced for administrative access and is not bypassed during routine operation.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Supports MFA for administrative access and enables it by default, as confirmed through the assessment activities.
  - Denies administrative access when only a single authentication factor is provided.
  - Provides documentation that accurately describes the MFA mechanism and confirms it is enabled by default.

- Where MFA for administrative access is not technically feasible, the product provides a clear and specific technical justification in vendor documentation, identifying the specific technical constraints related to the product's architecture, deployment model, or execution environment that prevent MFA implementation.

The assessment is **failed if any of the following occur**:

- The product supports MFA but does not enable it by default.
- Administrative access is granted when only a single authentication factor is provided.
- The technical justification provided for not supporting MFA is insufficient, unclear, or not specific to a genuine technical constraint.
- Documentation does not describe the MFA mechanism or contradicts the observed product behaviour.

- **Supporting Evidence:**

The following evidence shall be collected and attached to the assessment report:

- Vendor documentation describing the MFA mechanism and confirming it is enabled by default, or where MFA is not supported, the technical justification for its absence.
- Configuration inspection outputs confirming that MFA is enabled by default in the default operational configuration, where applicable.
- Test reports documenting the authentication attempts performed, the factors provided, and the product's observed response to each.
- Where MFA is supported, logs or audit records capturing authentication events confirming that single-factor access attempts were denied and that MFA is consistently enforced.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.6.4 REQ-AAC-04

- **Assessment reference: REQ-AAC-04**
- **Assessment Objective:**

The objective of this assessment is to verify that the product supports role-based or equivalent privilege control mechanisms that restrict administrative access rights to the minimum privileges necessary for the assigned role. This includes confirming that:

- The product implements role-based or equivalent privilege control mechanisms for administrative access.
- Administrative access rights assigned to each role are restricted to the minimum privileges necessary for that role to perform its documented functions, as defined in vendor documentation.
- Privilege control mechanisms enforce defined role boundaries and modifications to role privileges are subject to appropriate access control.

- **Assessment Preparation:**

- **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, with administrative role and privilege management functions accessible for testing.
- **Preconditions:** The product is installed and running in its default operational configuration. Vendor documentation describing the role-based or equivalent privilege control mechanisms implemented, including the defined roles and their associated access rights, is available prior to testing. Test accounts or equivalent configurations representing different administrative roles are established prior to testing where the product and environment permit.

- **Required Tools:** Mechanisms capable of attempting operations outside the defined privileges of a given role; logging or audit mechanisms to capture access attempts and their outcomes.
- **References:** Vendor documentation describing the privilege control mechanisms implemented, the defined roles, and the access rights associated with each role.

- **Assessment Activities:**
  - **Documentation Review:** Review vendor documentation to confirm that the product implements role-based or equivalent privilege control mechanisms for administrative access. Verify that the documentation defines the available roles and specifies the access rights associated with each, confirming that access rights are restricted to the minimum privileges necessary for each role's documented functions as defined in vendor documentation. Verify that the documentation describes how privilege boundaries between roles are enforced by the product.
  - **Configuration Inspection:** Inspect the product's configuration to verify that privilege control mechanisms are active in the default operational configuration and that role-based access rights are enforced as described in vendor documentation. Confirm that privilege control mechanisms enforce the defined role boundaries and that modifications to role privileges are subject to appropriate access control and are reflected in updated role definitions.
  - **Functional Testing:** For representative administrative roles defined by the product, attempt to perform operations that fall outside the documented privileges of each role and verify that the product prevents such attempts. Verify that operations within the documented privileges of each role are permitted. Confirm that no role can be used to escalate privileges beyond those assigned to it as documented. Where test accounts or equivalent role configurations cannot be established due to product or environment constraints, the assessor shall verify privilege boundary enforcement through available observation mechanisms and vendor-provided evidence, and shall document the reason direct testing was not performed.
  - **Runtime Observation:** During normal product operation, observe access control enforcement at the available observation boundary to confirm that privilege boundaries between roles are consistently maintained and that no role can access functions or data beyond its documented minimum necessary privileges as defined in vendor documentation.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Implements role-based or equivalent privilege control mechanisms for administrative access, as confirmed through the assessment activities.
  - Restricts administrative access rights for each role to the minimum privileges necessary for that role's documented functions, as defined in vendor documentation.
  - Prevents roles from performing operations outside their documented privilege boundaries.
  - Provides documentation that accurately defines the available roles and their associated minimum necessary access rights.

  The assessment is **failed if any of the following occur**:

  - The product does not implement role-based or equivalent privilege control mechanisms for administrative access.
  - Administrative access rights assigned to one or more roles exceed the minimum privileges necessary for that role's documented functions as defined in vendor documentation.
  - A role can be used to perform operations outside its documented privilege boundaries.
  - Modifications to role privileges are not subject to appropriate access control or are not reflected in updated role definitions.
  - Documentation does not define the available roles and their associated access rights, or contradicts the observed product behaviour.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation defining the available roles and the access rights associated with each, confirming that access rights are restricted to the minimum privileges necessary for each role.
  - Test reports documenting the roles used, the operations attempted outside documented privilege boundaries, and the product's observed response to each, or where direct testing was not performed, vendor-provided evidence and observation outputs used in its place, together with a documented justification.
  - Logs or audit records capturing access attempts and their outcomes, confirming that operations outside documented role privileges were prevented.
  - Configuration inspection outputs confirming that privilege control mechanisms are active and that role-based access rights are enforced in the default operational configuration.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.6.5 REQ-AAC-05

- **Assessment reference: REQ-AAC-05**
- **Assessment Objective:**

  The objective of this assessment is to verify that, when a web-based management console is provided, the antivirus/antimalware product implements secure authentication token generation mechanisms for user or session authentication. This includes confirming that:

  - The product ensures authentication tokens are:
    - Unique per session and/or authentication context.
    - Non-predictable through the use of cryptographically secure random generation.
    - Resistant to brute-force and guessing attacks, including online and replay attacks.
  - The product properly protects tokens throughout their lifecycle (i.e. generation, transmission, storage, expiration, and revocation).

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including an operational instance of the product's web management console, deployed as local interface (e.g. HTTPS service on the protected system), and/or remote or cloud-based management interface. Supported web browsers or API clients as documented by the vendor. Where applicable, both human-interactive (browser-based) and API-based access paths shall be included.
  - **Preconditions:** The web management console is installed, reachable, and fully operational. At least one valid administrative user account is created. Authentication logging and audit mechanisms are enabled by default. Default or recommended security configuration is applied (including HTTPS enabled). The assessor has credentials for authorised access and the ability to observe authentication flows.
  - **Required Tools:** Web application testing tools (e.g. browser developer tools). HTTP(S) traffic inspection or capture tools capable of observing encrypted sessions (where and when legally and technically permitted). Security testing tools for token analysis. Authentication or web application vulnerability testing tools. Log inspection and correlation tools.
  - **Reference Documentation:** Vendor documentation describing authentication mechanisms of the web management console, token types used (e.g. session cookies, bearer tokens, API tokens) and token lifetime, renewal, and revocation behaviour. Product security architecture or design documentation addressing authentication and session management.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation identifies the use of authentication tokens for the web management console. Confirm that token security properties (e.g. randomness, lifetime, scope) are described. Identify any documented limitations or configuration options affecting token strength or behaviour.

  - **Token Generation and Uniqueness Analysis:** Perform multiple authentication events (e.g. repeated logins, parallel sessions) and collect the resulting tokens. Verify that tokens are unique per session or authentication context and that no token reuse occurs across different sessions or users. Check that tokens cannot be derived or predicted from previous tokens or observable parameters.

  - **Non-Predictability and Strength Evaluation:** Analyse collected tokens to confirm enough and strong length and entropy consistent with cryptographically secure random generation and the absence of predictable structures, counters, timestamps, or user-derived values. Verify that tokens are generated using cryptographically secure random number generators provided by the execution platform.

  - **Resistance to Brute-Force and Abuse:** Attempt controlled, rate-limited guessing or replay of authentication tokens and observe product behaviour. Verify that invalid or expired tokens are consistently rejected and that repeated authentication failures trigger appropriate protections (e.g. rate limiting, temporary blocking, logging). Confirm that tokens have defined lifetimes and are invalidated upon logout or privilege changes.

  - **Secure Handling and Transmission:** Observe runtime behaviour to confirm that tokens are transmitted only over encrypted channels (e.g. HTTPS/TLS) and they are not exposed in URLs, logs, or error messages. Where applicable (e.g. browser-based access), verify the use of protective attributes such as secure cookies and appropriate scope restrictions.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The web management console uses authentication tokens that are unique per session or authentication context and non-predictable and generated using cryptographically secure methods.

  - Tokens exhibit the strength to resist brute-force and guessing attacks within the expected threat model.

  - Tokens are properly protected in transmission and invalidated according to defined lifecycle events.

  - No evidence is found of token reuse, predictability, or weak protection.

  The assessment is **failed if any of the following occur**:

  - Authentication tokens are predictable, reused, or derived from insufficiently random data.

  - Token length or entropy is short or low to provide resistance to brute-force attacks.

  - Tokens can be guessed, replayed, or abused without detection or mitigation.

  - Tokens are exposed through insecure channels, URLs, logs, or error messages.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting authentication and token collection activities and observed token characteristics and validation results.

  - Captures or logs demonstrating secure token transmission and rejection of invalid tokens.

  - Screenshots or console outputs showing authentication flows and error handling.

  - Configuration extracts confirming token lifetime, protection, and security settings.

  - Vendor documentation describing token generation and session management mechanisms.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.6.6 REQ-AAC-06

- **Assessment reference: REQ-AAC-06**
- **Assessment Objective:**

  The objective of this assessment is to verify that, when a web-based management console is provided, the product implements authentication tokens with a finite and well-defined lifetime. This includes confirming that:

  - The product prevents the use of non-expiring or indefinitely valid authentication tokens, which could enable persistent unauthorised access if compromised.
  - The product enforces token expiration consistently and reliably across normal operation, abnormal termination, and privilege changes.
  - The assessment ensures that access to the web management console is time-bounded and that the risk associated with token leakage, replay, or long-term misuse is appropriately mitigated.
- **Assessment Preparation:**
  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including a deployed instance of the product's web management console, operating as a local web interface, and/or a remote or cloud-based management service. Supported web browsers and, where applicable, API clients. When necessary, a network environment allowing observation of authentication behaviour over encrypted channels (e.g. HTTPS).
  - **Preconditions:** The web management console is installed, reachable, and operational. At least one valid administrative user account is configured. Authentication, session, and audit logging are enabled. Default or recommended security settings are applied. A set of authorised access credentials and the ability to initiate and terminate sessions.
  - **Required Tools:** Web application testing tools and browser developer consoles. Session and token inspection or replay testing tools. When possible, time manipulation or simulation tools (e.g. to observe behaviour after elapsed time). Log analysis tools for authentication and session events.
  - **Reference Documentation:** Vendor documentation describing authentication and session handling for the web management console, token lifetime, timeout, and renewal policies. Product security architecture documentation relevant to session management.
- **Assessment Activities:**
  - **Documentation Review:** Verify that vendor documentation explicitly defines authentication token lifetimes (e.g. idle timeout, absolute expiration), conditions under which authentication tokens are invalidated (e.g. logout, expiry, password change). Confirm that no documentation indicates the existence of perpetual or non-expiring authentication tokens.
  - **Configuration Inspection:** Inspect product configuration to verify that authentication token expiration mechanisms are enabled by default or are mandatory, authentication token lifetimes are bounded by defined parameters. Identify whether the authentication token expiry behaviour is configurable and whether insecure configurations (e.g. infinite lifetime) are possible.
  - **Runtime Authentication Token Expiration Testing:** Authenticate to the web management console and obtain a valid authentication token. Observe and record the authentication token's validity over time by remaining idle beyond the documented inactivity timeout, and/or exceeding the documented maximum session lifetime. Verify that, once expired, the authentication token is rejected by the product and requires re-authentication to regain access.

- **Abuse and Edge-Case Evaluation:** Attempt to reuse authentication tokens after browser restarts or session interruptions. Verify that authentication tokens cannot be used indefinitely across extended periods or system restarts. Observe logs and alerts to confirm that expiration and rejection events are recorded.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur:**

  - Authentication tokens issued by the web management console have a finite, defined lifetime.
  - Authentication tokens are invalidated upon expiration and cannot be reused thereafter.
  - No configuration or implementation allows tokens to remain valid indefinitely.
  - Expiration behaviour is consistently enforced and observable during runtime testing.

  The assessment is **failed if any of the following occur:**

  - The product issues authentication tokens that do not expire.
  - Token expiration is optional and can be configured to "never expire".
  - Tokens remain valid and reusable after documented timeout periods, logout, or other invalidation events.
  - Token expiration behaviour is undocumented, inconsistent, or unreliable.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting authentication token issuance and expiration behaviour and results of timeout and reuse attempts.
  - Logs or audit records showing authentication token expiration, invalidation, and access denial events.
  - Screenshots or captures demonstrating loss of access after authentication token expiration.
  - Configuration extracts defining authentication token lifetime parameters.
  - Relevant vendor documentation describing session and authentication token lifecycle management.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.6.7 REQ-AAC-07

- **Assessment reference: REQ-AAC-07**

- **Assessment Objective:**

  The objective of this assessment is to verify that, when a web-based management console is provided, the product adequately protects authentication tokens throughout their lifecycle, including generation, transmission, usage, and logging. This includes confirming that the authentication tokens are:

  - Transmitted exclusively over channels providing confidentiality and integrity protection.
  - Appropriately protected according to the interface type (e.g. browser-based UI, API, remote console).
  - Not exposed through logs, audit events, URLs, error messages, or other observable outputs.

- **Assessment Preparation:**
  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including an operational instance of the product's web management console, deployed as a local management interface, and/or a remote or cloud-based management console. Supported web browsers and/or API clients as documented by the vendor. Network connectivity allowing observation of authentication flows over encrypted channels (e.g. HTTPS/TLS). Where applicable, both interactive (browser-based) and programmatic (API-based) interfaces shall be included.
  - **Preconditions:** The web management console is installed, reachable, and fully operational. At least one valid administrative user account exists. Authentication, audit, and application logging is enabled. Default or recommended security configuration is applied. The assessor has authorised credentials and the ability to generate authentication events.
  - **Required Tools:** Web application testing tools and browser developer consoles. Network traffic inspection tools capable of validating secure transport (without breaking encryption, unless permitted). Log and audit inspection tools. API testing tools (where the console exposes APIs). Error and exception handling observation tools.
  - **Reference Documentation:** Vendor documentation describing authentication mechanisms and token handling, supported interface types (e.g. UI, API) and logging and auditing behaviour. Product security architecture documentation addressing token protection.
- **Assessment Activities:**
  - **Documentation Review:** Verify that vendor documentation describes secure transport requirements for the web management console, authentication token handling and protection mechanisms and logging and auditing policies with respect to sensitive data. Confirm that no documentation suggests transmission or exposure of authentication tokens in plaintext or insecure contexts.
  - **Secure Transmission Verification:** Observe authentication and session establishment traffic to confirm that authentication tokens are transmitted only over secure channels providing confidentiality and integrity protection (e.g. TLS-protected HTTPS) and no authentication tokens are transmitted over unencrypted or downgraded channels. Attempt access via insecure protocols (e.g. HTTP) and verify that authentication is rejected or redirected to a secure channel.
  - **Interface-Specific Authentication Token Protection Evaluation:** Verify that authentication token protection mechanisms are appropriate to the interface type (e.g. secure cookie attributes for browser-based interfaces, header-based token handling for APIs or scope, path, or binding restrictions limiting authentication token reuse). Confirm that authentication tokens are not exposed unnecessarily to client-side scripts or components beyond their intended scope.
  - **Logging, Audit, and Output Inspection:** Inspect audit logs, system logs, application logs, and diagnostic outputs to verify that authentication tokens do not appear in full or in usable form and are not included in URLs, query parameters, or referrer data. Trigger authentication failures and error conditions and confirm that error messages do not disclose authentication tokens.
  - **Abuse and Leakage Scenarios:** Attempt controlled misuse scenarios (e.g. replaying captured requests or forcing error conditions during authentication) and verify that authentication tokens are not exposed during these scenarios and that appropriate logging and rejection behaviour is observed.
- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:
  - Authentication tokens are transmitted exclusively over secure channels providing confidentiality and integrity protection.
  - Authentication token protection mechanisms are appropriate for the interface type and limit exposure and misuse.
  - Authentication tokens do not appear in logs, audit events, URLs, error messages, or other observable outputs.
  - No evidence of authentication token leakage or insecure handling is identified during testing.

The assessment is **failed if any of the following occur**:

- Authentication tokens are transmitted over unencrypted or integrity-unprotected channels.
- Authentication token protection mechanisms are missing or inappropriate for the interface type.
- Authentication tokens appear in logs, audit records, URLs, error messages, or diagnostic outputs.
- Authentication tokens can be trivially exposed or harvested through normal or error-handling behaviour.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting secure transmission verification, interface-specific authentication token handling observations and logging and error-handling checks.
  - Network observations or tool outputs confirming encrypted transport.
  - Log and audit extracts demonstrating absence of authentication tokens.
  - Screenshots or captures showing authentication tokens flows and error messages.
  - Vendor documentation describing authentication token protection and logging practices.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.6.8 REQ-AAC-08

- **Assessment reference: REQ-AAC-08**
- **Assessment Objective:**

  The objective of this assessment is to verify that, when a web-based management console is provided, the product effectively protects against authentication token fixation attacks. This includes confirming that:

  - The product ensures new authentication tokens generation whenever the authentication status changes, including but not limited to:
    - Initial authentication after unauthenticated access.
    - Privilege elevation or role changes.
    - Re-authentication following token renewal.
  - The product prevents attackers from reusing pre-existing or attacker-controlled tokens to gain authenticated or elevated access.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including a deployed instance of the product's web management console, operating as a local web interface, and/or a remote or cloud-based management console. Supported browsers and/or API clients as documented by the vendor. A network environment supporting secure communication (e.g. HTTPS/TLS). The environment shall allow observation and comparison of authentication tokens issued before and after authentication state changes.
  - **Preconditions:** The web management console is installed, reachable, and fully operational. At least one valid administrative user account exists. Authentication and session logging is enabled. Default or recommended security configuration is applied. The assessor has authorised access credentials and the ability to observe or capture token values (where permitted).
  - **Required Tools:** Web application testing tools and browser developer consoles. Web (e.g. HTTP(S)) inspection or debugging tools suitable for observing token use. Session analysis and replay testing tools. Log and audit inspection tools.

- **Reference Documentation:** Vendor documentation describing authentication and session lifecycle behaviour and authentication token creation, renewal, and invalidation logic. Product security architecture or session management documentation.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation explicitly states that authentication tokens are regenerated upon authentication state changes and session fixation risks are addressed by design. Identify documented transitions that trigger token regeneration (e.g. login, privilege escalation).

  - **Pre-Authentication Token Observation:** Access the web management console in an unauthenticated state. Record any token or session identifier issued prior to authentication (e.g. anonymous session token, pre-login identifier). Confirm whether unauthenticated tokens are present and how they are handled.

  - **Authentication State Change Testing:** Authenticate using valid credentials and observe if a new authentication token is issued and the pre-authentication token is invalidated or replaced. Compare pre- and post-authentication tokens to confirm they are different and cannot be reused interchangeably.

  - **Privilege and Role Change Evaluation:** Where applicable, perform authentication state changes such as privilege elevation (e.g. user to administrator) and/or role changes via the management interface. Verify that each such change results in issuance of a new authentication token and invalidation of the previous authentication token.

  - **Authentication Token Reuse and Fixation Attempts:** Attempt to reuse pre-authentication or pre-privilege-change tokens to access authenticated or elevated resources. Confirm that such attempts are rejected and logged appropriately. Observe runtime behaviour to ensure that previously issued authentication tokens do not remain valid across authentication boundaries.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - New authentication tokens are generated whenever the authentication status changes.

  - Pre-authentication or lower-privilege tokens are invalidated and cannot be reused after authentication or privilege changes.

  - Authentication token regeneration behaviour is consistent, reliable, and observable across tested scenarios.

  - No evidence of authentication token fixation or reuse across authentication boundaries is identified.

  The assessment is **failed if any of the following occur**:

  - Authentication tokens remain unchanged across authentication status changes.

  - Pre-authentication or lower-privilege tokens remain valid after login or privilege escalation.

  - The product permits reuse of previously issued authentication tokens to gain authenticated or elevated access.

  - Authentication token regeneration behaviour is undocumented, inconsistent, or unreliable.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting pre- and post-authentication token values and authentication transition scenarios executed.

  - Captures or observations demonstrating authentication token regeneration and invalidation.

  - Logs or audit records showing authentication state changes and session handling.

  - Screenshots or tool outputs illustrating rejection of authentication token reuse attempts.

  - Relevant vendor documentation describing session and authentication token lifecycle management.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.6.9 REQ-AAC-09

- **Assessment reference: REQ-AAC-09**
- **Assessment Objective:**

  The objective of this assessment is to verify that, when a web-based management console is provided, the product enforces the PoLP. This includes confirming that:

  - The product implements authorisation mechanisms based on well-defined roles.
  - The product ensures that access to management functions, configuration settings, and security-relevant operations is restricted according to assigned roles.
  - The product prevents privilege escalation, including gaining access to functions not authorised for a user's role, elevating privileges without explicit administrative authorisation and bypassing role restrictions (e.g. through direct requests, parameter manipulation, or API misuse).

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including an operational instance of the product's web management console, deployed as a local administrative interface, and/or a remote or cloud-based management console. Supported web browsers and, where applicable, API clients. Network connectivity appropriate for normal administrative access. Where applicable, both user-interface and API-based management paths shall be included.
  - **Preconditions:** The web management console is installed, reachable, and fully operational. Multiple user accounts are configured, representing distinct roles (e.g. administrator, operator, read-only user). Roles and permissions are assigned according to vendor-recommended practices. Authentication, authorisation, and audit logging are enabled. The assessor has valid credentials for accounts with different privilege levels.
  - **Required Tools:** Web application testing tools and browser developer consoles. API testing tools (where management APIs are exposed). Request interception or manipulation tools (where permitted). Log and audit inspection tools.
  - **Reference Documentation:** Vendor documentation describing supported roles and permissions, authorisation enforcement mechanisms and administrative procedures for role assignment. Product security architecture or access control documentation.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation defines supported roles and their intended permissions and clearly distinguishes between privileges available to different roles. Confirm that security-critical operations (e.g. policy modification, update configuration, user management) are documented as restricted to appropriate roles.
  - **Role Configuration and Inspection:** Inspect the configuration of user roles and permissions to verify that roles are explicitly defined and enforced and default roles follow the PoLP. Confirm that role assignment or modification requires appropriate administrative authorisation.
  - **Authorisation Enforcement Testing:** Authenticate using each configured role and attempt to access authorised functions expected for that role and access unauthorised functions outside the role's scope. Verify that authorised actions succeed and unauthorised actions are consistently denied.
  - **Privilege Escalation Attempts:** Attempt controlled privilege escalation scenarios (e.g. directly invoking privileged URLs, endpoints, or API calls, modifying request parameters or identifiers and/or reusing tokens or sessions from lower-privilege roles). Confirm that such attempts do not result in access to higher-privilege functions and are appropriately rejected.

- **Runtime Observation and Logging:** Observe system behaviour during denied access attempts to verify that access control failures are logged and errors do not disclose sensitive authorisation logic. Verify that role enforcement remains active across session renewals, system restarts, and configuration changes.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product implements role-based access controls that clearly separate privileges by role.
  - Access to management functions is consistently enforced according to assigned roles.
  - Attempts to perform unauthorised actions or escalate privileges are prevented and rejected.
  - No interfaces, endpoints, or functions allow bypassing role-based restrictions.
  - Authorisation enforcement is reliable and observable across tested scenarios.

  The assessment is **failed if any of the following occur**:

  - Role-based access controls are missing, incomplete, or inconsistently enforced.
  - Users can access functions or data beyond what their assigned role permits.
  - Privilege escalation is possible through, for example, parameter manipulation, direct requests, or API access or equivalent procedures.
  - Authorisation enforcement relies solely on client-side controls or undocumented assumptions.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting role definitions tested and authorised and unauthorised access attempts and outcomes.
  - Logs or audit records showing enforcement of access control decisions.
  - Screenshots or console outputs demonstrating access denial for unauthorised roles.
  - Configuration extracts showing role and permission assignments.
  - Vendor documentation describing RBAC design and enforcement mechanisms.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.6.10 REQ-AAC-10

- **Assessment reference: REQ-AAC-10**

- **Assessment Objective:**

  The objective of this assessment is to verify that, when a web-based management console is provided, the product implements effective protection against brute force authentication attacks. This includes confirming that:

  - The product detects repeated failed authentication attempts and applies appropriate mitigation measures, such as progressive or exponential authentication delays and/or temporary or permanent account lockout;
  - The product reduces the risk of credential guessing, credential stuffing, or automated attack campaigns targeting administrative access to the management console.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including an operational instance of the product's web management console, deployed as a local administrative interface, and/or a remote or cloud-based management console. Supported web browsers and, where applicable, API clients. Network connectivity allowing repeated authentication attempts within realistic attack conditions.

  - **Preconditions:** The web management console is installed, reachable, and fully operational. At least one valid user account exists with known authentication credentials. Authentication failure logging and audit mechanisms are enabled. Default or vendor-recommended security configuration is applied. The assessor has authorisation to perform repeated authentication attempts for testing purposes.

  - **Required Tools:** Web application testing tools and browser developer consoles. Automated authentication attempt or scripting tools. Log and audit inspection tools. Timing and response analysis tools to observe delays or lockouts.

  - **Reference Documentation:** Vendor documentation describing authentication mechanisms and brute force protection features, defined thresholds for failed authentication attempts and lockout or delay behaviour and recovery procedures. Product security architecture documentation addressing authentication resilience.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation explicitly describes implemented brute force protection mechanisms and thresholds or conditions triggering delays or lockouts. Confirm that recovery or unlock procedures (e.g. automatic or administrative) are documented.

  - **Configuration Inspection:** Inspect authentication configuration to verify that brute force protection is enabled by default or strongly recommended and thresholds for failed authentication attempts are defined and bounded. Verify that insecure configurations (e.g. unlimited failed attempts without mitigation) are not enabled by default.

  - **Failed Authentication Simulation:** Perform a series of consecutive failed authentication attempts using invalid credentials. Observe and record product behaviour, including the introduction of progressive delays between attempts and/or temporary or permanent account lockout. Continue attempts until the documented or observed threshold is reached.

  - **Effectiveness and Enforcement Verification:** Verify that, once brute force protection is triggered, further authentication attempts are delayed or blocked and valid credentials cannot be used to bypass enforced delays or lockout without completing the defined recovery process. Confirm that protection applies consistently across different client types (e.g. browser vs API); and different network conditions or session contexts.

  - **Logging and Alerting Observation:** Inspect logs and audit records to verify that failed authentication attempts are recorded and the activation of brute force protection mechanisms is logged. Verify that error messages do not disclose sensitive details that could assist attackers (e.g. distinguishing valid usernames).

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product implements brute force protection mechanisms, such as progressive delay or account lockout.

  - Repeated failed authentication attempts reliably trigger the configured mitigation measures.

  - Brute force protection cannot be bypassed through repeated attempts, session manipulation, or interface misuse.

  - Brute force protection behaviour is consistently enforced and observable during testing.

The assessment is **failed if any of the following occur**:

- Brute force protection mechanisms are absent or not enabled.
- Unlimited or excessive failed authentication attempts are permitted without mitigation.
- Implemented delays or lockouts are trivially bypassed or they do not assist on reducing the brute force authentication attacks effects on the product.
- Brute force protection behaviour is undocumented, inconsistent, or unreliable.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting failed authentication attempt scenarios and the triggering and enforcement of delays or lockouts.
  - Logs or audit records showing failed login attempts and mitigation events.
  - Screenshots or console outputs demonstrating enforced delays or account lockout.
  - Configuration extracts defining brute force protection thresholds and behaviour.
  - Vendor documentation describing authentication resilience mechanisms.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.6.11 REQ-AAC-11

- **Assessment reference: REQ-AAC-11**
- **Assessment Objective:**

  The objective of this assessment is to verify that the antivirus/antimalware product does not use shared default credentials. This includes confirming that:

  - The product does not implement shared default credentials that are identical across product instances, installations, or deployments.
  - The product prevents the use of well-known, static, or vendor-defined default usernames and/or passwords that could be exploited for unauthorised access.
  - The product ensures that any required initial authentication credentials are unique per installation or instance.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more fresh installations or initialised instances of the product (e.g. endpoint protection deployments, server-based installations, network appliances or virtual appliances and/or cloud-managed components, where applicable). Supported operating systems and platforms as documented by the vendor. Access to any user interfaces or services requiring authentication (e.g. local UI, web management console, administrative CLI, APIs). Where feasible, multiple independent installations shall be assessed to identify shared credential behaviour.
  - **Preconditions:** The product is installed using standard installation or provisioning procedures. No manual credential changes have been performed unless explicitly required by the product during setup. All authentication-protected interfaces are enabled using default or recommended configuration. The assessor has authorised access to installation media, initial credentials (if provided), and documentation.
  - **Required Tools:** Installation and provisioning tools or scripts. Authentication testing tools for local, web, or API interfaces. Documentation review tools. Credential enumeration or comparison utilities (for controlled and authorised testing).

- **Reference Documentation:** Vendor documentation describing installation and initial setup procedures and any default accounts or initial credentials. Product security architecture or access control documentation.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to identify any mention of default usernames, passwords, keys, or credentials and requirements for credential change during or after installation. Confirm that no documentation promotes or permits long-term use of shared default credentials.

  - **Installation and Initialisation Inspection:** Perform a fresh installation or initialisation of the product. Identify any authentication credentials present immediately after installation, including administrative accounts, service accounts and API keys or access tokens. Verify whether credentials are unique per installation or the user is required to define or change credentials before first use.

  - **Cross-Instance Credential Comparison:** Where feasible, install the product on multiple systems or instances. Attempt to authenticate to each instance using the same credentials. Verify that identical credentials do not grant access across different installations or deployments.

  - **Authentication Enforcement Testing:** Attempt to access the product using commonly known or vendor-documented default credentials. Confirm that such credentials are rejected or they are valid only until explicitly replaced, for example, during initial setup. Verify that the product does not permit continued operation with shared default credentials.

  - **Runtime and Configuration Validation:** Inspect configuration files, account databases, or management interfaces to verify that credentials are not hardcoded or globally shared and any default accounts are either disabled or require credential change. Observe logs to confirm enforcement of credential configuration requirements.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product does not use shared default credentials across installations or instances.

  - Any initial credentials are unique per installation or need to be changed before operational use.

  - Authentication interfaces do not accept well-known, static, or vendor-common credentials.

  - No evidence of systemic credential reuse or default access is identified.

  The assessment is **failed if any of the following occur**:

  - The product uses shared default usernames, passwords, or credentials.

  - Default credentials remain valid after installation without mandatory change.

  - Identical credentials grant access to multiple product instances.

  - Credentials are hardcoded, documented as static defaults, or otherwise trivially exploitable.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting installation and initial authentication behaviour and credential comparison across instances.

  - Screenshots or console outputs showing credential enforcement or rejection.

  - Configuration extracts demonstrating absence of shared default credentials.

  - Logs or audit records showing authentication attempts during initial setup.

  - Vendor documentation describing credential handling and initialisation procedures.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.7 Confidentiality protection

6.7.1 REQ-CON-01

- **Assessment reference: REQ-CON-01**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product protects stored sensitive data by applying or facilitating encryption at rest, either directly by the product or by external systems, in accordance with state of the art cryptography as defined in Annex K and in a manner proportional to the capabilities of the execution environment. This includes confirming that:

  - The product identifies all categories of sensitive data it stores, including but not limited to authentication credentials, cryptographic private keys, sensitive configuration parameters, and quarantined content, and protects each at rest by applying or facilitating encryption either directly by the product or through external systems.
  - Encryption at rest is implemented in accordance with state of the art cryptography as defined in Annex K, proportional to the capabilities of the execution environment.
  - The product is configured in its default operational configuration to apply or rely on encryption at rest for all categories of sensitive data it stores.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, with access to storage locations where sensitive data is retained by the product, at the available observation boundary. For products where encryption is provided by external systems, assessment shall be performed through vendor-provided documentation, storage configuration evidence, or equivalent mechanisms confirming the encryption applied by those external systems.
  - **Preconditions:** The product is installed and running in its default operational configuration. Vendor documentation identifying the categories of sensitive data stored by the product, the storage locations used, the encryption mechanisms applied, and whether encryption is implemented by the product or by external systems is available prior to testing. Where applicable, access to stored data or storage inspection mechanisms is available to verify the application of encryption.
  - **Required Tools:** Storage inspection mechanisms capable of examining the format and protection state of data stored by the product at the available observation boundary; analysis mechanisms capable of verifying the cryptographic properties of encryption applied to stored data, where accessible; where encryption is provided by external systems, vendor-provided storage configuration documentation or cryptographic implementation documentation serving as equivalent evidence.
  - **References:** Vendor documentation identifying sensitive data categories, storage locations, and encryption mechanisms applied, including identification of whether encryption is product-implemented or externally provided. State of the art cryptography (Annex K).

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to confirm that the product identifies all categories of sensitive data it stores and applies or facilitates encryption at rest for each. Vendor documentation shall at minimum address the sensitive data categories explicitly listed in the requirement and shall identify any additional categories of sensitive data stored by the product beyond those listed. Verify that the documentation specifies the encryption mechanisms applied to each sensitive data category, identifies whether encryption is implemented by the product or by external systems, and confirms that the mechanisms are implemented in accordance with state of the art cryptography as defined in Annex K, proportional to the capabilities of the execution environment. Where external systems provide encryption, verify that the documentation identifies the specific external system relied upon and provides evidence that the external system applies encryption at rest in a manner aligned with Annex K requirements and appropriate to the relevant execution environment.

  - **Configuration Inspection:** Inspect the product's configuration to verify that the product is configured in its default operational configuration to apply or rely on encryption at rest for all categories of sensitive data it stores. Where encryption is provided by external systems, verify through available configuration evidence that the external system's encryption is active and applied to the relevant storage locations.

  - **Functional Testing:** For representative categories of sensitive data stored by the product, examine the stored data at the available observation boundary and verify that sensitive data is not stored in plaintext or in a reversibly encoded form without cryptographic protection. Verify that the cryptographic mechanisms applied correspond to those described in vendor documentation and are implemented in accordance with state of the art cryptography as defined in Annex K, proportional to the capabilities of the execution environment. Where direct examination of stored data is not feasible at the available observation boundary, the assessor shall verify storage protection through vendor-provided storage configuration documentation, cryptographic implementation documentation, or equivalent evidence, and shall document the reason direct examination was not performed.

  - **Runtime Observation:** During normal product operation, observe storage activity at the available observation boundary to confirm that sensitive data written to storage by the product is consistently protected and that no sensitive data is stored in an unencrypted form during routine operation. Where storage activity is not directly observable due to the involvement of external systems, verify through vendor-provided monitoring interfaces, logs, or equivalent evidence that storage protection remains consistently applied during routine operation.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product protects all categories of sensitive data it stores by applying or facilitating encryption at rest, either directly by the product or by external systems, in accordance with state of the art cryptography as defined in Annex K and proportional to the capabilities of the execution environment, as confirmed through the assessment activities.

  - The product applies or relies on encryption consistently across all storage locations where sensitive data is retained.

  - The product provides documentation that accurately identifies the sensitive data categories stored, the storage locations used, the encryption mechanisms applied, and whether encryption is implemented by the product or by external systems, with reference to state of the art cryptography as defined in Annex K.

  The assessment is **failed if any of the following occur**:

  - One or more categories of sensitive data are stored without encryption, or the available evidence does not demonstrate that the applied cryptographic mechanisms are implemented in accordance with state of the art cryptography as defined in Annex K.

  - Sensitive data is stored in plaintext or in a reversibly encoded form without cryptographic protection at any storage location used by the product.

  - The product is not configured in its default operational configuration to apply or rely on encryption at rest for one or more categories of sensitive data it stores.

- Where external systems are relied upon to provide encryption, vendor documentation does not identify the specific external system relied upon or does not provide evidence that the external system applies encryption at rest in a manner aligned with state of the art cryptography as defined in Annex K appropriate to the relevant execution environment.
- Documentation does not identify the sensitive data categories stored or the encryption mechanisms applied, or contradicts the observed product behaviour.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation identifying the sensitive data categories stored, the storage locations used, and the encryption mechanisms applied, including identification of whether encryption is product-implemented or externally provided, with reference to state of the art cryptography as defined in Annex K.
  - Storage inspection outputs or vendor-provided cryptographic implementation documentation confirming that sensitive data is protected by encryption at rest across storage locations where sensitive data is retained, and that the cryptographic mechanisms are implemented in accordance with state of the art cryptography as defined in Annex K; where encryption is provided by external systems, vendor-provided storage configuration documentation, external system cryptographic evidence, or equivalent artefacts shall be provided in place of direct storage inspection outputs.
  - Configuration inspection outputs confirming that the product is configured in its default operational configuration to apply or rely on encryption at rest for all categories of sensitive data it stores.
  - Test reports documenting the sensitive data categories examined, the inspection methods used, and the results obtained, or where direct examination was not feasible, vendor-provided storage configuration documentation, cryptographic implementation documentation, or equivalent evidence provided in its place, together with a documented justification.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.7.2 REQ-CON-02

- **Assessment reference: REQ-CON-02**

NOTE: The assessment is not applicable if the product does not store authentication credentials or cryptographic private keys, provided that vendor documentation confirms this and the assessor has documented the outcome accordingly. In this case, only the necessary Supporting Evidence actions need to be carried out to justify this outcome.

- **Assessment Objective:**

  The objective of this assessment is to verify that, where the product stores authentication credentials or cryptographic private keys, the product protects such data against unauthorised access using mechanisms available in the execution environment. This includes confirming that:

  - The product identifies whether it stores authentication credentials or cryptographic private keys.
  - Where such data is stored, the product applies protection mechanisms available in the execution environment, such as platform credential storage, hardware-backed key storage, administrative user rights, or equivalent protection.
  - The protection mechanisms prevent unauthorised access to stored authentication credentials and cryptographic private keys.
  - Protection mechanisms are active in the default operational configuration.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, with access to storage locations where authentication credentials or cryptographic private keys are retained by the product, at the available observation boundary.

- **Preconditions:** The product is installed and running in its default operational configuration. As a first step, confirm whether the product stores authentication credentials or cryptographic private keys. If the product does not store either, this assessment is not applicable and no verdict is assigned; the assessor shall document this outcome and retain vendor documentation confirming that such data is not stored. Where such data is stored, vendor documentation identifying the storage mechanisms applied and the execution environment protection mechanisms used is available prior to testing.
- **Required Tools:** Storage inspection mechanisms capable of examining the protection state of authentication credentials and cryptographic private keys stored by the product at the available observation boundary; mechanisms capable of attempting unauthorised access to stored credentials or private keys, where technically feasible given the protection mechanisms in use.
- **References:** Vendor documentation identifying the execution environment protection mechanisms applied to stored authentication credentials and cryptographic private keys.

- **Assessment Activities:**
  - **Documentation Review:** Review vendor documentation to confirm whether the product stores authentication credentials or cryptographic private keys. Where such data is stored, verify that the documentation identifies the execution environment protection mechanisms applied and describes how they prevent unauthorised access to the stored data. Verify that the protection mechanisms used are available in and appropriate to the product's execution environment. Where platform credential storage or hardware-backed key storage is used, verify that the documentation confirms these mechanisms are active and appropriately configured. Where administrative user rights are used as the protection mechanism, verify that the documentation describes how access to the stored data is restricted to users or processes operating with administrative privileges only and that no lower-privilege account or process can access the stored credentials or private keys. Where other equivalent mechanisms are used, verify that the documentation provides justification demonstrating that the mechanism prevents unauthorised access in a manner appropriate to the execution environment.
  - **Configuration Inspection:** Where authentication credentials or cryptographic private keys are stored, inspect the product's configuration to verify that the protection mechanisms are active in the default operational configuration and that the mechanisms used correspond to those available in the execution environment as described in vendor documentation.
  - **Functional Testing:** Where authentication credentials or cryptographic private keys are stored, attempt to access the stored data without appropriate authorisation using available mechanisms in the deployment environment and verify that the product's protection mechanisms prevent such access. Verify that the protection mechanisms operate as described in vendor documentation. Where direct attempts to access stored data are not technically feasible due to the nature of the protection mechanisms used, the assessor shall verify protection through vendor documentation, system interface outputs, or equivalent evidence, and shall document the reason direct testing was not performed.
  - **Runtime Observation:** Where authentication credentials or cryptographic private keys are stored, observe the product's behaviour during normal operation at the available observation boundary to confirm that protection mechanisms are consistently active and that stored credentials and private keys are not exposed in an unprotected form during routine operation.

- **Assignment of Verdict:**

  The assessment is not applicable if the product does not store authentication credentials or cryptographic private keys, provided that vendor documentation confirms this and the assessor has documented the outcome accordingly.

  The assessment is **passed if all of the following occur**:

  - Stores authentication credentials or cryptographic private keys and protects such data using mechanisms available in the execution environment, as confirmed through the assessment activities.
  - Demonstrates that protection mechanisms prevent unauthorised access to stored authentication credentials and cryptographic private keys.

- Provides documentation that accurately identifies the protection mechanisms applied, confirms they are available in and appropriate to the execution environment, and provides justification or configuration evidence, as applicable, demonstrating that the chosen mechanism prevents unauthorised access.

The assessment is **failed if any of the following occur**:

- Authentication credentials or cryptographic private keys are stored without protection mechanisms available in the execution environment.
- Stored authentication credentials or cryptographic private keys can be accessed without appropriate authorisation, or where direct testing was not feasible, vendor documentation or system interface outputs do not demonstrate that protection mechanisms prevent unauthorised access.
- Protection mechanisms are not active in the default operational configuration.
- Where platform credential storage or hardware-backed key storage is used, the documentation does not confirm these mechanisms are active and appropriately configured.
- Where administrative user rights are used, the documentation does not demonstrate that access to stored credentials or private keys is restricted to users or processes operating with administrative privileges only.
- Where other equivalent mechanisms are used, the vendor documentation does not provide justification demonstrating that the mechanism prevents unauthorised access in a manner appropriate to the execution environment.
- Documentation does not identify the protection mechanisms applied or contradicts the observed product behaviour.

- **Supporting Evidence:**

The following evidence shall be collected and attached to the assessment report:

- Vendor documentation identifying whether authentication credentials or cryptographic private keys are stored and, where applicable, the execution environment protection mechanisms applied, including configuration evidence or justification, as applicable, demonstrating that the chosen mechanism prevents unauthorised access.
- Where the product does not store such data, vendor documentation confirming this, retained as the basis for the not applicable outcome.
- Storage inspection outputs or equivalent evidence confirming that protection mechanisms are active and that stored credentials and private keys are not accessible without appropriate authorisation.
- Test reports documenting the unauthorised access attempts performed and the product's observed response, or where direct testing was not feasible, vendor documentation, system interface outputs, or equivalent evidence provided in its place together with a documented justification.
- Configuration inspection outputs confirming that protection mechanisms are active in the default operational configuration.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.7.3 REQ-CON-03

- **Assessment reference: REQ-CON-03**
- **Assessment Objective:**

  The objective of this assessment is to verify that, where the product supports local malware processing without an RDPS, the product ensures effective user control over telemetry mechanisms, ensuring that users can disable any data transmission that reveals the presence or existence of specific files or uploads files or file content (in part or in full) to the vendor or third parties, confirming that:

  - Telemetry features are clearly identified and configurable.
  - Users can disable such telemetry without impairing essential product functionality beyond expected limitations.
  - When disabled, no prohibited data is transmitted under normal operating conditions.
  - The setting is enforceable and not bypassed by background processes or fallback mechanisms.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment. Network monitoring setup capable of capturing outbound traffic (e.g. proxy, packet capture system). Controlled test environment allowing safe handling of test files (benign samples with known identifiers).
  - **Preconditions:** The product is installed and running in its default operational configuration. User or administrator access to telemetry and privacy settings. Availability of documented telemetry features (e.g. cloud scanning, reputation services, sample submission). Network connectivity enabled to allow telemetry flows during testing.
  - **Required Tools:** Network traffic analysis tools (e.g. packet capture, proxy inspection tools). Log analysis tools for reviewing telemetry events. Test file generation tools (to create identifiable files with known hashes or markers). System monitoring tools to observe background processes. Test environment capable of simulating user activity and system events.
  - **References:** Vendor documentation describing telemetry functionality and data types collected, configuration options for enabling/disabling telemetry and privacy and data handling policies. Product user guides and administrative configuration manuals. Privacy notices and data handling documentation.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to identify all telemetry features that may reveal file presence or involve file upload, to confirm that user-accessible controls exist to disable such features, to verify clarity and completeness of user instructions regarding telemetry control, the categories of transmitted information and distinctions between essential and non-essential telemetry.
  - **Configuration Inspection:** Inspect product interfaces and configuration settings to verify that telemetry controls are accessible to users or administrators, disabling options are clearly identifiable (i.e. granular enough to cover both file presence disclosure and file upload scenarios) and settings can be configured without requiring unsupported modifications. Verify whether telemetry controls persist after reboot or update and can be centrally managed where applicable.
  - **Functional Testing - Telemetry Enabled:** With telemetry enabled, introduce test files into the system. Observe outbound network traffic and logs to identify the transmission of file identifiers (e.g. hashes, metadata) and the transmission of file content or samples.
  - **Functional Testing - Telemetry Disabled:** Disable telemetry features related to file presence and upload. Repeat the test scenario done in the previous step by introducing the same test files and monitoring network traffic and logs. Verify that no file content is transmitted to external entities, no metadata enabling inference of file presence is transmitted and no hidden or fallback telemetry occurs.

- **Runtime Behaviour Analysis:** Monitor background processes and scheduled tasks to ensure disabled telemetry settings are not bypassed by automated processes and no delayed or queued transmissions occur after disabling telemetry.
- **Configuration Robustness Verification:** Restart the system and product services and confirm that telemetry settings remain disabled and no re-enablement occurs without user action. Inspect whether administrative policies (if applicable) can enforce telemetry disabling.
- **Functional Impact Verification:** Verify that disabling telemetry does not prevent essential security functionality from operating and does not create unsafe or degraded protection states beyond documented limitations.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product provides user-accessible controls to disable telemetry revealing file presence and file upload.
  - When disabled, no telemetry data is transmitted that could allow inference of file presence; or upload file content.
  - Telemetry settings are enforced consistently and persist across system states.
  - No hidden or indirect mechanisms bypass user-configured settings.
  - Disabling telemetry does not require disabling core security protections.

  The assessment is **failed if any of the following occur**:

  - The product does not provide a mechanism to disable relevant telemetry.
  - File presence information or file content is transmitted despite telemetry being disabled.
  - Telemetry controls are unclear, incomplete, or misleading.
  - Settings can be bypassed, reverted, or ignored by background processes, fallback mechanisms, or updates.
  - Disabling telemetry requires disabling essential security functionality.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Network traffic captures demonstrating telemetry behaviour with enabled and disabled settings.
  - Logs showing telemetry events and their suppression when disabled.
  - Screenshots or configuration exports showing telemetry settings.
  - Test reports documenting test scenarios, observed data transmission and verification steps
  - Vendor documentation describing telemetry mechanisms and controls.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.7.4 REQ-CON-04

- **Assessment reference: REQ-CON-04**
- **Assessment Objective:**

  The objective of this assessment is to verify that where the product depends on a RDPS for malware analysis and lacks equivalent local malware processing capabilities, the product documentation:

  - clearly describes the dependency on RDPS.
  - transparently informs users of the operational and cybersecurity implications of such dependency.
  - identifies the residual risks associated with absence of local malware processing.
  - includes the residual risk associated with TH5.4 as defined in clause B.1.1.6.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment with connectivity to the RDPS and the capability to simulate RDPS unavailability or communication failure. Access to all user-facing documentation delivered with the product.
  - **Preconditions:** The product is installed and running in its default operational configuration. Availability of product manuals, deployment and administration guides, security documentation and operational documentation related to malware processing architecture. Identification of whether local malware processing capabilities are present or absent.
  - **Required Tools:** Documentation review tools. Network monitoring tools for observing RDPS communication. Functional testing tools to simulate remote service unavailability. Logging and monitoring tools.
  - **References:** Vendor documentation describing malware processing architecture, RDPS dependencies, detection workflows, fallback behaviours and operational limitations. clause B.1.1.6 defining TH5.4.

- **Assessment Activities:**

  - **Documentation Review:** Review the product documentation to determine whether any reliance on RDPS for malware processing is explicitly stated, the absence of equivalent local malware processing is clearly identified, the role of RDPS within the detection architecture is adequately described. Verify that the documentation includes operational implications of RDPS dependency, the scenarios where malware detection capability may be degraded and the residual cybersecurity risks resulting from absence of local processing. Verify that the documentation explicitly references and describes the residual risk related to TH5.4 as defined in clause B.1.1.6 and any possible equivalents.
  - **Architecture and Functional Verification:** Inspect the product configuration and behaviour to confirm whether malware analysis functions depend primarily or exclusively on RDPS and equivalent local malware processing capabilities are absent or significantly limited. Observe product behaviour during normal RDPS operation and RDPS unavailability or communication interruption. Verify whether malware detection functionality becomes degraded or unavailable without RDPS connectivity and behaviour observed during testing is consistent with the documented residual risks.
  - **User Information Verification:** Assess whether the residual risk information is understandable for the intended audience, sufficiently prominent within the documentation and appropriately linked to deployment and operational guidance. Verify whether the documentation informs the user about connectivity requirements, potential impacts on malware detection and limitations under disconnected or degraded network conditions.
  - **Consistency Verification:** Cross-check documented behaviour against actual runtime behaviour and deployment characteristics. Verify that no contradictory or misleading statements exist regarding local malware processing capabilities, independence from RDPS and detection availability during connectivity loss.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The documentation clearly states that the product relies on RDPS for malware processing.
  - The absence of equivalent local malware processing is accurately described.
  - Residual risks associated with such dependency are documented, especially related to TH5.4 as defined in clause B.1.1.6.
  - The observed product behaviour is consistent with the documented dependency and residual risks.

  The assessment is **failed if any of the following occur**:

  - Dependency on RDPS is not documented.
  - The absence of local malware processing is omitted, unclear, or misleading.
  - Residual risks are not adequately described, especially related to TH5.4 as defined in clause B.1.1.6.
  - Actual product behaviour contradicts the documented claims.
  - Users cannot understand the operational implications of RDPS dependency.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Copies or extracts of relevant product documentation.
  - Test reports documenting RDPS dependency verification, connectivity interruption scenarios and observed operational behaviour.
  - Logs or captures demonstrating communication with RDPS.
  - Screenshots or outputs showing degraded behaviour during RDPS unavailability.
  - Assessment records mapping documented claims to observed functionality.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.8 Integrity protection

6.8.1 REQ-INT-01

- **Assessment reference: REQ-INT-01**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product enforces verification its own executables files and updates, ensuring their integrity based on the cryptographic requirements specified in Annex K. This includes confirming that:

  - Implements integrity verification mechanisms based on digital signatures for its executable files and updates.
  - Uses cryptographic requirements as specified in Annex K, taking into account the capabilities and constraints of the execution environment.
  - Prevents execution, installation, or application of any executable file or update that fails integrity verification.
  - Records integrity verification failures as cybersecurity-relevant events.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported product deployment models. Supported operating systems and execution environments. Update delivery mechanisms as implemented by the product (e.g. online update service, offline packages). Where applicable, environments with different capability profiles (e.g. full OS vs constrained environment) shall be included to assess proportional implementation. Prepare a set of test files (valid, unsigned, and tampered executables or components).

  - **Preconditions:** The product is correctly installed and operational. Integrity verification, update, and logging functionality is enabled using default or recommended configuration. Access to legitimate update packages and executables is available. The assessor can introduce controlled integrity failures (e.g. modified or invalidly signed files) in an authorised manner.

  - **Required Tools:** Cryptographic inspection and verification tools. File integrity modification or tampering utilities (for controlled testing). Log and audit inspection tools. Update simulation or staging tools. Binary inspection tools to analyse signature verification behaviour.

  - **References:** Vendor documentation describing integrity verification and secure update mechanisms and supported cryptographic algorithms and trust anchors. Product security architecture documentation. Annex K cryptographic requirements.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation describes the use of digital signatures for executables and updates and references cryptographic mechanisms aligned with Annex K. Confirm that integrity verification is mandatory and not optional for security-relevant components.

  - **Verification of Integrity Mechanisms:** Inspect product configuration and binaries to confirm that executable files and updates are subject to digital signature verification prior to use and the verification occurs before execution, installation, or application. Verify that trust anchors (e.g. public keys or certificates) are securely provisioned and protected.

  - **Cryptographic Compliance Evaluation:** Verify that cryptographic algorithms and parameters used for signature verification meet or exceed the requirements defined in Annex K and are appropriate to the execution environment's capabilities. Confirm that deprecated or weak cryptographic mechanisms are not used.

  - **Integrity Failure Handling Tests:** Introduce controlled integrity failures by modifying executable files, altering update packages and/or using invalid or untrusted signatures. Verify that such files are not executed, installed, or applied and do not partially activate or degrade security functions.

  - **Event Logging and Monitoring Verification:** Inspect logs and audit trails to confirm that integrity verification failures are recorded as cybersecurity-relevant events and the logged information allows to support diagnosis and incident investigation. Verify that failure events do not expose sensitive cryptographic material.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product verifies the integrity of its executable files and updates using digital signatures.

  - Cryptographic mechanisms comply with the requirements specified in Annex K and are proportionate to the execution environment.

  - Executables files and updates that fail integrity verification are reliably prevented from being executed, installed, or applied.

  - Integrity verification failures are consistently detected and recorded as cybersecurity-relevant events.

  The assessment is **failed if any of the following occur**:

  - Executable files or updates are executed or installed without successful integrity verification.

  - Digital signature verification does not comply with Annex K cryptographic requirements.

- Integrity verification failures are ignored, bypassed, or inconsistently enforced.
- Integrity verification failures are not logged or are logged with an unclear format.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting integrity verification success and failure scenarios and the use of cryptographic mechanisms that comply with Annex K cryptographic requirements.
  - Logs or audit records showing detection and recording of integrity verification failures.
  - Screenshots or tool outputs demonstrating rejection of tampered executables files or updates.
  - Configuration extracts showing enabled integrity verification mechanisms.
  - Vendor documentation describing signed executable and update processes.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.8.2 REQ-INT-02

- **Assessment reference: REQ-INT-02**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product enforces end-to-end encryption and authentication for any transmitted data, guaranteeing confidentiality, integrity, and authenticity during such transmission. This includes confirming that:

  - The product uses secure communication mechanisms that provide confidentiality, integrity, and peer authenticity as appropriate.
  - The product applies cryptographic algorithms, key lengths, and parameters in accordance with the requirements defined in Annex K, taking into account the capabilities and constraints of the execution environment.
  - The product ensures that insecure, deprecated, or unprotected communication mechanisms are not used for product-related data transmission.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server component, network appliance, cloud-managed agent). All communication paths used by the product, including and where applicable, the update delivery channels, the management and control channels, the telemetry, reporting, or alerting interfaces and the inter-component or agent-to-backend communication. Supported operating systems, platforms, and network configurations. Where applicable, environments with differing capability profiles (e.g. full-featured OS vs constrained execution environment) shall be included. Prepare controlled, untrusted endpoints to simulate plaintext, downgraded, or spoofed server conditions.
  - **Preconditions:** The product is correctly installed and operational in standard configuration. All communication features are enabled using default or recommended configuration. Network connectivity allows observation of communication behaviour. The assessor has authorised access to relevant configuration, logs, and network inspection points.
  - **Required Tools:** Network traffic analysis and inspection tools. Cryptographic protocol analysis tools. Log and audit inspection tools. Configuration inspection utilities. Protocol testing or downgrade simulation tools (where authorised).

- **References:** Vendor documentation describing data flows and communication interfaces, supported communication protocols, cryptographic mechanisms and endpoint authentication. Product security architecture documentation. Annex K cryptographic requirements.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to identify all data transmission interfaces and purposes and claimed use of secure communication mechanisms. Verify that documented cryptographic mechanisms align with Annex K requirements and are appropriate to the execution environment.

  - **Identification of Data Transmission Paths:** Identify all instances where the product transmits data, including outbound and inbound network communications, local or remote management interfaces and update and telemetry channels. Confirm that no undocumented or unintended data transmission paths exist.

  - **Secure Communication Verification:** Observe runtime network traffic to verify that data is transmitted only using secure communication mechanisms (e.g. cryptographically protected protocols) and confidentiality and integrity protection are applied where required. Attempt access via insecure or downgraded protocols (where feasible) and verify that such communication is rejected or prevented.

  - **Cryptographic Compliance Evaluation:** Verify that cryptographic algorithms, key lengths, and protocol versions used for data transmission meet or exceed the requirements specified in Annex K and are proportionate to the capabilities of the execution environment. Confirm that deprecated, weak, or non-cryptographic protection mechanisms, based on the requirements specified in Annex K, are not used.

  - **Resistance to Modification and Interception:** Perform controlled tests to evaluate whether data modification in transit is detected and rejected and/or unauthorised interception or replay attempts are not successful. Observe system behaviour and logs to confirm correct handling of such events.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - All product data transmissions are protected against unauthorised access and modification.

  - Secure communication mechanisms compliant with Annex K are consistently used.

  - Cryptographic protections are appropriate to the execution environment and correctly enforced.

  - No unprotected, downgraded, or insecure data transmission paths are identified.

  The assessment is **failed if any of the following occur**:

  - Data is transmitted without confidentiality or integrity protection.

  - Secure communication mechanisms do not comply with Annex K cryptographic requirements.

  - Insecure or deprecated protocols are used for product data transmission.

  - Protection is inconsistently applied across different communication paths.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting identified data transmission paths and secure communication verification results.

  - Network captures or analysis outputs demonstrating encrypted and integrity-protected traffic.

  - Configuration extracts showing enabled secure communication settings.

  - Logs or audit records confirming secure communication enforcement or rejection of insecure attempts.

  - Vendor documentation describing communication architecture and cryptographic mechanisms.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.9 Data minimisation

6.9.1 REQ-DM-01

- **Assessment reference: REQ-DM-01**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product ensures that personal data is collected, processed, or transmitted based on to what is necessary in relation to that intended purpose. This includes confirming that:

  - Personal data processing by the product is lawful, fair, and transparent.
  - Data collection and transmission are limited to the minimum necessary for the product's cybersecurity functionality.
  - Personal data is secured during its processing (e.g. encrypted and access-controlled).

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment where telemetry, reporting, and update functions can operate normally. Enable logging and monitoring to identify personal data collection and transmission points. Obtain all relevant vendor documentation on privacy policy and DPIA or equivalent documentation. Set up network traffic capture and file system monitoring to analyse data handling behaviour.
  - **Preconditions:** The product is installed with default configuration and telemetry/data-sharing features active. User account creation and initial consent flows (if applicable) have been completed. Logging for data access, transmission, and configuration changes is enabled. Administrative credentials are available for privacy configuration inspection.
  - **Required Tools:** Network packet analysers to capture outbound data flows. File monitoring tools to detect locally stored personal data. Log analysis tools to correlate personal data-related events. Policy documentation analysis utilities or templates.
  - **References:** Vendor documentation and privacy policy declaration.

- **Assessment Activities:**

  - **Documentation Review:** Review the vendor's documentation to confirm that: i) data collection practices are clearly described, including the types of personal data collected, purpose, storage location, and retention period, ii) the product implements data minimisation, collecting only what is strictly necessary its intended purpose, iii) data is secured (e.g. anonymised or pseudonymised, encrypted and access-controlled) where possible and iv) the vendor has performed a DPIA or equivalent documentation.
  - **Functional Verification - Personal Data Collection:** Install and initialise the product under normal user conditions. Check whether the product provides accessible information about what data is collected and how it is used.
  - **Cybersecurity Validation - Network and Storage Inspection:** Capture network traffic during product operation (e.g. updates, reporting, telemetry). Analyse payloads to identify any transmitted personal data (e.g. usernames, IP addresses, device identifiers and equivalent data). Confirm that personal data, if transmitted, is secured (e.g. anonymised or pseudonymised, encrypted and access-controlled). Verify that only data necessary for product functionality (e.g. threat reports, system identifiers and equivalent data) is transmitted. Inspect local storage to ensure that personal data (if any) is stored in encrypted or access-controlled locations and data retention complies with vendor documentation and privacy policy.

- **Cybersecurity Validation - Data Minimisation and Purpose Limitation:** Analyse data categories collected by the product to ensure they correspond to legitimate and declared cybersecurity purposes to confirm that no excessive or unrelated data (e.g. user files, browsing history, or personal content) is collected.
- **Logging and Audit Verification:** Review product logs related to data processing to confirm that all personal data handling events (e.g. collection, transmission, deletion) are logged. Validate that no sensitive personal data is recorded in plaintext within logs.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Collects and transmits only the necessary personal data for that indeed purpose.
  - Protects all personal data during its processing (e.g. storage and transmission) via encryption and access control or equivalent methods.

  The assessment is **failed if any of the following occur**:

  - Personal Data collection, processing and transmission exceeds what is necessary for intended purposes.
  - Personal data is stored or transmitted in plaintext.
  - Logging of personal data-related actions is missing or incomplete.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation and privacy statements.
  - Network capture logs showing encrypted, minimal, or no personal data transmission.
  - File system inspection reports identifying storage location and encryption status of user data.
  - Audit logs capturing data collection, processing and transmission events.
  - DPIA or equivalent documentation.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.10 Availability protection

6.10.1 REQ-AP-01

- **Assessment reference: REQ-AP-01**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product identifies and informs about the critical cybersecurity relevant components of the product and the reason for that assignment. This includes confirming that:

  - The product systematically identifies all components that are critical to cybersecurity.
  - The product documents these components and their security relevance in a clear, structured, and auditable manner.
  - The product provides a clear and reasoned rationale explaining why each identified component is considered critical, taking into account the product's design and architecture, its intended use and deployment context and the declared cybersecurity objectives (e.g. prevention, detection, response, integrity, availability).

- **Assessment Preparation:**

  - **Test Environment:** This assessment is primarily focused on documentation and design elements and shall be conducted using the product's documented architecture, design descriptions, and security documentation and, where applicable, the representative deployment models of the product (e.g. endpoint agent, server component, cloud-managed service). No specific runtime test environment is required unless necessary to validate the correspondence between documentation and implementation.

  - **Preconditions:** The vendor has made available documentation describing the product architecture and security design. The declared cybersecurity objectives of the product are documented. The assessor has access to technical documentation to understand product components, interfaces, and data flows.

  - **Required Tools:** Documentation review and analysis tools. Architecture modelling or diagram review tools. Configuration inspection tools (to validate documented components against implementation). Issue-tracking or component inventory tools (where provided).

  - **References:** Vendor-provided documentation describing the product architecture and component structure, the security design and threat model and the declared cybersecurity objectives. Relevant ETSI or ISO/IEC standards addressing secure design, asset identification, or risk-based security management. Any internal component classification or security impact analysis used by the vendor.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to verify that cybersecurity-relevant components are explicitly identified and the documentation covers all relevant parts of the product, including software, services, update mechanisms, and management interfaces. Confirm that the documentation is sufficiently detailed to support independent understanding of the product's security-critical elements.

  - **Identification of Critical Components:** Verify that the identified critical components include, as applicable, the core detection and protection engines, the update, integrity, and self-protection mechanisms, the authentication, authorisation, and management components, the communication interfaces and cryptographic modules and the configuration, policy, and logging subsystems. Confirm that dependencies and supporting components critical to security objectives are included.

  - **Rationale and Justification Evaluation:** Assess whether the documentation provides a clear rationale for classifying each component as critical, based on its impact on security objectives if compromised, its exposure to threat actors or untrusted inputs, its role in enforcing or supporting security controls [i.7]. Verify that the rationale is consistent with the product's design, intended use, and declared cybersecurity objectives.

  - **Completeness and Consistency Checks:** Compare documented critical components against the architectural diagrams, the data flow descriptions and the observed product functionality. Verify that no obvious cybersecurity-critical components are omitted or not justified.

  - **Traceability and Maintenance Evaluation:** Verify that the documentation allows traceability between critical components and implemented security measures and it is maintained and updated in line with product evolution, where such information is available. Confirm that the identification supports downstream activities such as vulnerability handling [i.6] and security updates.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - All critical cybersecurity-relevant components of the product are identified and documented.

  - Each identified component includes a clear and reasoned rationale explaining its criticality.

  - The rationale is consistent with the product's design, intended use, and declared cybersecurity objectives.

  - The documentation is complete and coherent to support independent assessment and verification.

  The assessment is **failed if any of the following occur**:

  - Critical cybersecurity-relevant components are missing, undocumented, or ambiguously identified.

- The rationale for determining component criticality is absent, unclear, or inconsistent.
- Documentation does not align with the actual product design or functionality.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Vendor documentation identifying and describing critical cybersecurity-relevant components.
  - Architectural diagrams and component inventories.
  - Security design or threat model documentation explaining component criticality.
  - Cross-references linking components to declared cybersecurity objectives.
  - Assessment notes or reports documenting completeness and consistency checks.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.10.2 REQ-AP-02

- **Assessment reference: REQ-AP-02**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product protects the availability of its essential cybersecurity functions during and after an operational incident. This includes confirming that:

  - The product implements resilience measures to handle unexpected termination of cybersecurity-critical components or interruption of update operations.
  - The product ensures that, following such incidents, the product restores essential services where feasible, or enters a controlled degraded protection state that avoids unsafe behaviour and informs the security administrative user.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server-based protection, virtual appliance, cloud-managed agent), operating systems and execution environments officially supported by the vendor and/or update mechanisms and management interfaces relevant to essential cybersecurity functions. Where applicable, the environment shall support controlled simulation of component failures and interrupted update scenarios.
  - **Preconditions:** The product is installed and operating with its default configuration. Essential cybersecurity functions and the associated critical components are identified as documented by the vendor. Restart, recovery, logging, and notification mechanisms are enabled using default or recommended settings. Access to administrative interfaces and logs is available.
  - **Required Tools:** Process and service control tools (e.g. to terminate or suspend components). Update simulation or interruption tools. Log and audit inspection tools. System monitoring tools to observe service availability and recovery behaviour. Notification or alert inspection tools.
  - **Reference Documentation:** Vendor documentation describing essential cybersecurity functions, resilience, recovery, and fault-handling mechanisms and update procedures and failure handling. Product architecture and design documentation. Any internal definitions of degraded or fail-safe operating modes.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation identifies essential cybersecurity functions and describes expected behaviour during component failure or update interruption. Confirm that documented resilience measures align with the requirement options (i.e. recovery or degraded protection state).

  - **Identification of Cybersecurity-Critical Components:** Identify the components required to support essential cybersecurity functions (e.g. detection engines, update services, self-protection modules). Verify traceability between documented components and implemented services or processes.

  - **Unexpected Termination Simulation:** Intentionally terminate or disrupt a cybersecurity-critical component during normal operation. Observe and verify that the product re-establishes the service/function within a reasonable time, or transitions into a degraded protection state as documented.

  - **Update Interruption Testing:** Initiate a product or detection update operation and deliberately interrupt it (e.g. network loss, process termination). Verify that partially applied or unsafe updates are not activated and the product either resumes safely, rolls back, or enters a degraded protection state.

  - **Degraded Protection State Validation:** Where a degraded protection state is entered, verify that no unsafe update or content is applied, core safety guarantees are maintained and the security administrative user is clearly notified of the degraded protection state. Inspect notifications, alerts, or logs to confirm visibility and clarity of the degraded state.

  - **Recovery and Post-Incident Behaviour:** Verify that the product can recover from the degraded state once conditions are restored and maintains integrity and consistency of essential cybersecurity functions after recovery. Confirm that incident handling events are logged as cybersecurity-relevant events.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product protects the availability of its essential cybersecurity functions during and after operational incidents.

  - Unexpected termination of cybersecurity-critical components or interrupted updates results in service re-establishment, or entry into a controlled degraded protection state.

  - In a degraded protection state, no unsafe update or content is applied and the security administrative user is notified.

  - Resilience behaviour is consistent, reliable, and observable during testing.

  The assessment is **failed if any of the following occur**:

  - Essential cybersecurity functions become unavailable without recovery or controlled degradation.

  - Interrupted updates result in unsafe, partial, or inconsistent product states.

  - Degraded protection states allow unsafe behaviour or are not clearly indicated to the administrative user.

  - Resilience or recovery behaviour is undocumented, inconsistent, or unreliable.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting component termination and update interruption scenarios and observed recovery or degradation behaviour.

  - Logs or audit records showing incident detection, recovery actions, and degraded state transitions.

  - Screenshots or captures of administrative notifications indicating degraded protection.

  - Configuration extracts showing enabled resilience and recovery mechanisms.

  - Vendor documentation describing resilience design and failure handling.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.10.3 REQ-AP-03

- **Assessment reference: REQ-AP-03**
- **Assessment Objective:**

  The objective of this assessment is to verify that, where the product controls externally accessible interfaces used to support its documented essential cybersecurity functions, the product ensure their protection by implementing DoS mitigations that may impair such functions. This includes confirming that:

  - The product identifies the externally accessible interfaces under its control used to support essential cybersecurity functions.
  - The product implements mitigation measures to protect such interfaces against DoS conditions.
  - The product ensures that attempted or actual DoS conditions do not impair the availability or reliability of the product's documented essential cybersecurity functions.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including a deployment of the product exposing one or more externally accessible interfaces under product control, supported operating systems, platforms, and network configurations and network conditions allowing controlled generation of high request volumes or malformed traffic. Where applicable, both local and remote interface exposure scenarios shall be assessed.
  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings. Essential cybersecurity functions and their supporting interfaces are identified and documented. DoS mitigation mechanisms are enabled using default or recommended configuration. Access to administrative configuration, monitoring interfaces, and logs is available.
  - **Required Tools:** Network traffic generation and load-testing tools. Request rate simulation or throttling test tools. System and application monitoring tools (e.g. CPU, memory, connection counts). Log and audit inspection tools. Configuration inspection utilities.
  - **Reference Documentation:** Vendor documentation describing externally accessible interfaces under product control and implemented DoS mitigation mechanisms. Product architecture and design documentation. Documentation identifying essential cybersecurity functions and their dependencies.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation identifies externally accessible interfaces controlled by the product and describes implemented DoS mitigation measures. Confirm that such interfaces are linked to documented essential cybersecurity functions.
  - **Interface Identification and Classification:** Identify externally accessible interfaces that accept inbound requests, data, or control commands and are not solely protected by OS-level controls. Verify that these interfaces are correctly classified as security-relevant where they support essential cybersecurity functions.
  - **Configuration and Mechanism Inspection:** Inspect product configuration to verify the presence of DoS mitigation mechanisms (e.g. allow-listing or access control rules, rate limiting or request throttling, resource usage controls such as connection caps or memory limits, and/or auto-scaling or equivalent load management features). Verify that such mechanisms are enabled by default or clearly recommended.
  - **DoS Condition Simulation:** Perform controlled simulations of DoS-like conditions, such as high request rates, repeated connection attempts and/or resource-intensive requests. Observe whether mitigation mechanisms are activate as designed, prevent exhaustion of critical resources and preserve availability of essential cybersecurity functions.

- **Availability and Resilience Verification:** Verify that, during and after simulated DoS conditions essential cybersecurity functions remain available, or graceful degradation occurs without complete loss of protection. Inspect logs and monitoring outputs to confirm detection and handling of DoS conditions.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Externally accessible interfaces under the product control that support essential cybersecurity functions are protected by DoS mitigation measures.
  - Implemented mechanisms mitigate DoS conditions under realistic attack scenarios.
  - Essential cybersecurity functions remain available or degrade gracefully without loss of security guarantees.
  - DoS mitigation behaviour is consistent, documented, and observable during testing.

  The assessment is **failed if any of the following occur**:

  - Externally accessible interfaces lack strong and reliable DoS mitigation mechanisms.
  - DoS conditions can exhaust resources and impair essential cybersecurity functions.
  - Mitigation mechanisms are disabled by default, trivially bypassed or do not mitigate DoS conditions.
  - Availability degradation is uncontrolled or undocumented.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting identified externally accessible interfaces and DoS simulation scenarios and outcomes.
  - Logs or audit records showing activation of mitigation mechanisms.
  - Monitoring data demonstrating resource usage and availability during testing.
  - Configuration extracts showing enabled DoS mitigation controls.
  - Vendor documentation describing interface protection and availability measures.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.10.4 REQ-AP-04

- **Assessment reference: REQ-AP-04**

- **Assessment Objective:**

  The objective of this assessment is to verify that the product ensures robust safeguards against erroneous or malicious denylisting of legitimate entities depending on the UC, to confirm that:

  - For all UCs, mechanisms exist to review, override, and revert denylisting decisions in a controlled and auditable manner.
  - For all UCs, critical system components, product components, and trusted update sources are protected against irreversible denylisting.
  - For all UCs, the implemented controls reduce the risk of system disruption, false positives, and operational denial of service.
  - For UCs with High SP (i.e. UC3, UC4, UC5) and where technically feasible, detection decisions are validated using multiple independent indicators prior to enforcement, where technically feasible.

- For UCs with High SP (i.e. UC3, UC4, UC5), the product monitors and detects abnormal denylisting patterns, including large-scale or system-impacting events.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including endpoint protection components, management console (if applicable) and update and reputation services. Representative operating systems and application environments. Test environment supporting introduction of benign files, processes, and services for controlled evaluation.

  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings. Access to administrative and user roles, logging and audit capabilities, detection and response configuration settings and availability of test artefacts representing legitimate entities (e.g. files, processes, services).

  - **Required Tools:** Monitoring tools for system and process behaviour. Log collection and analysis tools. Traffic analysis tools (for reputation or cloud checks). Test harness for simulating detection and denylisting scenarios. Fault injection tools (to simulate incorrect or large-scale denylisting events).

  - **References:** Vendor documentation describing the detection logic and decision-making processes, the false positive mitigation and validation approaches, the denylisting management, override, and rollback procedures and the safeguards for critical components and update sources. Product operational and administrative guides.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to confirm the availability of override and recovery capabilities and the protections for essential components and trusted sources for all UCs, and the use of multiple independent indicators (e.g. signatures, heuristics, behavioural analysis, reputation) in detection decisions, the defined mechanisms for monitoring and detecting abnormal denylisting behaviour for UCs with High SP (i.e. UC3, UC4 and UC5).

  - **Validation of Detection Decisions:** Introduce test cases involving borderline or ambiguous detections. Verify that, where technically feasible, the product utilises multiple independent indicators before enforcing denylisting and avoids enforcement when insufficient evidence is present. Inspect logs and outputs to confirm decision rationale (e.g. multi-factor detection criteria).

NOTE 1: This activity only applies to products that fall in UCs with High SP (i.e. UC3, UC4, UC5).

  - **Detection of Abnormal Denylisting Behaviour:** Simulate scenarios involving multiple legitimate files or processes being flagged simultaneously and high-frequency or large-scale denylisting events. Verify that the product detects abnormal patterns (e.g. spikes, system-wide impact), generates alerts or triggers safeguards and prevents uncontrolled propagation of denylisting actions.

NOTE 2: This activity only applies to products that fall in UCs with High SP (i.e. UC3, UC4, UC5).

  - **Override and Reversion Mechanisms:** Initiate denylisting of legitimate entities and verify that authorised users can review decisions via management interfaces, override or allowlisting actions can be applied and previously denylisted entities can be restored. Confirm that all actions are controlled by appropriate access rights, actions are logged and auditable and automated safeguards (if implemented) can revert erroneous denylisting.

  - **Protection of Critical Components and Truste Sources:** Attempt to denylist essential product components, core OS components and trusted update servers or update packages. Verify that the product prevents such actions unless strict validation criteria are met, denylisting of critical elements cannot be irreversible and recovery mechanisms exist in case of erroneous blocking.

  - **Configuration and Runtime Behaviour Inspection:** Inspect configuration to ensure the denylisting policies include safeguards and thresholds and critical assets are explicitly protected. Monitor runtime logs to confirm the detection and handling of abnormal events, the execution of overrides and recovery actions and the traceability of all relevant decisions.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - For all UCs, authorised users and/or automated safeguards can review, override, and revert denylisting decisions in a controlled and auditable manner.
  - For all UCs, essential components and trusted update sources are protected against irreversible or unjustified denylisting.
  - For all UCs, all mechanisms operate reliably without causing systemic disruption.
  - For UCs with High SP (i.e. UC3, UC4, UC5) and where technically feasible, detection decisions use multiple independent indicators where technically feasible.
  - For UCs with High SP (i.e. UC3, UC4, UC5), the product monitors, detects and responds to abnormal or large-scale denylisting behaviour.

  The assessment is **failed if any of the following occur**:

  - For all UCs, no effective override or recovery mechanisms exist.
  - For all UCs, denylisting actions cannot be audited or are not controlled.
  - For all UCs, critical components or trusted sources can be permanently blocked without safeguards.
  - For all UCs, erroneous denylisting results in unrecoverable system impact.
  - For UCs with High SP (i.e. UC3, UC4, UC5) and where technically feasible, detection decisions rely on single or weak indicators where stronger mechanisms are feasible.
  - For UCs with High SP (i.e. UC3, UC4, UC5), the product fails to detect and prevent large-scale or abnormal denylisting events.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test reports covering detection validation, abnormal scenarios, and recovery mechanisms.
  - Logs demonstrating detection decision rationale, abnormal denylisting detection and override and rollback actions.
  - Screenshots or outputs from management interfaces showing the review and override processes and the alerts and safeguards activation
  - Configuration files and policies defining denylisting controls.
  - Vendor documentation describing implemented safeguards and architectures.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.11 Non-interference

6.11.1 REQ-NI-01

- **Assessment reference: REQ-NI-01**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product minimises system or network resources consumption in order to not block the availability of services on the host system or connected networks. This includes confirming that:

  - The product is designed and implemented to minimise consumption of system and network resources required for its operation.
  - Avoids causing resource exhaustion or starvation that could impair the availability of services on the host system, or the availability of services on connected networks.
  - Maintains acceptable operational behaviour under normal, peak, and adverse conditions (e.g. scans, updates, detection events).

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server protection, virtual appliance), supported operating systems and hardware profiles with typical minimum system requirements and representative production-grade configurations, and host systems running common workloads or services expected to coexist with the product. Where applicable, environments with constrained resources shall be included.
  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings. Essential cybersecurity functions are enabled. System monitoring and logging capabilities are active. Baseline system performance metrics (without product activity) are available for comparison.
  - **Required Tools:** System performance monitoring tools (e.g. CPU, memory, disk I/O). Network traffic and bandwidth monitoring tools. Load and stress testing tools. Log and audit inspection tools. Configuration inspection utilities.
  - **Reference Documentation:** Vendor documentation describing performance characteristics and system requirements and resource management or throttling mechanisms. Product architecture documentation. Documentation identifying essential cybersecurity functions and operational modes.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to verify that resource consumption considerations are addressed and guidance is provided on expected system and network usage. Confirm that performance-impacting operations (e.g. scans, updates) are documented.
  - **Baseline Resource Measurement:** Measure baseline system and network resource usage without active product operations. Record baseline metrics, for example CPU, memory, storage I/O, and network bandwidth or other possible parameters.
  - **Operational Resource Consumption Evaluation:** Observe resource consumption during normal product operation (e.g. real-time protection, scheduled or on-demand scanning and update and telemetry activities and other possibles processes). Verify that resource usage remains within reasonable and documented bounds that do not block the availability of services on the host system or connected networks**.**
  - **Peak and Stress Condition Testing:** Execute resource-intensive product operations (e.g. full system scan, large update). Observe whether host system services remain responsive and/or network services remain available and functional. Verify that the product applies internal controls such as prioritisation, throttling, or scheduling to limit impact.

- **Availability Impact Verification:** Verify that the product does not monopolise CPU or memory resources, exhaust storage or bandwidth and/or block or significantly degrade critical host or network services. Inspect logs and system behaviour for indicators of resource exhaustion or contention.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product minimises consumption of system and network resources in normal and peak operational scenarios.
  - Product operation does not block or unreasonably degrade availability of host system services or connected network services.
  - Resource management behaviour is consistent, controlled, and observable.
  - No evidence of uncontrolled resource exhaustion attributable to the product is identified.

  The assessment is **failed if any of the following occur**:

  - The product consumes excessive CPU, memory, storage, or bandwidth.
  - Host system or network services become unavailable or severely degraded due to product operation.
  - Resource-intensive operations lack appropriate limiting or scheduling mechanisms.
  - Resource consumption behaviour is undocumented, uncontrolled, or inconsistent.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting the baseline, operational, stress and peak usage scenarios resource measurements.
  - System and network monitoring outputs showing resource utilisation.
  - Logs indicating performance-related events or safeguards.
  - Configuration extracts showing enabled resource management features.
  - Vendor documentation describing performance and resource considerations.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.11.2 REQ-NI-02

- **Assessment reference: REQ-NI-02**

- **Assessment Objective:**

  The objective of this assessment is to verify that the product provides clear, complete, and accurate documentation describing the environments in which the product is intended to be deployed and used. This includes confirming that:

  - The product explicitly identifies:
    - the required physical and hardware environment;
    - the required logical and software environment;
    - the physical/system requirements necessary for correct and secure operation;
    - the supported OS types and their minimum supported versions.

- The product enables users and administrators to deploy and operate the product in a manner that ensures full functionality, security effectiveness, and availability, in line with the product's declared cybersecurity objectives.

- **Assessment Preparation:**

  - **Test Environment:** This assessment is documentation-focused and shall be conducted using the official product documentation provided to users, administrators, or integrators, including installation guides, deployment manuals, system requirement specifications and online knowledge bases or technical datasheets. No specific runtime or operational test environment is required unless necessary to validate consistency between documentation and product behaviour.

  - **Preconditions:** The latest version of the product documentation is available. Documentation corresponds to the assessed product version. The assessor has access to all publicly provided deployment and installation documentation.

  - **Required Tools:** Documentation review and comparison tools. Version control or document change-tracking tools. Configuration or installation validation tools (where needed to confirm documented requirements).

  - **Reference Documentation:** Vendor-provided documentation describing deployment environments, system and hardware requirements and supported operating systems and versions. Product release notes or lifecycle documentation.

- **Assessment Activities:**

  - **Documentation Review:** Review product documentation to verify that it clearly identifies the intended physical and hardware deployment environment (e.g. endpoint, server, appliance, virtualised environment) and the required hardware resources (e.g. CPU architecture, minimum memory, storage requirements, network interfaces). Verify that the information is presented in a clear, structured, and accessible manner.

  - **Logical and Software Environment Identification:** Verify that the documentation identifies the supported operating system families and editions, the minimum supported OS versions and the required system components, libraries, drivers, or dependencies. Confirm that any limitations or unsupported environments are clearly stated.

  - **Completeness and Clarity Evaluation:** Assess whether the documented requirements are clear and transparent to enable correct installation and operation of all product functions and they clearly distinguish mandatory requirements from recommendations. Verify that information is not ambiguous, outdated, or internally inconsistent.

  - **Consistency and Accuracy Checks:** Compare documented requirements against product installation behaviour and the release notes or version-specific constraints. Verify that documentation is consistent with the product's declared cybersecurity objectives and supported features.

  - **User Notification and Visibility:** Verify that critical deployment and system requirements are clearly visible to users prior to installation and are not hidden in obscure or secondary documentation. Confirm that failure to meet requirements is clearly indicated or warned against in the documentation.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product documentation clearly and accurately identifies the physical/hardware and logical/software deployment environments.

  - The product-related physical system requirements necessary for full product functionality are documented.

  - The supported OS types and minimum supported versions are explicitly stated.

  - The product documentation is complete, clear, consistent, and suitable to support secure and correct deployment.

The assessment is **failed if any of the following occur**:

- The deployment environment or system requirements are missing, incomplete, or unclear.
- The supported OS types or minimum versions are not documented or are ambiguous.
- The documentation is inconsistent with product behaviour or declared capabilities.
- The information provided to enable secure and correct product deployment is not clear and not all the required information is presented.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Copies or references to official product documentation describing deployment and system requirements.
  - Screenshots or extracts highlighting documented environments and supported OS versions.
  - Assessment notes demonstrating completeness and clarity checks.
  - Versioned documentation references corresponding to the assessed product release.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.12 Attack surface minimisation

6.12.1 REQ-ASM-01

- **Assessment reference: REQ-ASM-01**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product ensures that its processes, services, and components under the control of the manufacturer are protected against privilege abuse or escalation and use the minimum privilege necessary to perform the documented essential cybersecurity functions within the execution environment. This includes confirming that:

  - The product implements the PoLP for all processes, services, and components under the manufacturer's control operates with no more privileges than strictly necessary to perform its documented essential cybersecurity functions.
  - The product limits the impact of compromise, misuse, or malfunction of individual components by avoiding unnecessary elevated privileges.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server-based protection, appliance, cloud-managed agent). Supported operating systems and execution environments. Typical privilege separation mechanisms provided by the execution environment (e.g. user vs administrator, service accounts, system/root contexts, etc.). Where applicable, environments with different privilege models (e.g. desktop OS vs server OS) shall be included.
  - **Preconditions:** The product is installed and operational using default or recommended configuration. Essential cybersecurity functions and their supporting components are documented. Access is available to inspect process, service, and component execution contexts. Administrative access is available where required to observe privilege assignments.
  - **Required Tools:** Process and service inspection tools. OS-level privilege and permission inspection utilities. Configuration and policy inspection tools. Log and audit inspection tools. Debugging or tracing tools (where permitted).

- **Reference Documentation:** Vendor documentation describing essential cybersecurity functions and component architecture and execution model. Product security architecture or design documentation. OS or platform documentation describing privilege and permission models.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation identifies essential cybersecurity functions and the processes, services, or components implementing those functions. Confirm that documented privilege requirements are described or implied for major components.

  - **Component and Process Identification:** Identify processes, services, and components under the manufacturer's control, including core detection and engine processes, update and communication components and management, logging, and self-protection components. Map these components to documented essential cybersecurity functions.

  - **Privilege Level Inspection:** Inspect the execution privileges of identified components, including user or service account context, required system or kernel-level privileges and file system, network, and inter-process access rights. Verify that elevated privileges (e.g. administrator, system/root) are used only where necessary.

  - **Least-Privilege Enforcement Evaluation:** Attempt to identify components that run with broader privileges than required for their function and share high-privilege contexts unnecessarily. Verify that privilege separation exists between components with different security roles where feasible.

  - **Runtime and Abuse Scenario Observation:** Observe product behaviour during normal operation and during representative fault or stress scenarios. Verify that lack of unnecessary privileges does not impair required functionality and components cannot perform actions outside their documented functional scope. Inspect logs to ensure that privilege-related enforcement or failures are detectable.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Processes, services, and components operate with privileges limited to what is necessary for their documented essential cybersecurity functions.

  - Elevated privileges are used only where justified by functional necessity and execution environment constraints.

  - Privilege separation between components is implemented where feasible.

  - No component is found to operate with unjustified or excessive privileges.

  The assessment is **failed if any of the following occur**:

  - Components operate with excessive or unjustified privileges.

  - Privilege separation is absent where reasonably feasible.

  - Elevated privileges are used by default without clear functional necessity.

  - The privilege model is undocumented, inconsistent, or cannot be reliably verified.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting identified processes, services, and components and their privilege levels and mapping between privileges and essential functions.

  - Screenshots or tool outputs showing process/service execution contexts.

  - Configuration or policy extracts defining privilege assignments.

  - Logs or audit records supporting observed privilege behaviour.

  - Vendor documentation describing component architecture and privilege design.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.12.2 REQ-ASM-02

- **Assessment reference: REQ-ASM-02**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product ensures that any exposed local or network service for integration (for integration with other components or the underlying OS) are protected with authentication (where technically applicable) and access-controls. This includes confirming that:

  - The product identifies all local and network services it exposes for integration with other product components, external components; and/or the underlying operating system.
  - The product implements appropriate authentication mechanisms, where technically applicable, to ensure that only authorised entities can access such services.
  - The product enforces access controls that restrict service interaction according to the PoLP.
  - The product prevents unauthorised access, misuse, or abuse of exposed services that could compromise essential cybersecurity functions.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server-based protection, virtual appliance), operating systems and platforms officially supported by the product and all local and network services exposed by the product (e.g. local inter-process communication interfaces, loopback or local TCP/UDP services, network-accessible APIs or integration endpoints and equivalent services). Where applicable, both default and optional integration interfaces shall be included.
  - **Preconditions:** The product is installed, configured, and operational using default or recommended settings. Integration features exposing local or network services are enabled. Essential cybersecurity functions and their supporting services are documented. Access is available to observe service configuration, authentication behaviour, and logs.
  - **Required Tools:** Service enumeration and inspection tools. Network scanning and service interaction tools. Authentication and authorisation testing tools. OS-level permission and access control inspection utilities. Log and audit inspection tools.
  - **Reference Documentation:** Vendor documentation describing exposed local and network services, intended integration use cases and authentication and access-control mechanisms. Product architecture and design documentation. OS or platform documentation describing supported access-control mechanisms.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to verify that all the exposed local and network services are identified and the intended consumers of each service are described. Confirm that documentation specifies applied authentication and access-control measures.
  - **Service Identification and Exposure Analysis:** Enumerate all local and network services exposed by the product. Verify whether each service is required to support essential cybersecurity functions and accessible only locally or also via network interfaces. Identify services that accept inbound connections, requests, or commands.
  - **Authentication Mechanism Verification:** For each exposed service, verify whether authentication is implemented where technically applicable and mandatory before accessing sensitive operations or data. Attempt controlled access without valid authentication and confirm that access is denied where required.

- **Access-Control Enforcement Evaluation:** Verify that access controls restrict service usage based on identity (e.g. user, service, component), role or privilege level and/or execution context (e.g. local vs. remote). Attempt controlled privilege-violating interactions and confirm they are rejected.
- **Runtime Behaviour and Logging:** Observe runtime service behaviour to verify that authentication and access-control checks are consistently enforced and unauthorised access attempts are detected. Inspect logs to confirm that access violations are recorded without exposing sensitive information.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - All exposed local and network services are protected by appropriate access-control mechanisms.
  - All exposed local and network services are authenticated, where technically applicable, and enforced consistently.
  - Only authorised entities can access protected services and operations.
  - No exposed service allows unauthorised access or excessive privileges.

  The assessment is **failed if any of the following occur**:

  - Exposed services lack authentication where it is technically applicable.
  - Access controls are missing, inconsistently enforced, or they do not apply the access conditions.
  - Unauthorised entities can access or misuse exposed services.
  - Service protection relies solely on undocumented or external assumptions.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting exposed service identification and authentication and access-control tests performed.
  - Screenshots or tool outputs showing enforced authentication or access denial.
  - Configuration extracts defining service protection settings.
  - Logs or audit records showing unauthorised access attempts and enforcement actions.
  - Vendor documentation describing service exposure and protection mechanisms.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.12.3 REQ-ASM-03

- **Assessment reference: REQ-ASM-03**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product, unless required for the product's functionality, intended purpose or explicitly enabled by an authorised entity, ensures a disable by default state of all local or network services for optional integration with other components or the underlying OS, whenever they are provided. This includes confirming that:

  - The product adheres to the principle of secure-by-default configuration for all local or network services it provides.
  - The product ensures that optional integration services are disabled by default and enabled only when required for essential product functionality or explicitly activated by an authorised entity.

- The product reduces the exposed attack surface by preventing unnecessary service exposure without explicit operational or administrative intent.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server-based protection, appliance, cloud-managed agent). Supported operating systems and execution platforms. Product installations performed using default installation and configuration settings. Where applicable, both fresh installations and upgraded deployments shall be included.

  - **Preconditions:** The product is installed using default settings with no optional integrations explicitly enabled. All local and network services exposed by the product are identifiable. Administrative access is available to inspect configuration and enable services where permitted.

  - **Required Tools:** Service and process enumeration tools. Network scanning and port inspection utilities. Configuration inspection tools. Log and audit inspection tools. Administrative interfaces or management console access.

  - **Reference Documentation:** Vendor documentation describing optional integration services, default configuration behaviour and procedures for enabling services. Product architecture and design documentation. Documentation describing roles and authorisation mechanisms.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to identify all local and network services provided by the product and which services are essential and which are optional. Verify that documentation clearly states the default service states and the conditions under which optional services may be enabled.

  - **Default Installation Inspection:** Perform a fresh installation using default settings. Enumerate all local and network services exposed immediately after installation. Verify that optional integration services are disabled by default and only services strictly required for the essential product functionalities are enabled.

  - **Configuration and Enablement Validation:** Inspect configuration interfaces to verify that optional services can only be enabled through explicit action and their enablement requires appropriate authorisation (e.g. administrative user, management system). Verify that unauthorised users cannot enable optional services.

  - **Logging and Auditability Check:** Verify that service enablement and disablement actions are logged and auditable records identify the authorised entity performing the action. Confirm that no silent or undocumented service activation occurs.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Optional local or network services are disabled by default.

  - Only services required for essential functionalities are enabled after default installation.

  - Optional services can be enabled only through explicit action by an authorised entity.

  - Service exposure behaviour is clearly documented, controlled, and auditable.

  The assessment is **failed if any of the following occur**:

  - Optional integration services are enabled by default without necessity.

  - Services can be enabled implicitly or without appropriate authorisation.

  - Default installation exposes unnecessary local or network services.

  - Service enablement behaviour is undocumented, uncontrolled, or not auditable.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting default installation service exposure and optional service enablement tests.
  - Service enumeration outputs demonstrating disabled-by-default behaviour.
  - Configuration extracts showing service enablement controls.
  - Logs or audit records showing authorised service enablement actions.
  - Vendor documentation describing default service states and enablement procedures.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.13 Exploit mitigation

6.13.1 REQ-EM-01

- **Assessment reference: REQ-EM-01**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product is capable of disclosing identified incidents relevant to its essential cybersecurity functions. This includes confirming that:

  - The product provides a mechanism that, when enabled by the administrator, discloses identified incidents to the security administrative user.
  - The product provides a mechanism that, when enabled by the administrator, communicates clear and actionable information on possible mitigation steps.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server-based protection, cloud-managed service). A management interface used by the security administrative user, such as a web management console, a centralised management system and/or local administrative interface. A controlled environment capable of generating test incidents or alerts (e.g. simulated malware, policy violations, integrity failures or other possible incidents or test).
  - **Preconditions:** The product is installed, configured, and operating normally. Incident detection and logging mechanisms are enabled. An administrative option exists to enable or disable incident disclosure to the security administrative user. Access to the administrative user interface and relevant logs is available.
  - **Required Tools:** Incident or alert simulation tools. Management console or administrative interface access. Log and audit inspection tools. Notification inspection tools (e.g. dashboards, alerts, messages).
  - **Reference Documentation:** Vendor documentation describing the incident detection and classification, the incident notification and disclosure features and the configuration options for administrators. Product security architecture and incident-handling documentation.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to verify that the identified incidents are defined or classified and the disclosure and notification mechanisms are described. Confirm that the documentation clearly explains how an administrator enables incident disclosure and what information is provided.

- **Configuration and Enablement Verification:** Inspect configuration settings to verify that the incident disclosure can be enabled or disabled by an authorised administrative user and the disclosure is not enabled by implicit or undocumented means. Verify that configuration changes are auditable.
- **Incident Identification and Disclosure Testing:** Trigger one or more representative cybersecurity incidents (e.g. detection event, integrity failure, blocked attack). Verify that, when disclosure is enabled, the incident is presented to the security administrative user and the notification is delivered in a short period of time and clearly identifiable as a cybersecurity event.
- **Mitigation Information Evaluation:** Verify that disclosed incident information includes a description of the incident and a possible or recommended mitigation steps where applicable. Assess whether mitigation information is understandable and actionable and appropriate to the role of the security administrative user.
- **Behaviour When Disclosure Is Disabled:** Disable incident disclosure and trigger one or more representative cybersecurity incidents (e.g. detection event, integrity failure, blocked attack). Verify that the incidents are still internally recorded and their disclosure to the security administrative user does not occur when disabled.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product identifies cybersecurity incidents relevant to its operation.
  - When enabled by an administrator user, the product discloses identified incidents to the security administrative user.
  - Disclosed information includes possible mitigation steps where applicable.
  - Disclosure behaviour is configurable, controlled, and consistent with documentation.

  The assessment is **failed if any of the following occur**:

  - Identified incidents are not disclosed when disclosure is enabled.
  - Disclosure does not include meaningful mitigation information when such information is expected.
  - Incident disclosure cannot be controlled or enabled by an authorised administrator.
  - Disclosure behaviour is undocumented, inconsistent, or unreliable.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Tests or assessment reports documenting the incident scenarios executed and the disclosure behaviour observed.
  - Screenshots or captures of management interfaces showing incident notifications.
  - Logs or audit records demonstrating incident detection and disclosure actions.
  - Configuration extracts showing incident disclosure enablement settings.
  - Vendor documentation describing incident disclosure and mitigation guidance.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.13.2 REQ-EM-02

- **Assessment reference: REQ-EM-02**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product relies on the OS for runtime protection measures, and if prevention is not technically feasible, to implement feasible measures to detect such attempts and to preserve the integrity and continued secure operation of the product. This includes confirming that:

  - The product uses OS-provided security mechanisms to protect its critical cybersecurity-relevant components at runtime.
  - The product implements preventive controls to the extent technically feasible to protect against unauthorised modification, code injection or tampering.
  - Where full prevention is not feasible due to execution-environment constraints, the product implemented detection mechanisms to identify tampering attempts and response or preservation measures that maintain integrity and continued secure operation.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server-based protection, virtual appliance). Supported operating systems and platforms offering security functionality (e.g. process isolation and privilege separation, memory protection, code-signing enforcement, OS-level anti-tampering or integrity controls and/or other equivalent methods). Systems allowing controlled simulation of runtime attacks or tampering attempts. Where applicable, multiple OS variants or privilege models shall be included.
  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings. Critical cybersecurity-relevant components are identified and documented by the vendor. OS-level security features required by the product are enabled. Administrative access is available to inspect runtime behaviour and logs.
  - **Required Tools:** OS security and capability inspection tools. Process, memory, and privilege inspection tools. Debuggers or controlled injection/testing frameworks (where permitted). Log and audit inspection tools. Integrity and self-protection test utilities.
  - **Reference Documentation:** Vendor documentation describing the runtime protection and self-protection mechanisms and the dependencies on OS security features. Product security architecture and threat-model documentation. OS platform documentation describing applicable security mechanisms.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation identifies critical cybersecurity-relevant components and describes how OS security functionality is leveraged for runtime protection. Confirm that limitations of prevention and compensating detection measures are documented where applicable.
  - **OS Security Integration Verification:** Inspect product configuration and behaviour to verify the possible use of OS security features (e.g. protected process models, signed-code enforcement, memory execution protection, access-control enforcement, and/or other equivalent security features). Verify that such features are applied to critical components.
  - **Runtime Protection Effectiveness Testing:** Attempt controlled unauthorised actions (e.g. modifying executable or loaded components, injecting code or libraries, attaching debuggers or manipulating process memory and/or other possible actions). Verify that, where technically feasible, attempts are blocked or prevented by OS-level or product-level protections.
  - **Detection and Response Evaluation:** Where prevention is not technically feasible, perform controlled attempts to interfere with runtime components. Verify that such attempts are detected by the product and recorded as cybersecurity-relevant events. Observe whether the product preserves integrity of remaining components, maintains continued secure operation and enters a controlled degraded or protective state if required.

- **Post-Incident Integrity Verification:** Verify that, after detection of tampering, the product remains operational in a secure manner and no unauthorised changes persist. Inspect logs, alerts, or administrative notifications related to runtime protection events.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product leverages OS-provided security functionality to protect critical cybersecurity-relevant components at runtime.
  - Unauthorised modification, code injection, or tampering is prevented where technically feasible.
  - Where prevention is not feasible, detection mechanisms are implemented.
  - Integrity and continued secure operation of the product are preserved following detected attempts.
  - Runtime protection behaviour is consistent, documented, and observable during testing.

  The assessment is **failed if any of the following occur**:

  - The product does not leverage available OS security functionality for runtime protection.
  - Critical components can be modified, injected, or tampered without prevention or detection.
  - Detection mechanisms are missing or do not preserve secure operation.
  - Runtime protection behaviour is undocumented, inconsistent, or unreliable.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting runtime attack or tampering simulations and observed prevention or detection behaviour.
  - Logs or audit records showing detection of runtime tampering attempts.
  - Screenshots or tool outputs demonstrating enforced OS-level protections.
  - Configuration or policy extracts showing enabled runtime security mechanisms.
  - Vendor documentation describing runtime protection and self-protection design.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.14 Monitoring

6.14.1 REQ-MON-01

- **Assessment reference: REQ-MON-01**

- **Assessment Objective:**

  The objective of this assessment is to verify that the product supports logging and monitoring mechanisms of its own or leveraged by the OS to record and where technically feasible, to store cybersecurity-critical events, to access to or modify cybersecurity-critical data, services or functions and to provide access to this information only to authorised entities. This includes confirming that:

  - The product implements or leverages logging and monitoring mechanisms to record cybersecurity-critical events relevant to its essential cybersecurity functions.
  - The product ensures that such events are recorded and stored where technically feasible, with clear details to support (e.g. security monitoring, incident investigation, accountability and traceability).

- The product provides controlled access to recorded events for authorised entities.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server-based protection, cloud-managed service), supported operating systems with native logging and monitoring facilities and a management interface or log access mechanism available to authorised entities. Where applicable, both product-native and OS-integrated logging paths shall be included.

  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings. Logging and monitoring features are enabled using default or recommended settings. Administrative credentials are available to access logs and configure logging behaviour. Storage or forwarding mechanisms for logs (local or remote) are operational where applicable.

  - **Required Tools:** Log inspection and correlation tools. OS-level logging and event-viewer tools. SIEM interfaces (where applicable). Configuration and policy inspection tools. Test utilities to trigger representative cybersecurity-critical events.

  - **Reference Documentation:** Vendor documentation describing logging and monitoring capabilities, event types recorded by the product and access and retention behaviour. Product security architecture documentation. OS documentation describing available logging and monitoring mechanisms.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation identifies all cybersecurity-critical events that are logged and describes the logging and monitoring mechanisms and their configuration. If the use of the product owned logging and monitoring mechanisms are not available, confirm that the use of OS-level logging facilities is documented where applicable.

  - **Event Coverage Verification:** Trigger representative cybersecurity-critical events, such as malware or threat detections, security or detection-rule updates, configuration changes, access to or modification of cybersecurity-critical data, services, or functions and equivalent events. Verify that such events are recorded by the product or underlying OS logging mechanisms.

  - **Log Content and Quality Evaluation:** Inspect recorded log entries to verify that they clearly identify the event type, include contextual information (e.g. timestamp, component, action performed an other related data) and avoid unnecessary exposure of sensitive data. Verify that event records are suitable for security monitoring and analysis.

  - **Storage, Retention, and Feasibility Assessment:** Verify that logged events are stored where technically feasible, either locally or forwarded to a management system or log collector. Evaluate whether storage limitations or constraints are documented where full retention is not feasible.

  - **Access Control and Authorisation Checks:** Verify that access to recorded events is restricted to authorised entities. Confirm that administrative users or management systems can retrieve relevant records, unauthorised users cannot access or modify log data. Inspect audit trails to confirm logging access and configuration actions.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product records cybersecurity-critical events using product-native or OS-leveraged logging mechanisms.

  - Logged events include meaningful and usable information for security monitoring and investigation.

  - Events are stored where technically feasible.

  - Access to event records is provided to authorised entities and protected against unauthorised access.

  - Logging and monitoring behaviour is consistent with vendor documentation.

The assessment is **failed if any of the following occur**:

- Cybersecurity-critical events are not logged or are logged incompletely.
- Logging mechanisms are unreliable, disabled by default without justification, or undocumented.
- Event records lack details for investigation or accountability.
- Logs are inaccessible to authorised entities or improperly exposed to unauthorised access.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting all triggered cybersecurity-critical events and corresponding logged records.
  - Log extracts or screenshots demonstrating recorded events.
  - Configuration extracts showing logging and monitoring settings.
  - Evidence of authorised access to logs (e.g. management console views or equivalent methods).
  - Vendor documentation describing logging, monitoring, and retention behaviour.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.14.2 REQ-MON-02

- **Assessment reference: REQ-MON-02**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product keeps the integrity of detections, updates, configuration changes, access to quarantined files logs and blocks any unauthorised action on them. This includes confirming that:

  - The product implements tamper-resistant logging mechanisms.
  - The product preserves the integrity, reliability, and evidentiary value of logs related to malware detections, security and detection-rule updates, configuration changes and access to or modification of quarantined files.
  - The product protects logged records against unauthorised modification or deletion.
  - The product ensures that only authorised administrative entities can modify, delete, or manage such logs under controlled and auditable conditions.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server protection, cloud-managed service). Operating systems and storage environments supported by the product. Logging storage mechanisms, including local log storage and/or centralised or remote log collectors (where applicable). Where applicable, both product-native and OS-leveraged logging mechanisms shall be included.
  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings. Logging functionality is enabled using default or recommended settings. Logging storage locations (files, databases, or OS log facilities) are identifiable. Administrative credentials and non-administrative credentials are available for testing access controls.
  - **Required Tools:** Log inspection and manipulation tools. File system and permission inspection utilities. OS-level access-control and audit tools. Configuration and policy inspection tools. Test utilities to trigger representative log events.

- **Reference Documentation:** The assessment shall reference vendor documentation describing logging architecture and storage and log protection and access-control mechanisms. Product security architecture documentation. OS platform documentation relating to file integrity and access controls.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation identifies which cybersecurity-critical events are logged and describes log storage locations and protection measures. Confirm that log integrity and tamper-resistance mechanisms are documented.

  - **Log Event Generation and Verification:** Trigger representative cybersecurity-critical events, including threat or malware detections, security updates and detection-rule updates, configuration changes and access to quarantined files. Verify that each event type is logged reliably and clearly identifiable in the log records.

  - **Tamper-Resistance Evaluation:** Attempt controlled unauthorised actions to modify existing log entries, delete log files or records and/or overwrite or truncate logs. Verify that such attempts are prevented by access-control or integrity mechanisms or detected and recorded as cybersecurity-relevant events.

  - **Authorised Administrative Action Testing:** Perform authorised administrative actions, such as log rotation and/or log archival or deletion. Verify that such actions require appropriate administrative privileges and they are auditable and logged.

  - **Quarantined File Access Logging:** Attempt access to quarantined files using authorised and unauthorised contexts. Verify that access attempts are logged, the unauthorised access is prevented and recorded and confirm that corresponding log records are protected against tampering.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The Cybersecurity-critical events are recorded in logs.

  - Attempts to tamper with logs are prevented or reliably detected and recorded.

  - Log integrity protections are active and observable during testing.

  - Logs are protected against unauthorised modification or deletion.

  - Only authorised administrative entities can modify or delete logs.

  The assessment is **failed if any of the following occur**:

  - Cybersecurity-critical events are not logged or are logged incompletely.

  - Log protection relies solely on undocumented or weak mechanisms.

  - Logs can be modified or deleted by unauthorised entities.

  - Administrative log-management actions are not controlled or auditable.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting logged cybersecurity-critical events and tampering and access-control test results.

  - Log extracts or screenshots demonstrating integrity and protection.

  - Configuration or permission snapshots showing restricted log access.

  - Audit records showing authorised log-management actions.

  - Vendor documentation describing log integrity and tamper-resistance mechanisms.

The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.14.3 REQ-MON-03

- **Assessment reference: REQ-MON-03**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product uses integrity protection to detect unauthorised actions of cybersecurity-relevant logs. This includes confirming that:

  - The product implements integrity protection mechanisms for cybersecurity-relevant log records.
  - The product ensures that unauthorised modification or deletion of logs is prevented or reliably detected.
  - The product makes use of one of the following integrity protection controls:
    - cryptographically signing or hashing in accordance with the cryptographic requirements specified in Annex K in a manner proportional to the capabilities of the execution environment;
    - storing files in an append-only access-controlled location;
    - preventing modification or deletion without authorised administrative action.
- **Assessment Preparation:**
  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported product deployment models (e.g. endpoint agent, server protection, cloud-managed service). Supported operating systems and storage backends used for log persistence. Logging mechanisms implemented by the product, including the product-native logging and/or OS-integrated logging facilities (where leveraged). Where applicable, both local and centralised log storage shall be assessed.
  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings. Logging of cybersecurity-relevant events is enabled using default or recommended settings. Log storage locations and formats are identifiable. Administrative and non-administrative accounts are available for access-control testing.
  - **Required Tools:** Log inspection and analysis tools. Cryptographic verification tools (e.g. hash or signature validation). File system and access-control inspection utilities. OS-level audit and event-tracking tools. Test utilities to trigger representative cybersecurity-relevant events.
  - **Reference Documentation:** Vendor documentation describing log integrity protection mechanisms and the use of cryptographic or OS-level controls. Product security architecture documentation. Annex K cryptographic requirements.
- **Assessment Activities:**
  - **Documentation Review:** Verify that vendor documentation identifies cybersecurity-relevant logs requiring integrity protection and describes integrity protection mechanisms applied to logs. Confirm that cryptographic mechanisms, where used, are described as compliant with Annex K.
  - **Log Generation and Baseline Verification:** Trigger representative cybersecurity-relevant events related to the identified logs in the "Documentation Review". Verify that corresponding log records are created and stored.
  - **Integrity Protection Mechanism Evaluation:** Inspect log storage to verify that at least one of the following is implemented: a) cryptographic hashing or signing of log records or log files, b) append-only log storage with enforced access controls and/or equivalent integrity-preserving mechanisms appropriate to the environment. Where cryptographic controls are used, verify algorithm strength and configuration compliance with Annex K.

- **Unauthorised Modification and Deletion Testing:** Attempt controlled unauthorised actions to modify log contents, delete log entries or files and/or truncate or overwrite logs. Verify that such attempts are prevented by access-control or append-only mechanisms or detected through integrity verification mechanisms.
- **Administrative Action Validation:** Perform authorised administrative actions affecting logs, such as log rotation or archival or log deletion or reset. Verify that such actions require appropriate administrative privileges and resulting integrity changes are auditable and recorded.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Cybersecurity-relevant logs are protected by at least one of the integrity protection controls suggested in the requirement.
  - Unauthorised modification or deletion of log records is prevented or reliably detected.
  - Cryptographic integrity mechanisms, where used, comply with Annex K and are proportionate to the execution environment.
  - Log modification or deletion is permitted only through authorised administrative actions.
  - Integrity protection behaviour is consistent and observable during testing.

  The assessment is **failed if any of the following occur**:

  - Logs are stored without integrity protection.
  - Log records can be modified or deleted without detection or authorisation.
  - Cryptographic mechanisms, where used, are weak, misconfigured, or non-compliant with Annex K.
  - Integrity protection behaviour is undocumented, inconsistent or not applied.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting the integrity protection control evaluated and any modification and deletion test results.
  - Log extracts or verification outputs demonstrating integrity protection.
  - Where used, configuration or permission snapshots showing the use of integrity protection controls such as cryptographic signatures or hashes compliant with Annex K requirements, append-only or access-controlled storage.
  - Screenshots or tool outputs validating cryptographic integrity checks.
  - Vendor documentation describing implemented log integrity mechanisms.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.14.4 REQ-MON-04

- **Assessment reference: REQ-MON-04**

- **Assessment Objective:**

  The objective of this assessment is to verify that the product allows authorised entities to reach any related and stored log in the product. This includes confirming that:

  - The product implements access-control mechanisms that restrict access to stored log records.

- The product ensures that only authorised entities, such as administrative users, privileged users, or authorised management systems, can read, export; and/or manage cybersecurity-relevant logs.
- The product prevents unauthorised disclosure, modification, or misuse of log data that could compromise security, privacy, or incident-response effectiveness.

- **Assessment Preparation:**
  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models (e.g. endpoint agent, server-based protection, cloud-managed service), supported operating systems and storage mechanisms used for log persistence and management interfaces or log access mechanisms provided by the product. Where applicable, both local log access and centralised or remote log access scenarios shall be included.
  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings. Logging functionality is enabled using default or recommended configuration. Log storage locations and access interfaces are identifiable. Credentials for authorised (administrative or privileged) and unauthorised users are available for controlled testing.
  - **Required Tools:** Log access and inspection tools. OS-level permission and access-control inspection utilities. Configuration and policy inspection tools. Authentication and authorisation testing tools. Log management or monitoring interfaces.
  - **Reference Documentation: V**endor documentation describing the log access and management features and the role definitions and authorisation mechanisms. Product security architecture documentation. OS or platform documentation describing file and service access-control mechanisms.
- **Assessment Activities:**
  - **Documentation Review:** Verify that vendor documentation specifies which entities are authorised to access log records and describes access-control and role-based restrictions applied to logs. Confirm that documentation clearly differentiates privileged and non-privileged access.
  - **Log Access Configuration Inspection:** Inspect product configuration and access-control settings to identify the roles or identities permitted to access logs and the permissions granted to each role. Verify that default settings restrict log access to authorised entities only.
  - **Authorised Access Verification:** Authenticate as an authorised administrative or privileged user. Attempt to view log records, export or retrieve logs and/or perform permitted log-management actions. Verify that authorised access functions correctly and as documented.
  - **Unauthorised Access Testing:** Attempt to access logs using non-privileged user accounts and/or unauthenticated contexts (where applicable). Verify that access to logs is denied and no sensitive log content is disclosed.
  - **Management System Integration Validation:** Where logs are accessible via a management system or central console, verify that access is restricted to authorised systems and users and confirm that authentication and authorisation controls are enforced. Inspect logs or audit trails to confirm access attempts are recorded.
- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:
  - Access to stored log records is restricted to authorised entities only.
  - Authorisation mechanisms enforce role- or identity-based access control.
  - Unauthorised users or systems cannot read, export, or manage log records.
  - Log access behaviour is consistent, documented, and observable during testing.

  The assessment is **failed if any of the following occur**:
  - Log records are accessible to unauthorised users or systems.

- Access-control mechanisms are missing, inconsistently enforced, or they do not apply the defined access conditions.
- Default configuration exposes log data without appropriate authorisation.
- Log access behaviour is undocumented or unreliable.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting authorised and unauthorised log access attempts and observed enforcement of access controls.
  - Screenshots or tool outputs demonstrating restricted log access.
  - Configuration or policy extracts defining log access permissions.
  - Logs or audit records showing log access events.
  - Vendor documentation describing log access and authorisation mechanisms.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.14.5 REQ-MON-05

- **Assessment reference: REQ-MON-05**
- **Assessment Objective:**

  The objective of this assessment is to verify when the product's protection status changes, the authorised entities are notified. This includes confirming that:

  - The product detects cybersecurity-relevant events that lead to a change in the product's protection status, such as degradation of protection, temporary suspension of protection functions, restoration of normal protection following recovery and/or other status-related scenarios.
  - The product automatically notifies authorised entities when such status changes occur.
  - The product ensures that notifications are accurate, clearly informative to enable appropriate operational or security response and are generated as soon as the status change is identified.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server-based protection, cloud-managed service), a management interface or notification channel used by authorised entities (e.g. web management console, centralised management or monitoring system and/or local administrative interface). A controlled environment enabling generation of cybersecurity-relevant events that affect protection status.
  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings. Protection status indicators and notification mechanisms are enabled using default or recommended settings. One or more authorised entities (e.g. administrative users or management systems) are configured to receive notifications. Logging and monitoring mechanisms are active.
  - **Required Tools:** Incident or fault-injection tools to trigger protection status changes. Management console or administrative interface access. Notification inspection tools (e.g. dashboards, alerts, messages). Log and audit inspection tools.
  - **Reference Documentation:** Vendor documentation describing: protection status definitions and indicators and notification mechanisms and supported channels. Product architecture and incident-handling documentation.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation defines protection status states (e.g. normal, degraded, suspended) and describes events that trigger protection status changes. Confirm that notification behaviour and recipients are documented.

  - **Configuration and Authorisation Verification:** Inspect configuration to verify that notification of protection status changes is enabled by default or can be enabled by an authorised entity and only authorised entities can receive such notifications. Verify that notification configuration changes are auditable.

  - **Protection Status Change Simulation:** Trigger representative cybersecurity-relevant events that result in a change of protection status (e.g. failure or termination of a critical protection component, entry into a degraded protection state, recovery or restoration of full protection and/or other feasible actions). Observe system behaviour to confirm accurate detection of status changes.

  - **Notification Delivery Verification:** Verify that, upon a protection status change, notifications are generated and delivered to configured authorised entities. Confirm that notifications include clear indication of the protection status change and relevant contextual or explanatory information.

  - **Consistency and Reliability Checks:** Repeat selected scenarios to verify consistent notification behaviour across different types of protection status changes and different notification channels (where supported). Inspect logs to confirm notification events are recorded.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Cybersecurity-relevant events that change the product's protection status are reliably detected.

  - The product notifies authorised entities when such status changes occur.

  - Notifications are accurate, consistent with the observed protection state and generated as soon as the product status change is identified.

  - Notification behaviour is configurable, documented, and reliable.

  The assessment is **failed if any of the following occur**:

  - Protection status changes are not detected or are detected inconsistently.

  - Notifications are not generated or delivered when protection status changes occur.

  - Notifications are incomplete, misleading, or delayed without justification.

  - Notification mechanisms are undocumented, unreliable, or uncontrolled.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting the protection status change scenarios executed and the corresponding notification behaviour.

  - Screenshots or captures of notifications received by authorised entities.

  - Logs or audit records showing protection status changes and notification events.

  - Configuration extracts showing notification settings and authorised recipients.

  - Vendor documentation describing protection status and notification mechanisms.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.14.6 REQ-MON-06

- **Assessment reference: REQ-MON-06**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product allows an authorised user to disable any monitoring data transmission that is beyond what is strictly necessary for the intended purpose of the product. This includes confirming that:

  - The product clearly distinguishes between:
    - Monitoring data strictly necessary to perform its intended and documented cybersecurity functions.
    - Additional monitoring data collected or transmitted beyond that necessity (e.g. enhanced telemetry, analytics, usage statistics).
  - The product provides a user-accessible mechanism that allows an authorised user to disable the collection or transmission of such additional monitoring data.
  - The product ensures that disabling optional monitoring does not prevent the product from fulfilling its essential cybersecurity functions and does not weaken required security protections without appropriate user awareness.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server protection, cloud-managed agent). Network connectivity allowing observation of outbound data transmissions. Access to product configuration and management interfaces used by authorised users. Where applicable, both default and custom-configured deployments shall be assessed.
  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings. Monitoring and telemetry features are enabled using default or recommended configuration. The authorised user role (e.g. administrative user) is available. Network and logging tools are available to observe monitoring data flows.
  - **Required Tools:** Network traffic inspection and analysis tools. Product configuration and management interfaces. Log and audit inspection tools. Documentation review tools.
  - **Reference Documentation:** Vendor documentation describing monitoring and telemetry features, data categories collected and transmitted and configuration options for disabling optional monitoring. Product architecture and privacy or data-handling documentation.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to verify that monitoring data types are described and categorised and the distinction between necessary monitoring and optional or additional data is made. Confirm that documentation explains how an authorised user can disable optional monitoring.
  - **Identification of Monitoring Data:** Identify monitoring data recorded or transmitted by the product (e.g. operational telemetry, usage statistics and/or diagnostic or analytics data). Determine which data is strictly required for essential cybersecurity functions and which is optional.
  - **Configuration Mechanism Verification:** Inspect the product configuration to verify that a mechanism exists to disable optional monitoring and the mechanism is accessible only to authorised users. Verify that configuration changes are explicit, reversible, and auditable.
  - **Runtime Behaviour Observation:** Observe the product behaviour with the optional monitoring enabled and record the types and destinations of monitoring data transmitted. Disable optional monitoring using the authorised mechanism. Verify that additional monitoring data is no longer recorded or transmitted and essential cybersecurity functions continue to operate correctly.

- **Consistency and Transparency Checks:** Verify that disabling optional monitoring produces predictable and documented behaviour and does not silently re-enable monitoring without user action. Inspect logs or notifications to confirm visibility of configuration changes.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - Monitoring data beyond what is strictly necessary for the product's intended purpose is identifiable.
  - The product provides a mechanism allowing an authorised user to disable such additional monitoring.
  - Disabling optional monitoring does not impair essential cybersecurity functions.
  - Monitoring control behaviour is documented, reliable, and observable during testing.

  The assessment is **failed if any of the following occur**:

  - Optional or additional monitoring data cannot be distinguished from essential monitoring.
  - No mechanism exists to disable additional monitoring data collection or transmission.
  - Disabling monitoring is undocumented, unreliable, or not applied.
  - Additional monitoring continues despite being disabled by an authorised user.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting monitoring data flows before and after disabling optional monitoring.
  - Network captures or analysis outputs demonstrating reduced or halted transmission.
  - Screenshots or configuration extracts showing monitoring disablement mechanisms.
  - Logs or audit records reflecting configuration changes.
  - Vendor documentation describing monitoring features and user controls.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.15 Factory reset and data portability

6.15.1 REQ-FRD-01

- **Assessment reference: REQ-FRD-01**

- **Assessment Objective:**

  The objective of this assessment is to verify that the product notifies about the end of its support period. This includes confirming that:

  - The product clearly identifies and tracks the end of its declared support period.
  - The product notifies security/IT administrative and privileged users when the end of support is reached.
  - The product notifies end users, where technically feasible, in order to ensure awareness of the security implications.
  - The product enables information regarding continued use, upgrade, replacement, or migration of the product when the end of support period is reached.

- **Assessment Preparation**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server-based protection, cloud-managed service). Management interfaces or communication channels used to notify security/IT administrative users (e.g. management console, dashboards, alerts) and end users (e.g. local UI notifications), where applicable. A test configuration simulating or approaching the end of the product's declared support period. Where feasible, both centrally managed and standalone deployments shall be assessed.

  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings. The product's support period and end-of-support date are defined and documented by the manufacturer. Notification mechanisms (e.g. management console alerts, local UI notifications) are enabled using default or recommended settings. Administrative and privileged user accounts are configured. Logs and notification records are accessible.

  - **Required Tools:** Product management or administrative interfaces. Notification inspection tools (e.g. dashboards, alerts, system messages). Log and audit inspection tools. Configuration inspection utilities.

  - **Reference Documentation:** Vendor documentation describing product lifecycle and support periods and end-of-support communication mechanisms. Product security and lifecycle management documentation.

- **Assessment Activities:**

  - **Documentation Review:** Verify that vendor documentation defines the end of support period and describes how and when end-of-support notifications are issued. Confirm that notification targets (administrative users, privileged users, end users) are clearly identified.

  - **Configuration and Eligibility Verification:** Inspect configuration to verify that end-of-support notifications are enabled by default or cannot be unintentionally disabled and notification delivery mechanisms are correctly configured. Verify the existence of roles classified as security/IT administrative or privileged users.

  - **End-of-Support Notification Testing (Administrative Users):** Simulate or observe product behaviour at or near the declared end-of-support date. Verify that security/IT administrative and privileged users receive a notification and the notification clearly indicates that the product has reached end of support.

  - **End-User Notification Verification (Where Technically Feasible):** Where the product includes an end-user interface verify that end users are informed of end-of-support status where technically feasible. Confirm that end-user notifications are clear and distinguishable from routine messages and do not require administrative privileges to be visible.

  - **Content and Reliability Checks:** Verify that notifications include clear indication of end-of-support status and implications for security updates and protection. Confirm that notification events are logged and auditable.

- **Assignment of Verdict:**

  The assessment is **passed if all of the following occur**:

  - The product notifies security/IT administrative and privileged users when the end of the support period is reached.

  - End users are also informed where technically feasible.

  - Notifications are clear and reliably delivered and are sent when the end of support period is reached.

  - Notification behaviour is documented and observable.

  The assessment is **failed if any of the following occur**:

  - Administrative or privileged users are not notified of end-of-support status.

  - End-user notification is technically feasible but not implemented.

- Notifications are unclear, absent, or unreliable.
- End-of-support communication is undocumented or inconsistent.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting end-of-support notification behaviour.
  - Screenshots or captures of management console alerts and user notifications.
  - Logs or audit records showing end-of-support notification events.
  - Configuration extracts showing notification enablement.
  - Vendor documentation describing product lifecycle and end-of-support notification mechanisms.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

6.15.2 REQ-FRD-02

- **Assessment reference: REQ-FRD-02**
- **Assessment Objective:**

  The objective of this assessment is to verify that the product allows administrative users to manage the permanent removal of product data using a provided mechanism or following a set of instructions. This includes confirming that:

  - The product provides clear mechanisms and/or documented procedures that allow authorised administrative users to permanently remove product-related data.
  - The product ensures that such data removal covers all relevant product artefacts, including configuration data, product-generated files (e.g. quarantine data, temporary files) and log entries and audit records generated by the product.
  - The product enables secure and unambiguous decommissioning, uninstallation, or data sanitisation, reducing the risk of residual sensitive or security-relevant and product-related data remaining on the system.

- **Assessment Preparation:**

  - **Test Environment:** A fully installed and operational instance of the product in a representative deployment environment, including one or more supported deployment models of the product (e.g. endpoint agent, server-based protection, appliance, cloud-managed agent). Supported operating systems and storage environments where product data is stored. Access to installation, uninstallation, and administrative management interfaces. Where applicable, both standalone and centrally managed deployments shall be included.
  - **Preconditions:** The product is installed, configured, and operating normally using default or recommended settings, and has generated representative data, including configuration files and log entries. Authorised administrative user credentials are available. Product documentation related to installation, uninstallation, and data management is accessible.
  - **Required Tools:** Product administrative and management interfaces. Uninstallation or cleanup utilities provided by the vendor. File system and storage inspection tools. Log and configuration inspection utilities. Documentation review tools.
  - **Reference Documentation:** The assessment shall reference vendor-provided documentation describing data storage locations, uninstallation or cleanup procedures and data deletion or sanitisation mechanisms. Product architecture and lifecycle documentation.

- **Assessment Activities:**

  - **Documentation Review:** Review vendor documentation to verify that product data types (e.g. configuration, files, logs) are identified and instructions exist describing how authorised administrative users can permanently remove such data. Verify that documentation clearly distinguishes between temporary disablement and permanent data removal.

  - **Mechanism Availability Verification:** Inspect the product to verify the presence of built-in administrative mechanisms enabling permanent data removal or/and explicit uninstallation or cleanup options removing stored data. Verify that such mechanisms are accessible only to authorised administrative users.

  - **Data Removal Execution Testing:** Using provided mechanisms or documented procedures, perform controlled data removal actions. Verify that configuration data is removed, product-generated files (including quarantined files) are deleted or sanitised and log entries and audit data are removed as specified. Inspect the system to confirm the absence of residual product data.

  - **Authorisation and Protection Checks:** Attempt to invoke data-removal actions using unauthorised or non-privileged accounts. Verify that such actions are denied and only authorised administrative users can initiate permanent data removal.

  - **Completeness and Reliability Evaluation:** Verify that data removal procedures are complete and consistent across supported platforms and not dependent on undocumented manual steps. Confirm that failures or partial removal are detectable and reported.

- **Assignment of Verdict:**

  The assessment is passed if all of the following occur:

  - The product provides mechanisms or clear documented instructions for permanent removal of product data.

  - Removal covers configuration data, product-generated files, and log entries stored by the product.

  - Such actions are restricted to authorised administrative users.

  - Data removal is complete, reliable, and verifiable (e.g. no product-related data left in the host).

  The assessment is failed if any of the following occur:

  - No mechanism or instructions exist for permanent removal of product data.

  - Removal procedures are incomplete, unclear, or undocumented.

  - Non-privileged users can trigger permanent data removal.

  - Residual product data remains after documented removal procedures.

- **Supporting Evidence:**

  The following evidence shall be collected and attached to the assessment report:

  - Test or assessment reports documenting data removal procedures executed and post-removal verification results.

  - Screenshots or console outputs showing administrative data-removal actions.

  - File system or storage inspection outputs confirming removal of data.

  - Configuration extracts or logs indicating execution of data-removal mechanisms.

  - Vendor documentation describing data management and removal procedures.

  The assessment shall be considered incomplete if the supporting evidence is missing, inconsistent, or insufficient to independently verify the complete fulfilment of the requirement.

Annex A (informative): Relationship between the present document and the essential cybersecurity requirements of EU Regulation (EU) 2024/2847

The present document has been prepared under the Commission's standardisation request C(2025) 618 final [i.3] to provide one voluntary means of conforming to the requirements of Regulation (EU) 2024/2847 of the European Parliament and of the Council of 23 October 2024 on horizontal cybersecurity requirements for products with digital elements and amending Regulations (EU) 168/2013 and (EU) 2019/1020 and Directive (EU) 2020/1828 (Cyber Resilience Act) (CRA) [i.1].

Once the present document is cited in the Official Journal of the European Union under that Regulation, compliance with the normative clauses of the present document given in Table A.1 confers, within the limits of the scope of the present document, a presumption of conformity with the corresponding requirements of that Regulation and associated EFTA regulations.

Table A.1: Relationship between the present document and the requirements of Regulation (EU) 2024/2847 - the Cyber Resilience Act

Description	Essential Requirements of Regulation (EU) 2024/2847	Clause(s) of the present document	Remarks / Notes
Annex I, Part I, (1)	"Products with digital elements shall be designed, developed and produced in such a way that they ensure an appropriate level of cybersecurity based on the risks."	Clauses 5 and 6 Annex R: REQ-RDPS-L-AUTH-001 to -003, REQ-RDPS-L-INTEG-001 to -003, REQ-RDPS-L-CONF-001 to -003, REQ-RDPS-L-AVAIL-001 to -003, and the corresponding RDPS-side requirements REQ-RDPS-R-AUTH-001 to -003, REQ-RDPS-R-INTEG-001 to -003, REQ-RDPS-R-CONF-001 to -003, REQ-RDPS-R-AVAIL-001 to -003.	
Annex I, Part I, (2)(a)	"Products with digital elements shall be made available on the market without known exploitable vulnerabilities."	Clauses 5.3 and 6.3	• REQ-KEV-01 has no conditions.
Annex I, Part I, (2)(b)	"Products with digital elements shall be made available on the market with a secure by default configuration, unless otherwise agreed between manufacturer and business user in relation to a tailor-made product with digital elements, including the possibility to reset the product to its original state."	Clauses 5.4 and 6.4	• REQ-SBD-01/…/04 have no conditions. • REQ-SBD-05 is conditioned to "Where the product makes use of and RDPS feature, …"

Description	Essential Requirements of Regulation (EU) 2024/2847	Clause(s) of the present document	Remarks / Notes
Annex I, Part I, (2)(c)	"Products with digital elements shall ensure that vulnerabilities can be addressed through security updates, including, where applicable, through automatic security updates that are installed within an appropriate timeframe enabled as a default setting, with a clear and easy-to-use opt-out mechanism, through the notification of available updates to users, and the option to temporarily postpone them."	Clauses 5.5 and 6.5	• REQ-SU-01/03 have no conditions. • REQ-SU-02/04 are conditioned to "Where the product supports automatic cybersecurity updates, …" • REQ-SU-05 is conditioned to "…except where a controlled rollback to a previously verified and trusted version is performed by the product." • REQ-SU-06 is conditioned to "…in case of failed or interrupted update…"
Annex I, Part I, (2)(d)	"Products with digital elements shall ensure protection from unauthorised access by appropriate control mechanisms, including but not limited to authentication, identity or access management systems, and report on possible unauthorised access."	Clauses 5.6 and 6.6 Annex R: REQ-RDPS-L-AUTH-001 to -003 and REQ-RDPS-R-AUTH-001 to -003.	• REQ-AAC-01/04/11 have no conditions. • REQ-AAC-02 is conditioned to "… where available, …" • REQ-AAC-03 is conditioned to "… where technically feasible." • REQ-SU-05/…/10 is conditioned to "Where the product offers a web management console, …"
Annex I, Part 1, (2)(e)	"Products with digital elements shall protect the confidentiality of stored, transmitted or otherwise processed data, personal or other, such as by encrypting relevant data at rest or in transit by best practice mechanisms, and by using other technical means."	Clauses 5.7 and 6.7 Annex R on Confidentiality: REQ-RDPS-L-CONF-001 to -003 and REQ-RDPS-R-CONF-001 to -003.	• REQ-CON-01 have no condition. • REQ-CON-02 is conditioned to "Where the product stores authentication credentials or cryptographic private keys, …"
Annex I, Part I, (2)(f)	"Products with digital elements shall protect the integrity of stored, transmitted or otherwise processed data, personal or other, commands, programs and configuration against any manipulation or modification not authorised by the user, and report on corruptions."	Clauses 5.8 and 6.8 Annex R on Integrity: REQ-RDPS-L-INTEG-001 to -003 and REQ-RDPS-R-INTEG-001 to -003.	• REQ-INT-01 have no condition. • REQ-INT-02 is conditioned to "Where the product transmits data, …"
Annex I, Part I, (2)(g)	"Products with digital elements shall process only data, personal or other, that are adequate, relevant and limited to what is necessary in relation to the intended purpose of the product with digital elements (data minimisation)."	Clauses 5.9 and 6.9	• REQ-DM-01 has no condition.

Description	Essential Requirements of Regulation (EU) 2024/2847	Clause(s) of the present document	Remarks / Notes
Annex I, Part I, (2)(h)	"Products with digital elements shall protect the availability of essential and basic functions, also after an incident, including through resilience and mitigation measures against denial-of-service attacks."	Clauses 5.10 and 6.10 Annex R: REQ-RDPS-L-AVAIL-001 to -003 and REQ-RDPS-R-AVAIL-001 to -003.	• REQ-AP-01/02 have no condition. • REQ-AP-03 is conditioned to "Where the product controls externally accessible interfaces (e.g. interfaces that are not controlled by the OS) used to support its documented essential cybersecurity functions, …"
Annex I, Part I, (2)(i)	"Products with digital elements shall minimise the negative impact by the products themselves or connected products on the availability of services provided by other products or networks."	Clauses 5.11 and 6.11	• REQ-NI-01/02 have no condition.
Annex I, Part I, (2)(j)	"Products with digital elements shall be designed, developed and produced to limit attack surfaces, including external interfaces."	Clauses 5.12 and 6.12 Annex R: REQ-RDPS-L-AUTH-001 to -003, REQ-RDPS-R-AUTH-001 to -003	• REQ-ASM-01 have no condition. • REQ-ASM-02 is conditioned to "Where the product exposes local or network services for integration with other components or the underlying OS, …" • REQ-ASM-03 is conditioned to "Where the product provides local or network services for optional integration with other components or the underlying OS, …"
Annex I, Part I, (2)(k)	"Products with digital elements shall be designed, developed and produced to reduce the impact of an incident using appropriate exploitation mitigation mechanisms and techniques."	Clauses 5.13 and 6.13 Annex R: REQ-RDPS-L-AVAIL-002 to -003 and REQ-RDPS-R-AVAIL-002 to -003.	• REQ-EM-01 is conditioned to "When enabled by the administrator, …" • REQ-EM-02 is conditioned to "…, and where prevention is not technically feasible, …"

Description	Essential Requirements of Regulation (EU) 2024/2847	Clause(s) of the present document	Remarks / Notes
Annex I, Part I, (2)(l)	"Products with digital elements shall provide security related information by recording and monitoring relevant internal activity, including the access to or modification of data, services or functions, with an opt-out mechanism for the user."	Clauses 5.14 and 6.14	• REQ-MON-01 is conditioned to "…, where technically feasible, …" • REQ-MON-02/03/04 is conditioned to "Where the product creates and stores logs, …" • REQ-MON-05 has no condition. • REQ-MON-06 is conditioned to "Where the product records or transmits monitoring data beyond what is strictly necessary for its intended purpose, …"
Annex I, Part I, (2)(m)	"Products with digital elements shall provide the possibility for users to securely and easily remove on a permanent basis all data and settings and, where such data can be transferred to other products or systems, ensure that this is done in a secure manner."	Clauses 5.15 and 6.15	• REQ-FRD-01 is conditioned to "If technically feasible, …" • REQ-FRD-02 has no condition.
Annex I, Part II	Vulnerability Handling Requirements	N/A	• The present document scope does not cover this topic, refer to [i.6] for the associated requirements, their conditionality and assessment.

Key to columns:

Description A textual reference to the requirement.

Requirements of Regulation

Identification of point(s) defining the requirement in Regulation (EU) 2024/2847 - the Cyber Resilience Act.

Clause(s) of the present document

Identification of clause(s) addressing the requirement in the present document

Presumption of conformity stays valid only as long as a reference to the present document is maintained in the list published in the Official Journal of the European Union. Users of the present document should consult frequently the latest list published in the Official Journal of the European Union.

Other Union legislation may be applicable to the product(s) falling within the scope of the present document.

Annex B (informative): Security analysis

B.0 Introduction

This annex applies state of the art methodology to identify assets, threats, identify and evaluate risk factors, and define security profiles applicable to the different use cases identified in the product context.

B.1 Threats and Risk Factors Landscape

B.1.1 Threats

B.1.1.1 Introduction

The following cybersecurity threats have been identified as relevant for antivirus and antimalware software product. Manufacturers need to assess, mitigate, and validate these threats to ensure compliance with the present document based on the identified threats below and the cybersecurity requirements in clause 5.

The identified threats are grouped into categories addressing update integrity, self-protection, communication, supply chain dependencies, advanced attack techniques, and user interaction.

B.1.1.2 TH1 - Threats related to Updates and Signatures

TH1.1: Tampered Signature Database: An attacker modifies or replaces malware signature files to disable detection or trigger false detections.

TH1.2: Compromised Update Packages: Attackers distribute malicious or altered updates, for example through MITM attacks or compromised servers.

TH1.3: Rollback Attacks: Forcing the product to install an outdated and vulnerable version of the product databases or components.

TH1.4: Faulty Updates: A non-malicious signature update or code update may prevent the machine from performing normally, e.g. by preventing booting and/or by preventing installation of an update remediating the situation.

TH1.5: False-positive Operating System Update: A non-malicious operating system update may trigger a false positive detection by the antivirus or antimalware and thus prevent the machine from performing normally, e.g. by preventing booting and/or by preventing installation of an update remediating the situation.

TH1.6: Malicious Denylisting Compromising Availability: Malicious denylisting of legitimate files within the product or its upstream threat intelligence feed, introduced intentionally by a vendor or indirectly through a compromised or manipulated feed source with the goal of preventing the execution of such files.

B.1.1.3 TH2 - Threats to Core Components and Self-Protection

TH2.1: Disabling or Bypassing Self-Protection Modules: Malware or attackers attempt to disrupt the product self-protection mechanisms for example by terminating processes, altering configuration files, or unloading critical drivers of the product.

TH2.2: Tampering with Quarantine Mechanisms: Malicious code attempts to execute an actions against the quarantined files, for example restore them or execute them despite isolation.

TH2.3: Privilege Escalation on Product Services: Exploiting flaws in the product components to gain elevated privileges, for example kernel-level exploitation of scanning drivers.

TH2.4: Tampering with or Disabling Logging Mechanisms: The malware intents to affect the logging mechanism in disruptive manner, for example by limiting the creation of logs and/or modifying the stored product logs such as detection events logs, update logs, etc.

TH2.5: Incompatible OS Update: A non-malicious (and non-faulty) OS update causes parts or all of the antimalware or antivirus functionality to be ineffective or disabled.

NOTE: The antimalware product may rely on undocumented or unintended-for-external-usage OS behaviour and the OS vendor changes said behaviour.

B.1.1.4 TH3 - Threats to Communication and Data Exchange

TH3.1: MITM Attacks on Cloud Connections: Interception or injection of data between product and its secure cloud backend.

TH3.2: Compromised Threat Intelligence Feeds: Malicious or spoofed intelligence data leading to false decisions, for example whitelisting malicious files.

TH3.3: Leakage of Sensitive Data: Improper handling of logs, telemetry, and/or threat reports or the existence of vulnerabilities in the product components leading to unauthorised disclosure of personal data or sensitive system data.

B.1.1.5 TH4 - Supply Chain and Third-Party Component Threats

TH4.1: Malicious Third-Party Libraries: Insertion of vulnerable or intentionally compromised third-party components, either open-source or not, in the product.

TH4.2: Trojanised Distribution Channels: Attackers compromise the vendor's distribution platform to deliver rogue updates.

TH4.3: SBOM Manipulation: Inaccurate or incomplete SBOM leading to wrongly assessed or unassessed dependencies

B.1.1.6 TH5 - Advanced Threat Actor Techniques

TH5.1: Zero-Day Exploitation of the Product: Exploiting not-known or unpatched vulnerabilities in the product components.

TH5.2: Evasion Attacks: Malware designed to avoid detection, for example by employing mechanisms such as polymorphism, metamorphism, and/or fileless execution.

TH5.3: Insider Threats: Malicious use of privileged access to disable, misconfigure, or backdoor the product.

TH5.4: Malicious Denylisting Compromising Confidentiality: Malicious denylisting of legitimate files within the product or its upstream threat intelligence feed, introduced intentionally by a vendor or indirectly through a compromised or manipulated feed source with the goal of getting telemetry on the presence of files and/or the exfiltration of such files.

B.1.1.7 TH6 - User-Facing Threats

TH6.1: Social Engineering to Breach Protection: Attackers trick users into turning off product updates, real-time protection, or whitelisting malicious files.

TH6.2: Misleading Notifications: Exploited UI messages prompting users to perform unsafe actions.

B.1.2 Risks Factors

B.1.2.1 Introduction

As an antivirus/anti-malware is a developed software that aims to access an organization's information or resources to trigger more dangerous situations, the following Risks Factors (RF) have been identified to define possible angles from where an organization may be attacked to overpass a virus/anti-malware. The RFs are classified according to their main influence on likelihood (RFL) or magnitude (RFM).

B.1.2.2 Likelihood-related Risk Factors

- **RFL1 - Threat exposure surface**

All logical, wireless, and physical pathways an attacker can utilise to reach the target system. In the context of automated and connected vehicles, the exposure surface is dominated by remote, non-physical interfaces. More channels and protocols mean more attack opportunities and detection blind spots, especially since the methods to deliver malware has passed evolved from "file centric" injection towards direct injection.

- **RFL2 - Insider threat**

This factor addresses the risks posed by malicious or negligent insiders, who possess or can obtain privileged access and create shadow changes or exfiltrate data. Likelihood correlates with the number of privileged users, weak separation of duties, standing privileges instead of just-in-time access, permissive exception processes, limited monitoring and deterrence, and extensive third-party access.

- **RFL3 - User security awareness**

Users with lower security awareness may tend to neglect the security activities required to maintain the good security posture of the system, and/or antimalware solution. If the management and security of the product or the systems covered by the product is not actively maintained within the organization by the team, or department, the users without security knowledge may increase a risk associated with improper management or configuration of the product. In this risk, the aspects to be considered to assign a risk likelihood to happen need to vary on whether the affected product maintenance is done by a dedicated IT team, and individual team with certain experience or just by a final user with no experience.

- **RFL4 - Connectivity posture**

Where devices connect (e.g. segmented LAN, internet, roaming Wi-Fi®, VPNs) and how often. Untrusted or intermittent links complicate updates, telemetry, and command-and-control blocking. Network isolation, DNS filtering, and posture checks reduce risk. It is of especial interest the use of public unprotected wireless networks, which should be treated as a hostile environment. When a laptop or phone connects to an public unprotected wireless network, it is key to assume that anything sent or received could be monitored or tampered with. Conversely, private networks that are actively protected (e.g. through network segmentation and secure wireless configurations) inherently reduce risk due to their restricted access and closed architecture. In this risk, the aspects to be considered to assign a risk likelihood to happen need to vary on whether the protected system is connected to the internet through a public access network (e.g. bar/restaurant Wi-Fi), a private personal network (e.g. home or personal business Wi-Fi®) or a private company network (i.e. limited access for external users to access it).

- **RFL5 - Integration/API exposure**

The product may not operate in isolation and may instead be integrated into a broader cybersecurity ecosystem. This factor evaluates the attack surface introduced by these connections, including management consoles, APIs (e.g. SIEM, EDR, SOAR), and potential RDPS usage. The likelihood of a successful exploit correlates directly with the volume of integrations and the exposure of public endpoints. Furthermore, the probability of compromise is significantly elevated by weak access controls, specifically the absence of best practices for encryption in transit, authentication, and authorisation.

B.1.2.3 Magnitude-related Risk Factors

- **RFM1 - Data sensitivity**

The type of information an organization handles directly influences its exposure to cyber threats. Data sets that include personal data (e.g. full identifiers, account numbers, national IDs, etc.) are more attractive to be targeted and carry greater potential harm if compromised than basic contact lists containing only names and mailing addresses. Understanding the sensitivity that the managed data requires is key and classifying data according to its business value and potential impact if lost, altered, or disclosed, will allow to identify the necessary protection. In this risk, the aspects to be considered to assign the magnitude of loss or disruption need to vary on the whether protected system is within the organization premises or not, as in many organizations together with the products there are other products increasing the cybersecurity of the system. In summary, higher sensitivity demands stronger encryption, access control, and retention limits.

- **RFM2 - Availability criticality**

This factor addresses how much disruption the environment can tolerate, including disruption introduced by the product itself (e.g. scans, updates, quarantines). In high-availability contexts, control side effects can rival attack damage. Magnitude grows with complex dependencies, frequent change, external exposure, resource contention, and weak operational discipline, SLA penalties, cost per hour of downtime, safety or process interruptions, cascading failures to connected systems, and misalignment between actual MTTD/MTTR and defined RTOs.

- **RFM3 - Update channel trust & timeliness**

Integrity and freshness of engines, signatures, and patches (online, staged, or offline). Recently, antivirus and anti-malware software depend on RDPS such as remotely accessible databases that are being constantly updated with the latest signatures or other types of identification elements that allow to identify possible viruses and malwares when analysing a new file arriving to the protected system by the product. To this end, it is key to protect the communication between the product and the systems managing the delivery of the latest stored and/or processed data because this communication channel and the data exchanged through it can be an exploitable attack surface if not well protected.

- **RFM4 - Supply-chain & third-party dependencies**

This risk may be inherited from third-party components, including vendors, SDKs, application delivery mechanisms, drivers, firmware, and external content feeds. A compromise in any of these dependencies may enable attackers to bypass local security controls [i.7]. To mitigate such risks, suppliers should be carefully vetted, and the use of code signing, SBOMs, and integrity verification mechanism should be enforced. Additionally, third-party components should be sandboxed or otherwise constrained to limit their potential impact.

Furthermore, with the continuous growth of mobile communication devices and the widespread use of application distribution platforms, the source from which software associated with the product is obtained may rely on organizations different from the original developer. In this context, it is key that product vendors ensure their software is distributed through legitimate channels (e.g. most common application stores), so that it reaches end users in the most secure manner possible.

- **RFM5 - Tamper surface**

This factor addresses the ways in which an attacker may disable or weaken protections, including but not limited to terminating services, altering policies or signature databases, bypassing OS-level protections, forging or deleting logs, or exploiting low-integrity drivers. For products operating in consumer or unmanaged environments, magnitude reflects the window of unprotected operation and the resilience of the product's self-protection mechanisms against malware. For managed and/or centralised environments, the associated impact reflects the duration of reduced or absent protection, the opportunity for undetected lateral movement, degradation of forensic integrity, and the compliance risks arising from incomplete, missing or untrustworthy records.

- **RFM6 - Recovery capability**

Due to the increase on specific attacks such as ransomware, where access to the data is blocked, the capacity of an entity or actor to recover its own data is key. This factor captures the quality of product backups, related dependencies and binaries, and considers the tested ability to restore the product to a known-good state. Magnitude is reflected the product's internal failsafe mechanisms.

- **RFM7 - Change rate & configuration stability**

The pace of software and configuration change becomes important as updates can inadvertently introduce instability. However, for antivirus products where rapid, continuous updates are essential to functionality, a high change rate is expected. Magnitude depends on the risk of updates causing local system instability, outages, or temporary loss of protection, rather than enterprise metrics such as policy misconfiguration propagation or compliance drift at audit. To support this high change rate safely, the manufacturer bears the responsibility of enforcing strict integrity controls within their delivery channels (including CI/CD pipelines) to prevent the distribution of compromised or unstable updates. Mitigations should focus on these internal delivery controls, alongside safe rollout practices and automated rollback capabilities.

B.1.3 Threat justification and mitigation

Table B.1 maps each threat to the RFs that affect its impact and likelihood, and to the requirements that mitigate it:

Table B.1: Threat justification and mitigation

ID	Title	Contributing RFs	Requirement(s) for mitigation
[TH1.1]	**Tampered Signature Database**	RFL1, RFL4, RFL5; RFM1, RFM3, RFM5, RFM7	REQ-INT-01, REQ-INT-02, REQ-SU-01, REQ-SU-03, REQ-SU-05, REQ-SU-06, REQ-AP-01
[TH1.2]	**Compromised Update Packages**	RFL1, RFL4, RFL5; RFM1, RFM3, RFM4, RFM5, RFM7	REQ-SU-01, REQ-SU-02, REQ-SU-03, REQ-SU-04, REQ-SU-05, REQ-SU-06, REQ-INT-01, REQ-INT-02, REQ-AP-01, REQ-AP-03
[TH1.3]	**Rollback Attacks**	RFL1, RFL4, RFL5; RFM3, RFM5, RFM7	REQ-SU-05, REQ-SU-06, REQ-INT-01, REQ-AP-01, REQ-IM-01
[TH1.4]	**Faulty Updates**	RFL1, RFL4, RFL5, RFM2, RFM3, RFM5, RFM6, RFM7	REQ-SU-06, REQ-SU-07, REQ-AP-01, REQ-AP-02, REQ-NI-01, REQ-NI-02
[TH1.5]	**False-positive Operating System Update**	RFL5, RFM2, RFM3, RFM4, RFM6, RFM7	REQ-SU-01, REQ-SU-05, REQ-SU-07, REQ-SU-08, REQ-AP-02, REQ-AP-04, REQ-INT-01, REQ-AAC-02, REQ-AAC-04, REQ-NI-01
[TH1.6]	**Malicious Denylisting Compromising Availability**	RFL5, RFM2, RFM3, RFM4, RFM6, RFM7	REQ-AP-02, REQ-AP-03, REQ-AP-04, REQ-SU-01, REQ-SU-02, REQ-AAC-10, REQ-AAC-11, REQ-MON-01, REQ-MON-05
[TH2.1]	**Disabling or Bypassing Protection Modules**	RFL1, RFL2, RFL5; RFM2, RFM5, RFM6	REQ-ASM-01, REQ-AP-01, REQ-AP-02, REQ-EM-02, REQ-MON-02, REQ-AAC-09
[TH2.2]	**Tampering with Quarantine Mechanisms**	RFL1, RFL2, RFL5; RFM1, RFM5	REQ-ASM-01, REQ-MON-02, REQ-MON-03, REQ-AAC-01, REQ-AAC-09
[TH2.3]	**Privilege Escalation on Product Services**	RFL1, RFL2, RFL5; RFM2, RFM5	REQ-ASM-01, REQ-ASM-02, REQ-AAC-01, REQ-AAC-04, REQ-AAC-05, REQ-AAC-06, REQ-AAC-08, REQ-AAC-09
[TH2.4]	**Tampering with or Disabling Logging Mechanisms**	RFL1, RFL2, RFL5; RFM5, RFM6	REQ-MON-02, REQ-MON-03, REQ-MON-04, REQ-MON-06, REQ-AAC-07
[TH2.5]	**Incompatible OS Update**	RFL1, RFL3, RFL3, RFL4, RFL5, RFM2, RFM3, RFM4, RFM5, RFM7	REQ-SU-01, REQ-SU-02, REQ-SU-03, REQ-SU-04, REQ-SU-05, REQ-SU-06, REQ-SU-07, REQ-SU-09, REQ-AP-01, REQ-AP-02, REQ-NI-01, REQ-ASM-01, REQ-EM-02, REQ-AAC-02, REQ-AAC-04
[TH3.1]	**MITM Attacks on Cloud Connections**	RFL1, RFL4, RFL5; RFM1, RFM3, RFM5	REQ-CON-01, REQ-INT-02, REQ-SU-01, REQ-SU-03, REQ-SBD-05, REQ-AAC-07
[TH3.2]	**Compromised Threat Intelligence Feeds**	RFL1, RFL4, RFL5; RFM1, RFM3, RFM4	REQ-CON-01, REQ-INT-01, REQ-SU-01, REQ-SU-03, REQ-AP-03
[TH3.3]	**Leakage of Sensitive Data**	RFL1, RFL2, RFL4, RFL5; RFM1, RFM4	REQ-CON-01, REQ-CON-02, REQ-DM-01, REQ-MON-01, REQ-MON-04, REQ-MON-06, REQ-FRD-02, REQ-AAC-07
[TH4.1]	**Malicious Third-Party Libraries**	RFL1, RFL4, RFL5; RFM4, RFM5	REQ-KEV-01, REQ-SU-01, REQ-ASM-01, REQ-AP-01
[TH4.2]	**Trojanised Distribution Channels**	RFL1, RFL4, RFL5; RFM3, RFM4, RFM7	REQ-KEV-01, REQ-SU-01, REQ-SU-05, REQ-SU-06, REQ-INT-01, REQ-AP-01, REQ-SBD-05

ID	Title	Contributing RFs	Requirement(s) for mitigation
[TH4.3]	**SBOM Manipulation**	RFL4, RFL5; RFM4, RFM5	REQ-KEV-01, REQ-SU-03, REQ-MON-01
[TH5.1]	**Zero-Day Exploitation of the Product**	RFL1, RFL2, RFL5; RFM2, RFM5, RFM6	REQ-SBD-02, REQ-SBD-03, REQ-SBD-04, REQ-EM-02, REQ-AP-02, REQ-MON-01, REQ-MON-05, REQ-IM-02, REQ-ASM-03, REQ-AAC-09
[TH5.2]	**Evasion Attacks**	RFL1, RFL5; RFM5, RFM6	REQ-AAC-01, REQ-AAC-09, REQ-SU-01, REQ-MON-01, REQ-MON-05, REQ-AP-02
[TH5.3]	**Insider Threats**	RFL2, RFL5; RFM1, RFM5	REQ-AAC-02, REQ-AAC-03, REQ-AAC-04, REQ-AAC-05, REQ-AAC-06, REQ-AAC-08, REQ-AAC-09, REQ-AAC-11, REQ-MON-03, REQ-MON-04, REQ-CON-02, REQ-ASM-01, REQ-ASM-03, REQ-FRD-02
[TH5.4]	**Malicious Denylisting Compromising Confidentiality**	RFL5, RFM1, RFM3, RFM4, RFM7	REQ-SU-01, REQ-AAC-02, REQ-AAC-04, REQ-CON-01, REQ-CON-02, REQ-CON-03, REQ-MON-01, REQ-INT-01
[TH6.1]	**Social Engineering to Disable Protection**	RFL3, RFL5; RFM1, RFM5	REQ-SBD-01, REQ-AAC-02, REQ-AAC-03, REQ-AAC-05, REQ-AAC-06, REQ-AAC-08, REQ-AAC-09, REQ-AAC-10, REQ-MON-05, REQ-EM-01, REQ-FRD-01
[TH6.2]	**Misleading Notifications**	RFL3, RFL5; RFM5, RFM6	REQ-MON-01, REQ-MON-05, REQ-EM-01, REQ-SBD-01, REQ-AAC-07, REQ-FRD-01

B.2 Security Profiles

B.2.1 General

The present document establishes three cumulative SPs so that CRA [i.1] cybersecurity obligations can be applied proportionately to risk. These SPs define the applicable requirements depending on the product's intended use.

The proposed SPs are not isolated one from each other, but they have dependencies among them as illustrated in Figure B.1. Minimal Cybersecurity allows to achieve a basic set of cybersecurity needs, Medium Cybersecurity considers the needs defined in the Minimal Cybersecurity and extends them with updates or new needs, High Cybersecurity does the same by enhancing the needs from Medium Cybersecurity and considering more demanding needs.

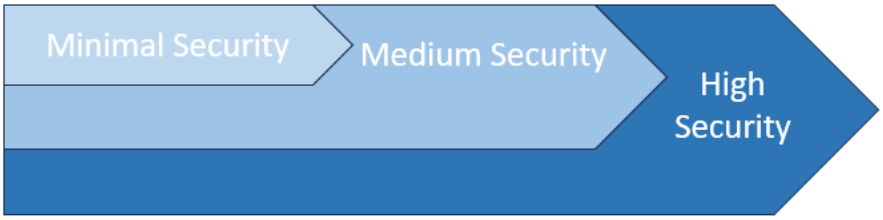

Figure B.1: SPs organization

B.2.2 Minimal Cybersecurity

Minimal Cybersecurity represents a baseline level of cybersecurity robustness applicable to the product, when it is characterised by limited exposure, integration complexity and privilege escalation potential.

At this level, the focus is on ensuring that the product adequately protects its CIA, including secure handling of updates, protection of privileged components, and enforcement of fundamental access control and integrity mechanisms.

B.2.3 Medium Cybersecurity

Medium Cybersecurity represents an increased level of cybersecurity robustness applicable to the product, when it is characterised by greater exposure, broader integration with external systems, or elevated privilege levels compared to Minimal Cybersecurity.

At this level, the product is expected to address more complex interaction patterns and dependency relationships, including enhanced protection of administrative interfaces, strengthened integrity safeguards, and increased resilience against manipulation or misuse in managed or interconnected contexts.

B.2.4 High Cybersecurity

High Cybersecurity represents the highest level of cybersecurity robustness applicable to the product when it is characterised by extensive integration dependencies, high privilege exposure, or significant potential impact in the event of compromise.

At this level, compromise of the product could result in substantial operational, financial, or societal consequences. Greater emphasis is therefore placed on strong protection of privileged components, integrity of inter-system communications, resilience against sophisticated misuse scenarios, and robustness under complex or high-dependency operating conditions.

B.3 Information on the methodology for the security analysis of cybersecurity risks used to develop the present document

B.3.1 Principles and references

This annex describes the security analysis methodology applied during drafting of the present document and provides a repeatable approach that manufacturers may adopt to determine cybersecurity risks for antivirus/antimalware software product. This methodology determines the residual cybersecurity risk of the product by combining qualitative judgments with numerical scoring on two axes (i.e. Likelihood (L) and Magnitude (M)) and then mapping results to SPs for proportionate requirements application.

This semi-quantitative approach meets the conditions established in the draft:

- it uses likelihood and magnitude;
- supports qualitative matrices and numeric scoring;
- allows alternative analysis techniques (e.g. fault trees) upstream of Likelihood/Magnitude;
- and keeps results consistent with the use-case and SP mapping.

The approach is consistent with widely used risk management practices (e.g. [i.14]), adapted to the specific scope and terminology of the present document:

- It distinguishes inherent and residual risk.
- It evaluates likelihood and magnitude using calibrated qualitative scales mapped to numeric values.
- It considers the draft's threat catalogue and RFs (clause B.1.2), UCs (clause 4.6), and SPs (clause B.2).
- It supports traceability into Annex A mapping to CRA [i.1] essential requirements.

B.3.2 Risk model

The proposed risk model defines the objects of assessment and their relationships (i.e. context) to compute the risk estimation. It establishes a coherent scope and trust boundaries prior to likelihood and impact determination. To apply the presented methodology, the risk model is based on the following elements:

- **Object of assessment:** The elements of value that the product relies on to operate safely. An "asset" is anything whose compromise (CIA, or lawful use) could lead to loss or disruption for the user, the operator, or the ecosystem. Based on this, in this methodology the product architectural components in clause 4.3 were considered (i.e. signature/heuristic engines, update/signature pipeline, quarantine store, telemetry/logs, etc.), together with the operational context (clause 4.4) and the type 2of users (clause 4.6).
- **Threats and RFs:** As listed in clause B.1.
- **Controls:** Requirements listed in clause 5.
- **Contexts:** The declared UC by the product vendor.

B.3.3 Risk criteria

B.3.3.1 Introduction

This clause sets the risk criteria and scales used by the methodology, including qualitative categories and their numeric mappings, the risk matrix, and aggregation rules. These criteria enable consistent and comparable estimation across use cases and product variants.

B.3.3.2 Risk Likelihood (L)

L expresses the estimated chance that a given risk factor for a UC will occur to avoid the product's operational objective. Table B.2 presents the proposed mapping between identified categories and their values.

Table B.2: L qualitative categories and numeric mapping

Category	Description (indicative frequency) *	Value
Rare	Threat requires multiple preconditions; strong controls; limited exposure.	1
Unlikely	Feasible but constrained by environment or controls.	2
Possible	Routinely observed in similar deployments.	3
Likely	Common in sector; known active adversaries.	4
Almost certain	Continuously attempted in large populations.	5

B.3.3.3 Magnitude of loss or disruption (M)

M expresses the worst credible level of adverse consequence if the scenario occurs. Assess impact across CIA, and compliance/reputation, and take the highest applicable dimension. Consider blast radius, persistence/duration of effects, propagation potential, impact on safety/critical services, legal/reporting obligations, and recovery effort/cost. Table B.3 presents the proposed mapping between identified categories and their values.

Table B.3: M qualitative categories and numeric mapping

Category	Illustrative consequence for product / environment	Value
Negligible	No data/ops impact and no material loss; transient, auto-recoverable.	1
Minor	Localised user disruption; recoverable quickly; user inconvenience; no data at risk	2
Moderate	Service interruption or privacy event requiring incident handling	3
Major	Material service outage, significant data exposure, or regulatory notification, quarantines/updates disabled.	4
Severe	Safety/public-service impact, widespread outage, o large-scale data compromise; legal/reporting impact.	5

B.3.3.4 Qualitative risk matrix

The qualitative risk matrix maps the L of a threat scenario (rows) against the M (columns) and yields a risk category: Low (L), Moderate (M), Significant (S), High (H), or Very High (VH). It provides a fast, defensible determination before (or alongside) any quantitative scoring as the risk category is determined by plotting (L, M). Manufacturers should adopt thresholds no less stringent than the following baseline.

Table B.4: Qualitative risk matrix

	M=1	M=2	M=3	M=4	M=5
L=1	L	L	M	M	S
L=2	L	L	M	S	H
L=3	L	M	S	H	H
L=4	M	M	S	H	VH
L=5	M	S	H	VH	VH

B.3.4 Process

The following lines provide the stepwise process applied of the assessment methodology from scoping to review by specifying the required inputs, activities, and outputs for each step to ensure repeatability, traceability, and decision transparency. The steps of the implemented methodology are described as follows:

1) **Object of Assessment (OoA) and Context Definition:** The OoA is the product subsystems and its deployment environment (i.e. functional modules and operational context) and the selected applicable UC.

2) **Using the identified threats:** Using the identified threats in clause B.1, the attack surfaces can be detected and the associated RFs identified.

3) **RFs Likelihood (L) and Magnitude (M):** The identified RFs in clause B.1 need to be analysed, one by one, to identify how they affect the UC based on the selected threat in terms of likelihood and magnitude. The outcome of this analysis are the L and M values assigned.

4) **Inherent Risk Computation:** To obtain the inherent risk, the mathematical formulas defined in clause B.3.5 need to be used to compute the Inherent Risk value associated to each threat.

5) **Individual SP classification:** For each individual Inherent Risk value (i.e. one per selected threat), an SP is mapped based in clause B.3.5.

6) **Aggregation of individual SPs to UC SP:** By computing the mean value of all the different computed individual SPs (i.e. one for each selected threat) the overall SP for the UC is computed and with the obtained value, the mapping to the corresponding final SP can be done. By identifying the SP for each UC, it was possible to create the mapping tables presented in clause 5 for both points of view: the SPs and the UCs. The same process can be done if a new UC is proposed and the specific requirements across all clause 5 can be properly identified and selected.

B.3.5 Risk computation and mapping

This clause presents the description of the mathematical steps about the selected security analysis methodology and the resulting association of the SP identified in clause 4.5 with the UCs in clause 4.6 based on the security analysis procedure described in this clause and the methodology applied in clause B.4 for all UCs.

The complete process is presented in the following steps:

- Assign numeric values to L and M as in clauses B.4.2 and B.4.3.
- Inherent Risk Score (R_inh):
  - The level of risk before any controls or mitigations are considered.
  - Per threat (clause B.1.1) based on the Risk Factors in clause B.1.2: $R_{inh} = \mathrm{L} \times \mathrm{M}$ (range 1 to 25).

- Map the obtained R_inh to the associated SP for the studied threat:

Table B.5: Inherent risk values to SP mapping

R_inh ranges	1 - 5	6 - 15	16 - 25
SP	Minimal Cybersecurity	Medium Cybersecurity	High Cybersecurity

  - This process is repeated based on the number of identified threats in clause B.1.1
  - Tie the SP decision to the draft's cybersecurity requirement sets (clause 5).

- Repeat the previous process for all threats in clause B.1.1 like the examples in clause B.4. Once they are completed, identify the overall SP based on "mean value" from all the individual SPs.

The following diagram (Figure B.2) presents in an illustrative way the previous process for the product vendor to identify the right risk classification and the corresponding SP for its product.

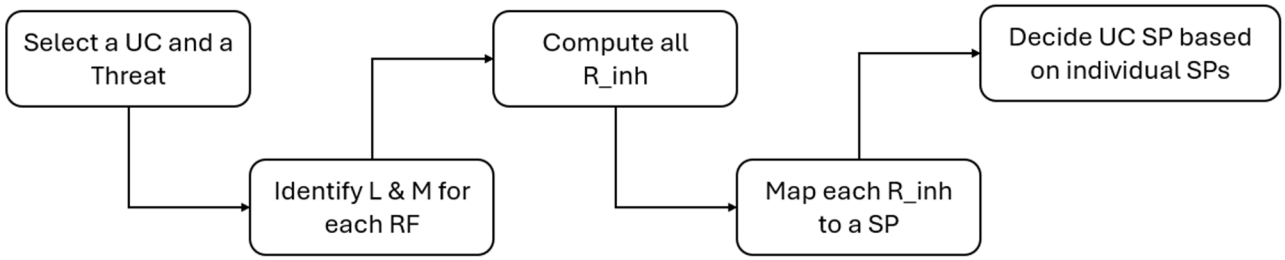

Figure B.2: Illustrative security analysis methodology

B.3.6 Example of Risk Analysis

The following text aims to illustrate and example of how the previous methodology should be applied to base on the threats and requirements presented in the present document.

Scenario: UC1 - Limited Operational Context.

Threat: TH1.1 - Tampered Signature Database

Components: Update System, Signature DB, Secure Communications

Computed values:

- L & M Identification:

Table B.6: Example of UC associated risk factors values per threat

L	RFL1	RFL2	RFL3	RFL4		RFL5		Mean RFL	L (rounded Mean RFL)
	2	1	1	2		1		1.4	1
M	RFM1	RFM2	RFM3	RFM4	RFM5	RFM6	RFM7	Mean RFM	M (rounded Mean RFM)
	2	2	2	2	2	2	2	2	2

- Inherent risk: $R_{inh} = 1 \times 2 = 2$
- Individual SP identification: Due to the R_inh achieving the value of 2 and based in the mapping in Table B.5, the assigned Individual SP for the selected threat TH1.1 in the UC1 is Minimum Cybersecurity.

UC SP identification:

- The previous step needs to be applied to all the identified threats in clause B.1.1 so the overall SP value can be determined by applying a "mean value" procedure:
  - Individual SP mapping to numerical values: Minimum (1), Medium (2), High (3)
  - $UC\ SP = \frac{\sum Individual\ SP}{total\ number\ of\ threats\ in\ Annex\ B.1.1}$
  - Final UC SP decision: 0 - 1 (Minimal SP), 1,01 - 2,0 (Medium SP), 2,01 - 3 (High SP).
- With the identified UC SP, the associated cybersecurity requirements for the assessed UC can be identified in clause 5 either per UC or per SP (if the assessed UC does not belong to any of those in the present document).

B.4 Use Cases Security Analysis

B.4.1 Introduction

This annex presents the detailed values used for the complete security analysis of the UC based on the threats and RFs towards the definition of the SP for each UC that allows to identify the requirements in clause 5 that the product under assessment needs to take into account.

NOTE: All values in the L and M rows correspond to the rounded values in the "Mean RFL/RFM" raw, which contain the mean values obtained.

B.4.2 UC1: Limited Impact Context

This annex B.4.2 demonstrates the complete security analysis done for UC1 based on the methodology in clause B.3.

Table B.7: UC1 associated risk factors values per threat

	RFL1	RFL2	RFL3	RFL4	RFL5	Mean	L	RFM1	RFM2	RFM3	RFM4	RFM5	RFM6	RFM7	Mean	M
TH1.1	2	1	1	2	1	1,4	1	2	2	2	2	2	2	2	2	2
TH1.2	2	1	1	2	2	1,6	2	1	2	3	2	2	2	2	2	2
TH1.3	1	1	1	2	2	1,4	1	1	1	3	1	2	1	2	1,6	2
TH1.4	2	1	1	2	2	1,6	2	1	2	3	2	2	2	2	2	2
TH1.5	2	1	1	2	1	1,4	1	1	1	2	1	1	1	2	1,3	1
TH1.6	2	1	1	2	2	1,6	2	2	1	3	2	2	1	2	1,9	2
TH2.1	1	1	1	1	2	1,2	1	1	2	1	1	2	1	1	1,3	1
TH2.2	1	1	2	1	1	1,2	1	2	1	1	1	2	1	1	1,3	1
TH2.3	2	1	1	1	2	1,4	1	2	2	1	1	3	2	1	1,7	2
TH2.4	1	1	1	1	1	1,0	1	1	1	1	1	2	1	1	1,1	1
TH2.5	1	1	2	1	2	1,4	1	2	2	1	1	3	2	1	1,7	2
TH3.1	2	1	1	3	2	1,8	2	2	1	3	1	1	1	2	1,6	2
TH3.2	2	1	1	3	2	1,8	2	2	1	3	2	1	1	2	1,7	2
TH3.3	2	1	2	2	2	1,8	2	3	1	1	1	1	1	1	1,3	1
TH4.1	1	1	1	1	2	1,2	1	1	2	1	3	1	2	1	1,6	2
TH4.2	2	1	1	2	2	1,6	2	1	2	3	3	2	2	2	2,1	2
TH4.3	1	1	1	1	1	1,0	1	1	1	1	3	1	1	1	1,3	1
TH5.1	3	1	1	2	2	1,8	2	2	2	1	1	2	2	1	1,6	2
TH5.2	2	1	2	1	1	1,4	1	2	1	1	1	1	1	2	1,3	1
TH5.3	1	3	1	1	1	1,4	1	2	1	1	1	2	2	1	1,4	1
TH5.4	2	1	1	1	2	1,4	1	2	2	1	1	2	2	1	1,6	2
TH6.1	2	1	3	1	1	1,6	2	2	1	1	1	1	1	1	1,1	1
TH6.2	2	1	2	1	1	1,4	1	2	1	1	1	1	1	1	1,1	1

Table B.8: UC1 security analysis values

	L	M	R_inh	Individual SP / Threat	UC SP
TH1.1	1	2	2	1	
TH1.2	2	2	4	1	
TH1.3	1	2	2	1	
TH1.4	2	2	4	1	
TH1.5	1	1	1	1	
TH1.6	2	2	4	1	
TH2.1	1	1	1	1	
TH2.2	1	1	1	1	
TH2.3	1	2	2	1	
TH2.4	1	1	1	1	
TH2.5	1	2	2	1	
TH3.1	2	2	4	1	**1,0 → Minimal**
TH3.2	2	2	4	1	
TH3.3	2	1	2	1	
TH4.1	1	2	2	1	
TH4.2	2	2	4	1	
TH4.3	1	1	1	1	
TH5.1	2	2	4	1	
TH5.2	1	1	1	1	
TH5.3	1	1	1	1	
TH5.4	1	2	2	1	
TH6.1	2	1	2	1	
TH6.2	1	1	1	1	

B.4.3 UC2: Baseline Impact Context

This annex B.4.3 demonstrates the complete security analysis done for UC2 based on the methodology in clause B.3.

Table B.9: UC2 associated risk factors values per threat

	RFL1	RFL2	RFL3	RFL4	RFL5	Mean	L	RFM1	RFM2	RFM3	RFM4	RFM5	RFM6	RFM7	Mean	M
TH1.1	3	3	3	3	2	2,8	3	3	2	3	4	4	3	3	3,1	3
TH1.2	3	1	2	3	3	2,4	2	2	2	4	2	3	2	3	2,6	3
TH1.3	3	2	2	3	3	2,6	3	2	2	4	2	3	2	2	2,4	2
TH1.4	3	1	1	3	2	2,0	2	2	2	4	2	3	2	3	2,6	3
TH1.5	3	1	1	3	1	1,8	2	1	1	3	1	2	1	2	1,6	2
TH1.6	3	1	1	3	2	2,0	2	2	2	4	2	3	2	3	2,6	3
TH2.1	4	2	3	3	3	3,0	3	2	3	2	3	3	3	3	2,7	3
TH2.2	3	2	3	2	2	2,4	2	2	2	2	2	2	2	2	2,0	2
TH2.3	4	2	3	3	3	3,0	3	2	2	1	1	4	3	2	2,1	2
TH2.4	2	1	1	1	1	1,2	1	2	2	2	2	3	2	1	2,0	2
TH2.5	2	1	2	2	2	1,8	2	2	2	1	1	4	3	2	2,1	2
TH3.1	4	1	2	4	3	2,8	3	2	1	3	1	2	1	2	1,7	2
TH3.2	4	1	2	4	3	2,8	3	2	1	3	2	2	1	2	1,9	2
TH3.3	4	2	3	4	3	3,2	3	3	2	2	2	2	2	2	2,1	2
TH4.1	4	2	2	3	3	2,8	3	1	2	1	3	2	2	2	1,9	2
TH4.2	4	2	2	4	4	3,2	3	1	2	4	3	3	2	3	2,6	3
TH4.3	3	2	2	2	2	2,2	2	2	2	2	4	2	2	2	2,3	2
TH5.1	4	2	2	4	3	3,0	3	2	2	1	1	3	2	2	1,9	2
TH5.2	4	2	3	3	2	2,8	3	3	2	2	2	2	2	4	2,4	2
TH5.3	3	3	2	3	2	2,6	3	3	2	2	2	3	3	2	2,4	2
TH5.4	3	1	1	2	2	1,8	2	2	2	1	1	3	2	2	1,9	2
TH6.1	4	3	4	4	2	3,4	3	3	2	3	2	3	3	3	2,7	3
TH6.2	3	2	3	3	2	2,6	3	2	1	1	1	1	1	1	1,1	1

Table B.10: UC2 security analysis values

	L	M	R_inh	Individual SP / Threat	UC SP
TH1.1	3	3	9	2	
TH1.2	2	3	6	2	
TH1.3	3	2	6	2	
TH1.4	2	3	6	2	
TH1.5	2	2	4	1	
TH1.6	2	3	6	2	
TH2.1	3	3	9	2	
TH2.2	2	2	4	1	
TH2.3	3	2	6	2	
TH2.4	1	2	2	1	
TH2.5	2	2	4	1	
TH3.1	3	2	6	2	**1,70 (2) → Medium**
TH3.2	3	2	6	2	
TH3.3	3	2	6	2	
TH4.1	3	2	6	2	
TH4.2	3	3	9	2	
TH4.3	2	2	4	1	
TH5.1	3	2	6	2	
TH5.2	3	2	6	2	
TH5.3	3	2	6	2	
TH5.4	2	2	4	1	
TH6.1	3	3	9	2	
TH6.2	3	1	3	1	

B.4.4 UC3: Elevated Impact Context

This annex B.4.4 demonstrates the complete security analysis done for UC3 based on the methodology in clause B.3.

Table B.11: UC3 associated risk factors values per threat

	RFL1	RFL2	RFL3	RFL4	RFL5	Mean	L	RFM1	RFM2	RFM3	RFM4	RFM5	RFM6	RFM7	Mean	M
TH1.1	3	4	3	3	4	3,4	3	3	3	3	4	4	3	3	3,3	3
TH1.2	3	3	3	4	4	3,4	3	4	4	4	4	4	4	3	3,9	4
TH1.3	3	3	2	4	4	3,2	3	3	3	4	3	4	3	4	3,4	3
TH1.4	3	2	2	3	3	2,6	3	3	3	4	3	3	3	3	3,1	3
TH1.5	3	2	2	3	2	2,4	2	2	2	4	2	3	2	3	2,6	3
TH1.6	3	2	2	3	3	2,6	3	3	3	4	3	3	3	3	3,1	3
TH2.1	4	3	3	3	4	3,4	3	3	4	3	3	4	3	3	3,3	3
TH2.2	3	3	3	2	3	2,8	3	4	3	3	3	4	3	3	3,3	3
TH2.3	3	3	3	3	4	3,2	3	3	3	2	2	4	3	3	2,9	3
TH2.4	4	4	3	3	4	3,6	4	3	3	4	4	4	3	3	3,4	3
TH2.5	3	2	2	2	3	2,4	2	3	3	2	2	4	3	2	2,7	3
TH3.1	4	3	3	4	4	3,6	4	4	3	4	3	4	3	4	3,6	4
TH3.2	3	3	3	4	4	3,4	3	4	3	3	4	4	3	4	3,6	4
TH3.3	4	3	3	3	4	3,4	3	5	3	3	3	3	3	3	3,3	3
TH4.1	3	3	2	3	4	3,0	3	3	4	3	5	4	4	4	3,9	4
TH4.2	4	3	3	4	4	3,6	4	3	4	5	5	4	4	4	4,1	4
TH4.3	3	3	2	3	3	2,8	3	3	3	4	5	3	3	3	3,4	3
TH5.1	4	3	2	4	4	3,4	3	4	4	3	3	4	3	3	3,4	3
TH5.2	4	4	3	3	4	3,6	4	4	3	3	4	4	3	4	3,6	4
TH5.3	3	4	3	3	4	3,4	3	4	3	4	3	4	3	4	3.6	4
TH5.4	4	2	2	3	3	2,8	3	3	3	2	2	4	3	3	2,9	3
TH6.1	3	3	3	3	3	3,0	3	3	3	3	3	3	3	3	3,0	3
TH6.2	3	3	3	3	3	3,0	3	3	3	3	3	3	3	3	3,0	3

Table B.12: UC3 security analysis values

	L	M	R_inh	Individual SP / Threat	UC SP
TH1.1	3	3	9	2	**2,13 (3) → High**
TH1.2	3	4	12	2	
TH1.3	3	3	9	2	
TH1.4	3	3	9	2	
TH1.5	2	3	6	2	
TH1.6	3	3	9	2	
TH2.1	3	3	9	2	
TH2.2	3	3	9	2	
TH2.3	3	3	9	2	
TH2.4	4	3	12	2	
TH2.5	2	3	6	2	
TH3.1	4	4	16	3	
TH3.2	3	4	12	2	
TH3.3	3	3	9	2	
TH4.1	3	4	12	2	
TH4.2	4	4	16	3	
TH4.3	3	3	9	2	
TH5.1	3	3	9	2	
TH5.2	4	4	16	3	
TH5.3	3	4	12	2	
TH5.4	3	3	9	2	
TH6.1	3	3	9	2	
TH6.2	3	3	9	2	

B.4.5 UC4: High Impact Context

This annex B.4.5 demonstrates the complete security analysis done for UC4 based on the methodology in clause B.3.

Table B.13: UC4 associated risk factors values per threat

	RFL1	RFL2	RFL3	RFL4	RFL5	Mean	L	RFM1	RFM2	RFM3	RFM4	RFM5	RFM6	RFM7	Mean	M
TH1.1	4	4	3	3	5	3,8	4	5	5	5	4	4	5	3	4,4	4
TH1.2	4	3	3	4	5	3,8	4	4	4	5	4	4	4	4	4,1	4
TH1.3	4	4	3	4	5	4,0	4	4	4	4	4	4	4	4	4,0	4
TH1.4	4	4	3	4	4	3,8	4	4	4	5	4	4	4	3	4,0	4
TH1.5	4	4	3	4	5	4,0	4	3	3	4	3	4	3	3	3,3	3
TH1.6	4	4	3	4	4	3,8	4	4	4	5	4	4	4	3	4,0	4
TH2.1	4	4	4	4	4	4,0	4	4	5	4	4	4	4	4	4,1	4
TH2.2	4	4	4	3	4	3,8	4	5	4	4	4	4	4	4	4,1	4
TH2.3	4	3	3	3	4	3,4	3	4	4	3	3	5	4	3	3,7	4
TH2.4	4	4	3	3	4	3,6	4	3	4	4	4	4	4	4	3,9	4
TH2.5	3	3	3	3	4	3,2	3	4	4	3	3	5	4	3	3,7	4
TH3.1	4	4	3	4	5	4,0	4	5	4	4	4	4	4	4	4,1	4
TH3.2	4	3	3	4	4	3,6	4	4	3	4	4	4	3	4	3,7	4
TH3.3	4	4	4	4	5	4,2	4	5	4	4	4	4	4	4	4,1	4
TH4.1	4	3	2	3	4	3,2	3	4	5	4	4	4	4	4	4,1	4
TH4.2	4	4	3	4	5	4,0	4	3	4	4	4	4	4	4	3,9	4
TH4.3	4	4	3	3	5	3,8	4	4	4	4	5	4	4	4	4,1	4
TH5.1	5	4	3	4	4	4,0	4	5	5	4	4	5	5	4	4,6	5
TH5.2	5	4	3	4	4	4,0	4	4	4	4	4	4	4	4	4,0	4
TH5.3	4	4	3	3	4	3,6	4	4	3	4	3	4	4	4	3,7	4
TH5.4	4	4	3	4	4	3,8	4	4	4	3	3	5	4	3	3,7	4
TH6.1	4	3	3	3	3	3,2	3	3	3	3	3	3	3	3	3,0	3
TH6.2	3	3	3	3	3	3,0	3	3	3	3	3	3	3	3	3,0	3

Table B.14: UC4 security analysis values

	L	M	R_inh	Individual SP / Threat	UC SP
TH1.1	4	4	16	3	
TH1.2	4	4	16	3	
TH1.3	4	4	16	3	
TH1.4	4	4	16	3	
TH1.5	4	3	12	2	
TH1.6	4	4	16	3	
TH2.1	4	4	16	3	
TH2.2	4	4	16	3	
TH2.3	3	4	12	2	
TH2.4	4	4	16	3	
TH2.5	3	4	12	2	
TH3.1	4	4	16	3	**2,74 (3) → High**
TH3.2	4	4	16	3	
TH3.3	4	4	16	3	
TH4.1	3	4	12	2	
TH4.2	4	4	16	3	
TH4.3	4	4	16	3	
TH5.1	4	5	20	3	
TH5.2	4	4	16	3	
TH5.3	4	4	16	3	
TH5.4	4	4	16	3	
TH6.1	3	3	9	2	
TH6.2	3	3	9	2	

B.4.6 UC5: Critical Impact Context

This annex B.4.6 demonstrates the complete security analysis done for UC5 based on the methodology in clause B.3.

Table B.15: UC5 associated risk factors values per threat

	RFL1	RFL2	RFL3	RFL4	RFL5	Mean	L	RFM1	RFM2	RFM3	RFM4	RFM5	RFM6	RFM7	Mean	M
TH1.1	4	4	3	3	4	4,2	4	5	5	5	5	5	5	3	4,7	5
TH1.2	4	4	3	3	4	4,2	4	5	5	5	4	5	5	4	4,7	5
TH1.3	4	4	3	3	4	4,2	4	4	4	5	4	5	4	4	4,3	4
TH1.4	4	5	3	3	5	4,0	4	5	5	5	4	5	5	3	4,6	5
TH1.5	4	5	3	3	4	3,8	4	4	4	4	4	4	4	3	3,9	4
TH1.6	4	5	3	3	5	4,0	4	5	5	5	4	5	5	3	4,6	5
TH2.1	4	4	4	3	4	4,2	4	4	5	4	4	5	4	4	4,3	4
TH2.2	4	4	4	3	4	3,8	4	5	4	4	4	5	4	4	4,3	4
TH2.3	3	4	4	2	3	4,0	4	5	5	4	4	5	5	3	4,4	4
TH2.4	4	4	3	3	4	3,8	4	4	5	4	4	5	4	4	4,3	4
TH2.5	4	5	4	3	5	4,2	4	5	5	4	4	5	5	3	4,4	4
TH3.1	4	4	3	3	4	4,0	4	5	4	4	4	4	4	4	4,1	4
TH3.2	4	4	4	3	4	4,0	4	5	5	4	4	5	5	4	4,6	5
TH3.3	4	4	4	3	4	4,2	4	5	4	4	4	4	4	4	4,1	4
TH4.1	3	4	4	2	3	3,6	4	4	5	4	5	5	5	4	4,6	5
TH4.2	4	4	3	3	4	4,0	4	4	5	5	5	5	5	4	4,7	5
TH4.3	4	4	3	3	4	4,0	4	4	4	4	5	4	4	4	4,1	4
TH5.1	4	5	3	3	4	4,0	4	5	5	4	4	5	5	4	4,6	5
TH5.2	4	4	4	3	4	4,2	4	4	4	4	4	4	4	4	4,0	4
TH5.3	4	4	3	3	4	3,8	4	5	4	4	4	5	5	4	4,4	4
TH5.4	5	4	3	3	4	3,8	4	5	5	3	4	5	5	3	4,3	4
TH6.1	3	3	4	2	3	3,6	4	4	4	3	4	4	4	3	3,7	4
TH6.2	3	3	4	2	3	4,0	4	4	4	3	4	4	4	3	3,7	4

Table B.16: UC5 security analysis values

	L	M	R_inh	Individual SP / Threat	UC SP
TH1.1	4	5	20	3	
TH1.2	4	5	20	3	
TH1.3	4	4	16	3	
TH1.4	4	5	20	3	
TH1.5	4	4	16	3	
TH1.6	4	5	20	3	
TH2.1	4	4	16	3	
TH2.2	4	4	16	3	
TH2.3	4	4	16	3	
TH2.4	4	4	16	3	
TH2.5	4	4	16	3	
TH3.1	4	4	16	3	**3,0 → High**
TH3.2	4	5	20	3	
TH3.3	4	4	16	3	
TH4.1	4	5	20	3	
TH4.2	4	5	20	3	
TH4.3	4	4	16	3	
TH5.1	4	5	20	3	
TH5.2	4	4	16	3	
TH5.3	4	4	16	3	
TH5.4	4	4	16	3	
TH6.1	4	4	16	3	
TH6.2	4	4	16	3	

Annex C (informative): Relationship between the present document and any related ETSI standards

This annex identifies possible relationships between product identified in the CRA [i.1]. The objective in this clause is to determine if the product in the present document overlaps in some aspects or have dependencies with other product requirements and how to proceed in that case:

Platform & OS layer → The product in the present document is dependent on the underlying OS hosting it. Without an OS, the product will not be able to perform. To this end, the product manufacturer may take into account the cybersecurity requirements identified [i.8] for the OSs available in the market and determine how its product performs based on the OS. For example to validate that the OS the product relies on is capable to provide a secure file management.

Firewall → While the product in the present document focuses on the analysis of incoming files from the outside of the protected system domain, Firewalls [i.9] focus on network traffic. These two product may not collide among them but cooperate and so, no overlaps or dependencies are identified.

SIEM and SOAR → The product do not overlap with SIEM or SOAR but assist them on achieving a more secure domain by generating and feeding information related to malware threats (e.g. malware family/name, detection method, file data, etc.). With this data, SIEM may identify patterns and warn the SOAR to trigger reactive processes such as scanning or quarantine actions. Based on this, SIEM and SOAR-based product manufacturers should take into account the requirements identified in the present document to properly offer their product on the market. Additionally, manufacturers with a product depending on the present document may check for possible requirements defined in [i.10].

BOOT MANAGERS → The product present in the present document may have some ties with Boot Managers during the earliest parts of startup to anchor ZT and spot tampering. In terms of dependencies, the product presented in the present document depends on the boot chain for a verified start, early loading, and enforcement of driver signing; it also leverages TPM/attestation to confirm nothing altered the boot path. In return, the OS/boot sequence may rely on the product to classify sensitive drivers conservatively, avoid breaking boot entries, and use supported APIs (not patching boot components). Both sides have to cooperate on recovery flows and cleanup of temporary boot entries to keep systems stable and trustworthy. Based on these possible dependencies, product manufacturers may check the cybersecurity requirements in [i.11] to ensure possible aspects not presented in the present document.

Annexes D to F:
Void

Annex G (informative): Guidelines on the implementation of the present document

The present document has been developed under the core idea of using UCs (or SPs) to determine the cybersecurity requirements for a product under assessment. To this end, the reader of the product may use the present document as stated in the following approach:

Step 1: To check if the product is in scope of the present document provided in clause 1 by:

- verifying that the product fits within the "technical description" of the "category of product" number "4" by Regulation (EU) 2025/2392 [i.2]; and
- identifying the product context by identifying:
  - the product's architectural components; and
  - all functions provided by the product; and
  - the architectural components' operational environments; and
  - the architectural components' interfaces and communication types; and
- verifying that the product is covered within the product context described in clause 4 by verifying that:
  - the functions provided by the product are covered within clause 4.1; and
  - the product architectural components are covered within the product architecture in clause 4.2; and
  - the architectural components' operational environments are covered within clause 4.3; and
  - the architectural functions offered to/required from other products are covered within clause 4.4.

NOTE 1: The identification of the items mentioned above can be reused for assessing the compliance of the product with the requirements in clause 5.

NOTE 2: If a product fits in the specification present in Regulation (EU) 2025/2392 [i.2] but is not covered within the product context described in clause 4, it is assumed that the product is not or not completely covered by the present document. Such cases should be notified to ETSI at the following e-mail address: cybersupport@etsi.org to consider those products in potential revisions of the present document.

Step 2: To determine the UC and/or the SP by:

- identifying the UC where the product under assessment may belong among the UCs in clause 4.6; or

NOTE 3: Identifying the type of user (clause 4.5) that is expected to use the product under, may help to better identify the right UC.

- if the product under assessment does not fit in any UC, the security analysis has to be applied to determine the SP for the product under assessment following the methodology presented in clause B.3 and implemented to the current UCs in clause B.4.

NOTE 4: If the product under assessment does not fit any of the existing UCs in the present document, a new UC and its security analysis (clause B.4) to determine the SP should be proposed to update the present document. Such proposal should be addressed to ETSI at the following e-mail address: cybersupport@etsi.org.

Step 3: To select concrete cybersecurity requirements for the product under assessment by checking the applicability table at the end of each requirements subclause in clause 5.

NOTE 5: If any of the selected requirements specify the need of RDPS feature or cryptography requirements, Annex R contains the specific applicability table for RDPS-related requirements; and all Annex K requirements apply.

Step 4: To assess the compliance of the product for each selected requirement in clause 5 by performing the associated assessment for compliance described in clause 6 by:

- preparing the assessment for the product; and
- performing the assessment activities for the product; and
- generating the supporting evidences for the assessment; and
- assigning an assessment verdict for the product.

Annexes H to J:
Void

Annex K (normative): Generic cryptographic requirements and assessment

K.1 Cryptography

K.1.1 Requirement

The product's default configuration shall only use cryptographic mechanisms that meet at least one of the following criteria:

1) **ACM-listed:** the cryptographic mechanism is listed in the ECCG Agreed Cryptographic Mechanisms (ACM) catalogue [1];

2) **ACM-extended:** the cryptographic mechanism is not listed in the ECCG Agreed Cryptographic Mechanisms (ACM) catalogue [1] and meets at least one of the following conditions:

   a) the cryptographic mechanism is listed in clause K.3.2 as an ACM-extended cryptographic mechanism for the specific product function(s);

   b) where the cryptographic mechanism is not listed in clause K.3.2, the cryptographic mechanism meets all the following criteria:

      i) the cryptographic mechanism, or where applicable the ACM-listed cryptographic mechanism on which it is based, is not deprecated per the ECCG Agreed Cryptographic Mechanisms (ACM) catalogue [1];

      ii) the cryptographic mechanism has been specified, developed or maintained through a transparent process by a recognised European, international or sector-specific standards development organization, or by an industry specification organization accountable for the relevant specification, including *[CEN, CENELEC, EPC, ETSI, ISO, IEC, ISO/IEC JTC 1, IETF, IEEE, ITU-T, NIST, 3GPP, O-RAN Alliance];* or the cryptographic mechanism is listed as suitable in a publicly available cryptographic catalogue maintained by a recognised national or governmental cybersecurity authority, where the catalogue is maintained under a documented revision and retirement process, including *[ENISA Cryptographic Studies, SOG-IS Agreed Cryptographic Mechanisms, ETSI Standards, CAVP, CMVP, CSRC, BSI Algorithm Catalogue, CRYPTREC]*;

      iii) the cryptographic mechanism is described in a valid, publicly available and uniquely referenceable specification;

      iv) the cryptographic properties of the cryptographic mechanism are known;

      v) no known weakness affects the cryptographic mechanism in a way that affects its cryptographic properties;

      vi) the cryptographic mechanism is required for a specific set of product functions;

      vii) any additional criteria specified by the vertical standard, where applicable.

3) **Interoperability-based:** the cryptographic mechanism is listed in clause K.4.2 as an interoperability-based cryptographic mechanism for specific product function(s) and external specification(s) or external requirement(s).

NOTE 1: The reference to the product's default configuration is intended to define a clear and assessable baseline, corresponding to the configuration in which the product is placed on the market. The product can provide several configurations that fulfil the requirement.

NOTE 2: For products supplied as hardware platforms or components to be configured by an integrator, references in this annex to the product's default configuration refer to the configuration specified for the intended operational use of the product after integration, including the relevant integration assumptions and configuration constraints.

NOTE 3: The applicable ACM catalogue version is identified in the normative references of the present document. The lifecycle treatment of mechanisms affected by ACM deprecation dates, expiry dates, migration conditions or usage limitations is addressed in clause K.2.

NOTE 4: In this clause, an external specification or external requirement means a specification or requirement that is imposed on the product, and which requires the use of a specific cryptographic mechanism for the product to interoperate with an identified system, platform or operational context. An external requirement can be a regulatory requirement, operational constraint, technical interoperability constraint, or platform compatibility requirement.

NOTE 5: Inclusion of a cryptographic mechanism under the interoperability-based criterion does not classify that mechanism as state-of-the-art cryptography.

EXAMPLE: Examples of product functions include secure communication based on TLS 1.3; authenticated encryption of communicated data based on AES-GCM; storage confidentiality based on AES-XTS; firmware signature verification based on Ed25519; and key derivation based on HKDF.

K.1.2 Assessment of product cryptographic configuration

K.1.2.0 General

The assessment in clause K.1.2 verifies the cryptographic mechanisms used in the product's default configuration against clause K.1.1.

For ACM-extended cryptographic mechanisms, the assessment verifies that the cryptographic mechanism is either listed in clause K.3.2 or fulfils the criteria specified in clause K.1.1 item 2)b). For interoperability-based cryptographic mechanisms, the assessment verifies that the cryptographic mechanism is listed in clause K.4.2. The assessment also verifies that the product uses the cryptographic mechanism in accordance with the relevant characteristics, product functions, use cases where applicable, external specifications or external requirements, and conditions specified in the present document.

K.1.2.1 Assessment of ACM-listed cryptographic mechanisms

K.1.2.1.1 Assessment objective

The purpose of this assessment case is to verify that cryptographic mechanisms claimed to comply with clause K.1.1 item 1) and used in the product's default configuration are listed in the ECCG Agreed Cryptographic Mechanisms (ACM) catalogue [1].

K.1.2.1.2 Assessment preparation

- Preconditions for the assessment: The product's default configuration shall be used for the assessment.
- The documentation shall provide a list of cryptographic mechanisms used in the product's default configuration and claimed under clause K.1.1 item 1), including the related product function, relevant parameters where applicable, and the relevant ACM entry.

K.1.2.1.3 Assessment activities

The assessment shall include verification that:

a) each cryptographic mechanism claimed under clause K.1.1 item 1) is identified by reference to a relevant ACM entry;

b) the ACM entry corresponds to the cryptographic mechanism used in the product's default configuration, including relevant parameters where applicable;

c) where the ACM entry includes lifecycle information, such as expiry date, deprecation date, migration condition or usage limitation, this information is reflected in the documentation.

K.1.2.1.4 Assessment evidence

The assessment evidence shall include, as applicable:

a) description of the product's default configuration;

b) list of cryptographic mechanisms claimed under clause K.1.1 item 1);

c) references to the relevant ACM entries;

d) relevant parameters, profiles, cipher suites or configuration constraints, where applicable;

e) relevant lifecycle information from the ACM catalogue, where applicable.

K.1.2.1.5 Assessment verdict

- The verdict **PASS** shall be assigned if the required evidence has been provided, is complete, is applicable to the assessed configuration, and demonstrates compliance with clause K.1.1 item 1).
- The verdict **FAIL** shall be assigned otherwise.

K.1.2.2 Assessment of ACM-extended cryptographic mechanisms

K.1.2.2.1 Assessment objective

The purpose of this assessment case is to verify that cryptographic mechanisms claimed to comply with clause K.1.1 item 2) and used in the product's default configuration are either listed as ACM-extended cryptographic mechanisms in clause K.3.2 or fulfil the criteria specified in clause K.1.1 item 2)b), and are used for the claimed product function(s).

K.1.2.2.2 Assessment preparation

- Preconditions for the assessment: The product's default configuration shall be used for the assessment.
- The documentation shall provide a list of cryptographic mechanisms used in the product's default configuration and claimed under clause K.1.1 item 2), including the related product function, use case where applicable, relevant parameters where applicable, and at least one of the following:

  a) the relevant entry in clause K.3.2; or

  b) evidence that the cryptographic mechanism fulfils all applicable criteria in clause K.1.1 item 2)b)

K.1.2.2.3 Assessment activities

The assessment shall include verification that:

a) each cryptographic mechanism claimed under clause K.1.1 item 2) is either listed in clause K.3.2 as an ACM-extended cryptographic mechanism or supported by evidence demonstrating fulfilment of the applicable criteria in clause K.1.1 item 2)b);

b) the cryptographic mechanism used by the product corresponds to the mechanism listed in clause K.3.2 or described in the evidence provided under clause K.1.1 item 2)b);

c) the product function using the cryptographic mechanism, and the use case where applicable, correspond to the related product function and use case specified in clause K.3.2 or justified under clause K.1.1 item 2)b);

d) the relevant characteristics, parameters, profiles, cipher suites or configuration constraints of the cryptographic mechanism correspond to those specified in clause K.3.2 or justified under clause K.1.1 item 2)b), where applicable;

e) where clause K.3.2 or the justification under clause K.1.1 item 2)b) defines conditions or limitations for the use of the cryptographic mechanism, these conditions or limitations are reflected in the product's default configuration;

f) where technically feasible, the cryptographic mechanism, parameters and configuration identified in the documentation correspond to the assessed product configuration.

K.1.2.2.4 Assessment evidence

The assessment evidence shall include, as applicable:

a) description of the product's default configuration;

b) list of cryptographic mechanisms claimed under clause K.1.1 item 2);

c) reference to the relevant entry in clause K.3.2, where the mechanism is listed in clause K.3.2;

d) evidence that the cryptographic mechanism fulfils the criteria in clause K.1.1 item 2)b), where compliance with clause K.1.1 item 2) is claimed on the basis of those criteria;

e) identification of the related product function using the cryptographic mechanism and the use case where applicable;

f) relevant characteristics, parameters, profiles, cipher suites or configuration constraints, where applicable;

g) evidence that the cryptographic mechanism is used in accordance with the conditions or limitations specified in clause K.3.2 or in the justification under clause K.1.1 item 2)b), where applicable.

K.1.2.2.5 Assessment verdict

- The verdict **PASS** shall be assigned if the required evidence has been provided, is complete, is applicable to the assessed configuration, and demonstrates compliance with clause K.1.1 item 2).
- The verdict **FAIL** shall be assigned otherwise.

K.1.2.3 Assessment of interoperability-based cryptographic mechanisms

K.1.2.3.1 Assessment objective

The purpose of this assessment case is to verify that cryptographic mechanisms claimed to comply with clause K.1.1 item 3) and used in the product's default configuration are listed as interoperability-based cryptographic mechanisms in clause K.4.2 and are used in accordance with the characteristics, product functions, use cases where applicable, external specifications or external requirements, and conditions specified in clause K.4.2.

K.1.2.3.2 Assessment preparation

- Preconditions for the assessment: The product's default configuration shall be used for the assessment.
- The documentation shall provide a list of cryptographic mechanisms used in the product's default configuration and claimed under clause K.1.1 item 3), including the related product function, use case where applicable, relevant parameters where applicable, the external specification or external requirement requiring its use, and the relevant entry in clause K.4.2.

K.1.2.3.3 Assessment activities

The assessment shall include verification that:

a) each cryptographic mechanism claimed under clause K.1.1 item 3) is listed in clause K.4.2 as an interoperability-based cryptographic mechanism;

b) the cryptographic mechanism used by the product corresponds to the cryptographic mechanism listed in clause K.4.2;

c) the product function using the cryptographic mechanism, and the use case where applicable, correspond to the related product function and use case specified in clause K.4.2;

d) the external specification or external requirement requiring the use of the cryptographic mechanism corresponds to the external specification or external requirement specified in clause K.4.2;

e) the relevant characteristics, parameters, profiles, cipher suites or configuration constraints of the cryptographic mechanism correspond to those specified in clause K.4.2, where applicable;

f) the conditions or limitations specified in clause K.4.2 for the use of the cryptographic mechanism are reflected in the product's default configuration;

g) where technically feasible, the cryptographic mechanism, parameters and configuration identified in the documentation correspond to the assessed product configuration.

K.1.2.3.4 Assessment evidence

The assessment evidence shall include, as applicable:

a) description of the product's default configuration;

b) list of cryptographic mechanisms claimed under clause K.1.1 item 3);

c) reference to the relevant entry in clause K.4.2;

d) identification of the related product function using the cryptographic mechanism and the use case where applicable;

e) identification of the external specification or external requirement requiring use of the cryptographic mechanism;

f) relevant characteristics, parameters, profiles, cipher suites or configuration constraints, where applicable;

g) evidence that the cryptographic mechanism is used in accordance with the conditions or limitations specified in clause K.4.2.

K.1.2.3.5 Assessment verdict

- The verdict **PASS** shall be assigned if the required evidence has been provided, is complete, is applicable to the assessed configuration, and demonstrates compliance with clause K.1.1 item 3).
- The verdict **FAIL** shall be assigned otherwise.

NOTE 1: The assessment of the external specification or external requirement itself is outside the scope of this assessment case. The assessment verifies that the external specification or external requirement is identified in clause K.4.2 and that the cryptographic mechanism is used by the product for the related product function.

NOTE 2: Documentation review is a minimum assessment activity under the present annex. Functional and/or resistance testing activities can be added by vertical standards where relevant for the product category, product function or cryptographic mechanism.

K.2 Crypto agility

K.2.1 Requirement

Where the product's default configuration uses a cryptographic mechanism for which the ACM catalogue, or the present document, specifies a deprecation date, expiry date, migration condition or usage limitation falling within the intended lifetime of the product, the product shall provide means for addressing the affected cryptographic mechanism by one or more of the following:

a) updating the cryptographic mechanism;

b) using another cryptographic mechanism that complies with clause K.1.1 and is not subject to the relevant deprecation date, expiry date, migration condition or usage limitation;

c) disabling the use of the affected cryptographic mechanism; or

d) limiting the use of the affected product functions accordingly.

EXAMPLE: Where a security mechanism uses a hybrid cryptographic construction, for example a hybrid key-encapsulation mechanism combining a classical and a post-quantum primitive, the lifecycle status of each constituent primitive is relevant to the lifecycle status of the construction. Hybridisation can be used as part of a planned migration strategy where permitted by the ACM catalogue or by the present document. However, hybridisation does not, by itself, extend the recommended usage lifetime of a constituent primitive that has been deprecated.

NOTE 1: Where a hybrid cryptographic construction includes multiple cryptographic primitives, the lifecycle status of each constituent primitive is relevant to the lifecycle status of the construction. Hybridisation is not assumed to mitigate the deprecation of a constituent primitive unless the ACM catalogue or the present document classifies the hybrid construction as suitable for the corresponding product function.

NOTE 2: Crypto agility supports maintaining appropriate cryptographic protection within the intended lifetime of the product, in addition to the capability of updating cryptographic mechanisms on the product in accordance with secure update and secure communication mechanisms.

NOTE 3: Deprecation information from sources other than the ACM catalogue or the present document can be considered as input to vulnerability management or security risk assessment but does not by itself trigger the requirement in clause K.2.1.

NOTE 4: Where the product provides cryptographic capabilities for use by other products, components or layers, the mechanism required by clause K.2.1 can consist of update, configuration, disabling, deprecation or limitation capabilities usable by the integrating product, system operator or user, where applicable.

NOTE 5: The applicability condition of this requirement can express exceptions based on product characteristics, e.g. cryptographic implementation based on immutable hardware co-processor(s) or hardware-based root of trust, or long-lived silicon used in a long-lived non-updateable environment.

K.2.2 Assessment of crypto-agility

K.2.2.1 Assessment objective

The purpose of this assessment case is to verify whether, where the product's default configuration uses a cryptographic mechanism for which the ACM catalogue, or the present document, specifies a deprecation date, expiry date, migration condition or usage limitation falling within the intended lifetime of the product, the product provides means to address the affected cryptographic mechanism in accordance with clause K.2.1.

K.2.2.2 Assessment preparation

- Preconditions for the assessment: The product's default configuration shall be used for the assessment.
- The documentation shall identify:

  a) the intended lifetime of the product;

  b) the cryptographic mechanisms used in the product's default configuration;

  c) the related product function(s) for each cryptographic mechanism;

  d) the lifecycle information applicable to each cryptographic mechanism, where specified by the ACM catalogue or the present document, including any deprecation date, expiry date, migration condition or usage limitation.

K.2.2.3 Assessment activities

The assessment shall include verification that:

a) for each cryptographic mechanism used in the product's default configuration, the lifecycle information specified by the ACM catalogue or the present document has been identified, where such information is specified;

b) where the ACM catalogue or the present document specifies a deprecation date, expiry date, migration condition or usage limitation for a cryptographic mechanism, this information is reflected in the documentation;

c) where a deprecation date, expiry date, migration condition or usage limitation falls within the intended lifetime of the product, the product provides an applicable mechanism for updating the cryptographic mechanism, using another cryptographic mechanism that complies with clause K.1.1 and is not subject to the relevant deprecation date, expiry date, migration condition or usage limitation, disabling the use of the affected cryptographic mechanism, or limiting the use of the affected product functions accordingly;

d) where another cryptographic mechanism is used, that cryptographic mechanism complies with clause K.1.1 and is not subject to the relevant deprecation date, expiry date, migration condition or usage limitation.

K.2.2.4 Assessment evidence

The assessment evidence shall include, as applicable:

a) documentation of the intended lifetime of the product;

b) list of cryptographic mechanisms used in the product's default configuration;

c) identification of the related product function(s) for each cryptographic mechanism;

d) references to the ACM catalogue entry or to the provision of the present document used to determine lifecycle information, where applicable;

e) identification of any deprecation date, expiry date, migration condition or usage limitation falling within the intended lifetime of the product;

f) description of the means provided by the product, such as update of the cryptographic mechanism, use of another cryptographic mechanism that complies with clause K.1.1 and is not subject to the relevant lifecycle constraint, disabling the use of the affected cryptographic mechanism, or limitation of the use of the affected product functions.

K.2.2.5 Assessment verdict

- The verdict **PASS** shall be assigned if the required evidence has been provided, is complete, is applicable to the assessed configuration, and demonstrates compliance with clause K.2.1.
- The verdict **FAIL** shall be assigned otherwise.

K.3 ACM-extended cryptographic mechanisms

K.3.1 Requirement

Where the product's default configuration uses ACM-extended cryptographic mechanisms, these mechanisms shall comply with the ACM-extended criterion in clause K.1.1 item 2).

K.3.2 List of ACM-extended cryptographic mechanisms

Table K.1 lists the ACM-extended cryptographic mechanisms specified by the present document.

Table K.1: ACM-extended cryptographic mechanisms

ACM-extended cryptographic mechanism	Type of cryptographic mechanism	Characteristics / parameters	Related product function(s) / use case(s), where applicable	Cryptographic properties	Specification / reference	Conditions or limitations, where applicable
Curve25519	Primitive	-	-	Confidentiality, Authentication, Key Establishment	[2]	
X25519	Primitive	256 bit	TLS, IPC	Key establishment	[2]	
Ed25519	Primitive	256 bit	Package Signature, TLS, IPC	Authentication	[3], [4]	
Curve448	Primitive	-	-	Confidentiality, Authentication, Key Establishment	[2]	
X448	Primitive	448 bit	TLS, IPC	Key establishment	[2]	
Ed448	Primitive	456 bit	Package Signature, TLS, IPC	Authentication	[3], [4]	
Argon2id	Algorithm	Refer to 4. Parameter Choice in IETF RFC 9106 [5] and the ACN's guidance.	Password-based hashing, Key derivation	Cryptographic hashing, Integrity, Authentication	[5], [i.15]	
scrypt	Algorithm	Refer to 2. scrypt Parameters in IETF RFC 7914 [6] and the ACN's guidance.	Password-based hashing, Key derivation	Cryptographic hashing, Integrity, Authentication	[6]	
ChaCha20	Algorithm	256 bit	TLS, data encryption	confidentiality	[7]	
Poly1305	Algorithm	256 bit	MAC	Authentication, Integrity	[7]	
AEGIS-X2	Algorithm	256 bit	Authenticated data encryption	Authentication, confidentiality, integrity	[i.16], [i.17]	

NOTE 1: The combination of the mechanism mentioned in Table K.1: ACM-extended cryptographic mechanisms together with the mechanisms in ACM that is appropriate for the cryptographic use-case to form a cryptographic protocol are allowed. For example ChaCha20-Poly1305 or X25519ML-KEM768.

NOTE 2: The use-case defined in Table K.1 marked in "Related product function(s) / use case(s), where applicable" is a non exhaustive list and listed algorithm can be used for other appropriate use-cases as well.

K.3.3 Assessment

Compliance with the ACM-extended criterion in clause K.1.1 item 2) is assessed in clause K.1.2.2.

K.4 Interoperability-based cryptographic mechanisms

K.4.1 Requirement

Where the product's default configuration uses interoperability-based cryptographic mechanisms, these mechanisms shall be selected from the interoperability-based cryptographic mechanisms listed in clause K.4.2 for the specified product functions and external specifications or external requirements and shall be used in accordance with the conditions or limitations specified in clause K.4.2.

K.4.2 List of interoperability-based cryptographic mechanisms

No interoperability-based cryptographic mechanisms are specified.

K.4.3 Assessment

Compliance with the requirement in clause K.4.1 is assessed in clause K.1.2.3.

NOTE: The assessment of the external specifications or external requirements themselves is outside the scope of this assessment. Their relevance is deemed confirmed by their inclusion in the present document.

Annexes L to Q:
Void

Annex R (normative): Requirements on RDPS

R.0 Introduction

This annex specifies additional technical requirements and assessment provisions for products with digital elements that rely on Remote Data Processing Solutions ("RDPS") for the provision or support of one or more product functions.

The requirements applicable to the product functions themselves remain specified in the core part of the concerned standard and continue to apply to the product as a whole, including where a product function is performed locally, supported by an RDPS, or performed in part through an RDPS.

This annex specifies supplementary requirements and assessment provisions for the product-facing RDPS boundary, i.e. for the additional attack surface, interactions, and dependency conditions introduced where a product function is performed or supported across the local product side and the RDPS side.

Where the product does not rely on remote data processing solutions for the provision or support of any product function, this annex is not applicable.

This annex does not extend to the full remote service environment or underlying third-party infrastructure, except to the extent necessary to define assumptions, dependencies, and assessment conditions relevant to the product-facing RDPS boundary and the secure operation of the RDPS-dependent product function.

R.1 RDPS as a product-boundary extension

When a product relies on RDPS for the provision or support of one or more product functions, the use of RDPS is considered, for the purposes of the present annex, to introduce an additional product-facing boundary and interface between the local product side and the RDPS side.

The present annex does not restate or replace the requirements applicable to the product function as such. It specifies only the supplementary requirements needed to address the increased attack surface introduced by the RDPS boundary.

The present annex therefore addresses the security implications introduced by that additional boundary and interface, including:

a) the exchanged interactions across the RDPS boundary;

b) the relevant product-side and RDPS-side endpoints; and

c) the dependency conditions arising where the secure operation of the product function relies on interactions across that boundary.

The scope of the present annex, as illustrated in Figure R.1, is limited to the RDPS interface between the local product side and the RDPS side.

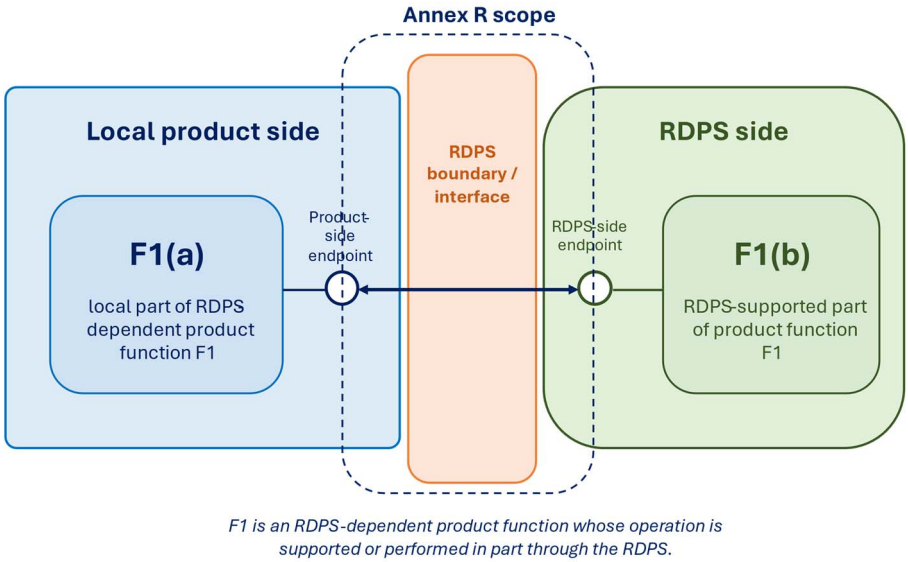

Figure R.1: RDPS model for Annex R - scope focused on the product ↔ RDPS boundary

For the purposes of the present annex:

- local product side means the product elements within the product boundary that participate in the interaction with the RDPS;
- RDPS means the remote data processing solution performing or supporting operations relevant to the RDPS-dependent product function;

RDPS interface means the communication boundary between the product-side endpoint and the RDPS-side endpoint.

R.2 Threat Model

R.2.0 Introduction

This clause does not restate or replace threats for local product components that are already defined in the main clauses of the standard. Instead, it focuses on:

- threats introduced by the local product ↔ RDPS interface, and
- threats introduced by the RDPS as a remote service capability.

The threat model:

- identifies RDPS assets; and
- defines a set of threats applicable to RDPS.

R.2.1 Assets

The threat model considers the following assets:

- **DA-RDPS-001 - RDPS interaction content**

The information exchanged across the RDPS boundary, including, as applicable, requests, commands, responses, decisions, configuration data, secrets, update-related data, and personal data where such data is exchanged.

For the purposes of DA-RDPS-001, Table R.1 identifies categories of RDPS interaction content and the security properties that are relevant to them.

Table R.1: Security properties for DA-RDPS-001

Content type	Data authenticity	Integrity	Confidentiality
Commands / requests & Responses / decisions	X	X	
Configuration data	X	X	
Secrets (e.g. keys)	X	X	X
Code or update data	X	X	
Confidential code or update data	X	X	X
Personal data		X	X

NOTE: In Table R.1, "data authenticity" refers to the authenticity or trusted origin of the exchanged interaction content. It does not refer to authentication of the RDPS boundary endpoint.

- **DA-RDPS-002 - RDPS-dependent product function**

Product function for which one or more operations depend on an RDPS, including the associated security-relevant state, decisions, outputs, or configuration.

For the purposes of DA-RDPS-002, Table R.2 identifies the security property relevant to the RDPS-dependent product function.

Table R.2: Security property for DA-RDPS-002

Content type	Availability
RDPS-dependent product function	X

R.2.2 Threat catalogue

The following threats constitute the baseline threat set considered for the RDPS boundary within the scope of Annex R, covering the RDPS interaction content exchanged across the boundary and the availability of the RDPS-dependent product function.

- **T-RDPS-01: Loss of data authenticity of interactions across the RDPS boundary**

An attacker causes an endpoint to accept interaction content across the RDPS boundary whose claimed source or origin is not genuine.

Threat type: Spoofing / impersonation

Security impact: The receiving endpoint may rely on interaction content that does not originate from the intended source, which can affect the behaviour, state, configuration, or security of the RDPS-dependent product function.

- **T-RDPS-02: Loss of integrity of interactions across the RDPS boundary**

An interaction exchanged across the RDPS boundary is modified, substituted, injected, corrupted, or replayed before it is processed by the receiving endpoint.

Threat type: Tampering

Security impact: The receiving endpoint may act upon interaction content whose contents no longer correspond to what was sent or intended by the legitimate source, which can affect the behaviour, state, configuration, or security of the RDPS-dependent product function.

- **T-RDPS-03: Unauthorised disclosure of interactions across the RDPS boundary**

Data exchanged across the RDPS boundary is disclosed to an unauthorised party.

Threat type: Information disclosure

Security impact: An attacker may obtain interaction content whose disclosure can affect the security or secure operation of the RDPS-dependent product function.

- **T-RDPS-04: Unavailability, unacceptable delay, or intolerable degradation of interactions across the RDPS boundary**

The local product side or the RDPS side is unable to send, receive, or process interactions across the RDPS boundary, or receives them only after an unacceptable delay, or experiences degradation of communication performance across the RDPS boundary to an extent that is not tolerable for the successful operation of the RDPS-dependent product function.

Threat type: Denial of service

Security impact: The RDPS-dependent product function may fail to operate as intended, may operate with incomplete, outdated, or degraded information, may remain blocked, or may continue without the required interaction being available when needed.

R.2.3 Assets and Threats Mapping

Table R.3: Mapping of Threats and Assets

Threat ID	Threat title	Related asset(s)
T-RDPS-01	Loss of data authenticity of interactions across the RDPS boundary	DA-RDPS-001
T-RDPS-02	Loss of integrity of interactions across the RDPS boundary	DA-RDPS-001
T-RDPS-03	Unauthorised disclosure of interactions across the RDPS boundary	DA-RDPS-001
T-RDPS-04	Unavailability, unacceptable delay, or intolerable degradation of interactions across the RDPS boundary	DA-RDPS-002

R.3 Security Requirements

R.3.1 General

The requirements in the present clause are organized by requirement family and, within each family, as a sequence of requirements expressing increasing rigor for the protection of the RDPS boundary and the RDPS-dependent product function.

Within a given requirement family, the lower-numbered requirement defines the baseline security outcome to be achieved, while the higher-numbered requirements define strengthened requirements for cases where increased protection is needed.

The requirement families in the present clause address the primary security properties identified for the assets in clause R.2.1, namely data authenticity, integrity, confidentiality, and availability.

R.3.2 Applicability of Annex R requirements

The requirements in the present clause address the protection of the security properties identified for the assets in clause R.2.1.

Where a security property is identified as relevant for a given asset in the asset tables of clause R.2.1, the corresponding requirement family or families in the present clause shall be considered for applicability to that asset.

For DA-RDPS-001, the primary requirement families considered for applicability are:

- data authenticity of interactions;
- integrity of interactions; and
- confidentiality of exchanged data.

For DA-RDPS-002, the primary requirement family considered for applicability is availability.

Similar to the requirements in clause 5, Table R.4 presents the applicability of the RDPS requirements defined in this annex for the UCs and SPs previously defined in clause 4.6 and clause B.3. Given the family-structure that RDPS requirements follow, for each family, the requirement with lower number is assigned to UCs with Minimal SP, the requirement with the middle number is assigned to UCs with Medium SP, and finally, the requirements with the highest number are assigned to UCs with high SP.

NOTE: The requirements defined in this Annex R do not follow the same "inclusion" rule than those in clause 5. This means that a product belonging to a UC with High SP, does not need to apply the requirements assigned to Minimal or Medium SPs as it happens with the requirements in clause 5. The reason for this change is that requirements in Annex R are evolutive: requirements with ID "001" defines the basic security, requirements with ID "002" increase the strength compared to requirements with ID "001", and requirements with ID "003" define the strongest security compared to its family requirements with lower IDs "001" and "002".

Table R.4: Standard-specific selection of Annex R requirements

Requirement ID	Family	UC1	UC2	UC3	UC4	UC5	Minimal SP	Medium SP	High SP
REQ-RDPS-L-AUTH-001	Data authenticity	X					X		
REQ-RDPS-L-AUTH-002	Data authenticity		X					X	
REQ-RDPS-L-AUTH-003	Data authenticity			X	X	X			X
REQ-RDPS-L-INTEG-001	Integrity	X					X		
REQ-RDPS-L-INTEG-002	Integrity		X					X	
REQ-RDPS-L-INTEG-003	Integrity			X	X	X			X
REQ-RDPS-L-CONF-001	Confidentiality	X					X		
REQ-RDPS-L-CONF-002	Confidentiality		X					X	
REQ-RDPS-L-CONF-003	Confidentiality			X	X	X			X
REQ-RDPS-L-AVAIL-001	Availability	X					X		
REQ-RDPS-L-AVAIL-002	Availability		X					X	
REQ-RDPS-L-AVAIL-003	Availability			X	X	X			X
REQ-RDPS-R-AUTH-001	Data authenticity	X					X		
REQ-RDPS-R-AUTH-002	Data authenticity		X					X	
REQ-RDPS-R-AUTH-003	Data authenticity			X	X	X			X
REQ-RDPS-R-INTEG-001	Integrity	X					X		
REQ-RDPS-R-INTEG-002	Integrity		X					X	
REQ-RDPS-R-INTEG-003	Integrity			X	X	X			X
REQ-RDPS-R-CONF-001	Confidentiality	X					X		
REQ-RDPS-R-CONF-002	Confidentiality		X					X	
REQ-RDPS-R-CONF-003	Confidentiality			X	X	X			X
REQ-RDPS-R-AVAIL-001	Availability	X					X		
REQ-RDPS-R-AVAIL-002	Availability		X					X	
REQ-RDPS-R-AVAIL-003	Availability			X	X	X			X

R.3.3 Local product side requirements

R.3.3.1 Data authenticity of interactions received from the RDPS side

- **REQ-RDPS-L-AUTH-001:** The local product side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, that the interaction originates from the intended RDPS-side endpoint.

  Rationale: This requirement protects against acceptance of interaction content whose claimed source or origin is not genuine.

- **REQ-RDPS-L-AUTH-002:** The local product side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, the data authenticity of that interaction using cryptographic endpoint authentication, an authenticated communication channel, or an equivalent cryptographic mechanism that provides assurance of the origin of the interaction.

  Rationale: This requirement strengthens protection against spoofed or falsely originated interaction content by requiring cryptographic assurance of origin.

- **REQ-RDPS-L-AUTH-003:** The local product side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, that:

  a) the cryptographic authenticity of the interaction is bound to the intended RDPS-side endpoint; and

  b) replayed or stale interactions are detected and rejected where replay or reuse could affect the secure operation of the RDPS-dependent product function.

  Rationale: This requirement provides stronger protection by requiring source-bound authenticity and replay protection.

R.3.3.2 Integrity of interactions received from the RDPS side

- **REQ-RDPS-L-INTEG-001:** The local product side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, that the interaction has not been modified, substituted, corrupted, or injected.

  Rationale: This requirement protects against loss of integrity of received interaction content.

- **REQ-RDPS-L-INTEG-002:** The local product side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, the integrity of that interaction using a cryptographic integrity-protection mechanism.

  Rationale: This requirement strengthens protection against unauthorised modification or substitution of received interactions.

- **REQ-RDPS-L-INTEG-003:** The local product side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, that the interaction is protected by a cryptographic integrity-protection mechanism bound to the authenticated source of the interaction.

  Rationale: This requirement provides stronger protection by requiring cryptographic integrity protection bound to the authenticated source.

R.3.3.3 Confidentiality of data exchanged with the RDPS side

- **REQ-RDPS-L-CONF-001:** Where disclosure of data exchanged with the RDPS side could lead to unacceptable cybersecurity risk for the RDPS-dependent product function, the local product side shall protect the confidentiality of that data during exchange with the RDPS side.

  Rationale: Confidentiality is not required for every exchange. However, where disclosure of data exchanged with the RDPS side could affect the security or secure operation of the RDPS-dependent product function, confidentiality protection is required.

- **REQ-RDPS-L-CONF-002:** Where disclosure of data exchanged with the RDPS side could lead to unacceptable cybersecurity risk for the RDPS-dependent product function, the local product side shall protect the confidentiality of that data during exchange with the RDPS side using cryptographic confidentiality protection.

  Rationale: This requirement strengthens confidentiality protection by requiring cryptographic protection of exchanged data where disclosure would create unacceptable cybersecurity risk.

- **REQ-RDPS-L-CONF-003:** Where disclosure of data exchanged with the RDPS side could lead to unacceptable cybersecurity risk for the RDPS-dependent product function, the local product side shall use cryptographic confidentiality protection that is established with the intended RDPS-side endpoint and that prevents disclosure of the exchanged data to endpoints other than that intended endpoint.

  Rationale: This requirement provides stronger protection by ensuring that confidential exchanged data is disclosed only to the intended peer endpoint.

R.3.3.4 Availability

- **REQ-RDPS-L-AVAIL-001:** Where the availability or timeliness of interactions received from the RDPS side is necessary for the secure operation of the RDPS-dependent product function, the local product side shall:

  a) detect unavailability of the RDPS side, unacceptable delay of expected interactions, or intolerable degradation of the interaction;

  b) use timeout or equivalent timeliness controls for expected RDPS-side interactions; and

  c) apply a defined behaviour for the RDPS-dependent product function.

  Rationale: This requirement ensures that loss, delay, or degradation of RDPS-side interactions is detected and handled in a defined manner before it adversely affects the secure operation of the RDPS-dependent product function.

- **REQ-RDPS-L-AVAIL-002:** Where the availability or timeliness of interactions received from the RDPS side is necessary for the secure operation of the RDPS-dependent product function, the local product side shall:

  a) detect unavailability of the RDPS side, unacceptable delay of expected interactions, or intolerable degradation of the interaction;

  b) use timeout or equivalent timeliness controls for expected RDPS-side interactions;

  c) apply a defined degraded behaviour or secure state when the required RDPS-side interactions are unavailable, delayed beyond the acceptable time, or degraded beyond acceptable limits; and

  d) support recovery of the RDPS-dependent product function using locally maintained state, configuration, or other recovery information where such information is necessary to restore secure operation.

  Rationale: This requirement strengthens protection by requiring controlled degraded operation or transition to a secure state, together with support for restoration of secure operation.

- **REQ-RDPS-L-AVAIL-003:** Where the availability or timeliness of interactions received from the RDPS side is necessary for the secure operation of the RDPS-dependent product function, the local product side shall:

  a) detect unavailability of the RDPS side, unacceptable delay of expected interactions, or intolerable degradation of the interaction;

  b) use timeout or equivalent timeliness controls for expected RDPS-side interactions;

  c) apply a defined degraded behaviour or secure state when the required RDPS-side interactions are unavailable, delayed beyond the acceptable time, or degraded beyond acceptable limits;

  d) support recovery of the RDPS-dependent product function using locally maintained state, configuration, or other recovery information where such information is necessary to restore secure operation;

  e) support continuity of operation through failover, redundancy, or an equivalent resilience mechanism where such a mechanism is necessary.

  Rationale: This requirement provides stronger protection by requiring resilience measures.

R.3.4 RDPS side requirements

R.3.4.1 Data authenticity of interactions received from the local product side

- **REQ-RDPS-R-AUTH-001:** The RDPS side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, that the interaction originates from the intended local product-side endpoint.

  Rationale: This requirement protects against acceptance of interaction content whose claimed source or origin is not genuine.

- **REQ-RDPS-R-AUTH-002:** The RDPS side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, the data authenticity of that interaction using cryptographic endpoint authentication, an authenticated communication channel, or an equivalent cryptographic mechanism that provides assurance of the origin of the interaction.

  Rationale: This requirement strengthens protection against spoofed or falsely originated interaction content by requiring cryptographic assurance of origin.

- **REQ-RDPS-R-AUTH-003:** The RDPS side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, that:

  a) the cryptographic authenticity of the interaction is bound to the intended local product-side endpoint; and

  b) replayed or stale interactions are detected and rejected where replay or reuse could affect the secure operation of the RDPS-dependent product function.

  Rationale: This requirement provides stronger protection by requiring source-bound authenticity and replay protection.

R.3.4.2 Integrity of interactions received from the local product side

- **REQ-RDPS-R-INTEG-001:** The RDPS side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, that the interaction has not been modified, substituted, corrupted, or injected.

  Rationale: This requirement protects against loss of integrity of received interaction content.

- **REQ-RDPS-R-INTEG-002:** The RDPS side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, the integrity of that interaction using a cryptographic integrity-protection mechanism.

  Rationale: This requirement strengthens protection against unauthorised modification or substitution of received interactions.

- **REQ-RDPS-R-INTEG-003:** The RDPS side shall verify, before acting upon a received interaction that can influence the behaviour, state, configuration, or security of the RDPS-dependent product function, that the interaction is protected by a cryptographic integrity-protection mechanism bound to the authenticated source of the interaction.

  Rationale: This requirement provides stronger protection by requiring cryptographic integrity protection bound to the authenticated source.

R.3.4.3 Confidentiality of data exchanged with the local product side

- **REQ-RDPS-R-CONF-001:** Where disclosure of data exchanged with the local product side could lead to unacceptable cybersecurity risk for the RDPS-dependent product function, the RDPS side shall protect the confidentiality of that data during exchange with the local product side.

  Rationale: This requirement ensures confidentiality where disclosure of exchanged data could affect the security or secure operation of the RDPS-dependent product function.

- **REQ-RDPS-R-CONF-002:** Where disclosure of data exchanged with the local product side could lead to unacceptable cybersecurity risk for the RDPS-dependent product function, the RDPS side shall protect the confidentiality of that data during exchange with the local product side using cryptographic confidentiality protection.

  Rationale: This requirement strengthens confidentiality protection by requiring cryptographic protection during exchange.

- **REQ-RDPS-R-CONF-003:** Where disclosure of data exchanged with the local product side could lead to unacceptable cybersecurity risk for the RDPS-dependent product function, the RDPS side shall use cryptographic confidentiality protection that is established with the intended local product-side endpoint and that prevents disclosure of the exchanged data to endpoints other than that intended endpoint.

Rationale: This requirement provides stronger protection by ensuring that confidential exchanged data is disclosed only to the intended peer endpoint.

R.3.4.4 Availability

- **REQ-RDPS-R-AVAIL-001:** Where the availability or timeliness of interactions received from the local product side is necessary for the secure operation of the RDPS-dependent product function, the RDPS side shall:

  a) detect unavailability of the local product side, unacceptable delay of expected interactions, or intolerable degradation of the interaction;

  b) use timeout or equivalent timeliness controls for expected local product-side interactions; and

  c) apply a defined behaviour for the RDPS-dependent product function.

  Rationale: This requirement ensures that loss, delay, or degradation of local product-side interactions is detected and handled in a defined manner before it adversely affects the secure operation of the RDPS-dependent product function.

- **REQ-RDPS-R-AVAIL-002:** Where the availability or timeliness of interactions received from the local product side is necessary for the secure operation of the RDPS-dependent product function, the RDPS side shall:

  a) detect unavailability of the local product side, unacceptable delay of expected interactions, or intolerable degradation of the interaction;

  b) use timeout or equivalent timeliness controls for expected local product-side interactions;

  c) apply a defined degraded behaviour or secure state when the required local product-side interactions are unavailable, delayed beyond the acceptable time, or degraded beyond acceptable limits; and

  d) support recovery of the RDPS-dependent product function using maintained state, configuration, or other recovery information where such information is necessary to restore secure operation.

  Rationale: This requirement strengthens protection by requiring controlled degraded operation or transition to a secure state, together with support for restoration of secure operation.

- **REQ-RDPS-R-AVAIL-003:** Where the availability or timeliness of interactions received from the local product side is necessary for the secure operation of the RDPS-dependent product function, the RDPS side shall:

  a) detect unavailability of the local product side, unacceptable delay of expected interactions, or intolerable degradation of the interaction;

  b) use timeout or equivalent timeliness controls for expected local product-side interactions;

  c) apply a defined degraded behaviour or secure state when the required local product-side interactions are unavailable, delayed beyond the acceptable time, or degraded beyond acceptable limits;

  d) support recovery of the RDPS-dependent product function using maintained state, configuration, or other recovery information where such information is necessary to restore secure operation; and

  e) support continuity of operation through failover, redundancy, or an equivalent resilience mechanism where such a mechanism is necessary.

  Rationale: This requirement provides stronger protection by requiring resilience measures.

R.3.5 Mapping of Threats and Requirements

Table R.5: Threats and Requirements mapping

Threat ID	Requirement(s) - Local product side	Requirement(s) - RDPS side
T-RDPS-01	REQ-RDPS-L-AUTH-001 REQ-RDPS-L-AUTH-002 REQ-RDPS-L-AUTH-003	REQ-RDPS-R-AUTH-001 REQ-RDPS-R-AUTH-002 REQ-RDPS-R-AUTH-003
T-RDPS-02	REQ-RDPS-L-INTEG-001 REQ-RDPS-L-INTEG-002 REQ-RDPS-L-INTEG-003	REQ-RDPS-R-INTEG-001 REQ-RDPS-R-INTEG-002 REQ-RDPS-R-INTEG-003
T-RDPS-03	REQ-RDPS-L-CONF-001 REQ-RDPS-L-CONF-002 REQ-RDPS-L-CONF-003	REQ-RDPS-R-CONF-001 REQ-RDPS-R-CONF-002 REQ-RDPS-R-CONF-003
T-RDPS-04	REQ-RDPS-L-AVAIL-001 REQ-RDPS-L-AVAIL-002 REQ-RDPS-L-AVAIL-003	REQ-RDPS-R-AVAIL-001 REQ-RDPS-R-AVAIL-002 REQ-RDPS-R-AVAIL-003

R.4 Conformity assessment

R.4.1 Local product side requirements

R.4.1.1 REQ-RDPS-L-AUTH-001

- **Assessment objective:**

  The objective of the assessment is to verify that the local product side verifies, before acting upon a relevant received interaction, that the interaction originates from the intended RDPS-side endpoint.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the interactions received from the RDPS side that can influence the behaviour, state, configuration, or security of that function;
  - the intended RDPS-side endpoint for those interactions; and
  - the mechanism used by the local product side to determine the origin of the received interaction.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the mechanism used by the local product side to determine the origin of the relevant interaction;
  - verify that the origin of the interaction is checked before the interaction is acted upon;
  - verify that interactions from the intended RDPS-side endpoint are accepted; and
  - verify that interactions not originating from the intended RDPS-side endpoint are rejected or not acted upon.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the local product side verifies the origin of the relevant interaction before acting upon it; and

- interactions from endpoints other than the intended RDPS-side endpoint are rejected or not acted upon.

The assessment is **failed if any of the following occur**:

- the local product side acts upon a relevant interaction without verifying its origin; or
- the local product side accepts and acts upon a relevant interaction not originating from the intended RDPS-side endpoint.

- **Assessment evidence:**

The following evidence may be used to support the assessment:

- design or configuration evidence describing how origin of the interaction is determined;
- evidence showing acceptance of interactions from the intended RDPS-side endpoint; and
- evidence showing rejection or non-processing of interactions from other endpoints.

R.4.1.2 REQ-RDPS-L-AUTH-002

- **Assessment objective:**

The objective of the assessment is to verify that the local product side verifies the data authenticity of relevant received interactions using cryptographic endpoint authentication, an authenticated communication channel, or an equivalent cryptographic mechanism providing assurance of origin.

- **Assessment preparation:**

Before starting the assessment, the following shall be identified:

- the RDPS-dependent product function;
- the relevant received interactions;
- the cryptographic mechanism used to provide assurance of origin; and
- the conditions under which the local product side accepts or rejects those interactions.

- **Assessment activities:**

The assessment shall include the following activities:

- inspect the design and configuration of the cryptographic mechanism used to provide assurance of origin;
- verify that the local product side checks the authenticity-protection status of the interaction before acting upon it;
- verify that a valid interaction protected by the intended mechanism is accepted; and
- verify that an interaction lacking valid cryptographic assurance of origin is rejected or not acted upon.

- **Assessment verdict:**

The assessment is **passed if all of the following occur**:

- the local product side uses a cryptographic mechanism providing assurance of origin for relevant received interactions;
- the local product side verifies that protection before acting upon the interaction; and
- interactions without valid assurance of origin are rejected or not acted upon.

The assessment is **failed if any of the following occur**:

- the local product side acts upon a relevant interaction without verifying cryptographic assurance of origin; or

- the local product side accepts and acts upon an interaction for which the required authenticity mechanism is absent or invalid.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for the cryptographic authenticity mechanism;
  - traces or logs showing successful acceptance of valid protected interactions; and
  - traces or logs showing rejection or non-processing of interactions without valid assurance of origin.

R.4.1.3 REQ-RDPS-L-AUTH-003

- **Assessment objective:**

  The objective of the assessment is to verify that the local product side verifies that the cryptographic authenticity of the received interaction is bound to the intended RDPS-side endpoint and that replayed or stale interactions are detected and rejected where replay or reuse could affect secure operation.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the relevant received interactions;
  - the mechanism used to bind authenticity to the intended RDPS-side endpoint; and
  - the replay or freshness protection mechanism, where applicable.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the mechanism that binds the authenticity of the interaction to the intended RDPS-side endpoint;
  - inspect the design and configuration of the freshness, anti-replay, or equivalent mechanism used;
  - verify that a valid interaction bound to the intended RDPS-side endpoint is accepted;
  - verify that an interaction with valid-looking authenticity protection but not bound to the intended RDPS-side endpoint is rejected or not acted upon; and
  - where replay or reuse could affect secure operation, verify that replayed or stale interactions are detected and rejected.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the local product side verifies that authenticity is bound to the intended RDPS-side endpoint; and
  - replayed or stale interactions are rejected where replay or reuse could affect secure operation.

  The assessment is **failed if any of the following occur**:

  - the local product side accepts and acts upon an interaction whose cryptographic authenticity is not bound to the intended RDPS-side endpoint; or
  - the local product side fails to detect and reject replayed or stale interactions where such protection is required.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for source binding and anti-replay/freshness controls;
  - evidence showing acceptance of valid bound interactions; and
  - evidence showing rejection of misbound, replayed, or stale interactions.

R.4.1.4 REQ-RDPS-L-INTEG-001

- **Assessment objective:**

  The objective of the assessment is to verify that the local product side checks that a relevant received interaction has not been modified, substituted, corrupted, or injected before acting upon it.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the relevant received interactions; and
  - the mechanism used by the local product side to detect modification, substitution, corruption, or injection.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the integrity-checking mechanism;
  - verify that integrity is checked before the interaction is acted upon;
  - verify that an unmodified valid interaction is accepted; and
  - verify that modified, substituted, corrupted, or injected interactions are rejected or not acted upon.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the local product side checks the integrity of relevant received interactions before acting upon them; and
  - modified, substituted, corrupted, or injected interactions are rejected or not acted upon.

  The assessment is **failed if any of the following occur**:

  - the local product side acts upon a relevant interaction without verifying its integrity; or
  - the local product side accepts and acts upon an altered, substituted, corrupted, or injected interaction.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design or configuration evidence of the integrity-checking mechanism;
  - evidence showing acceptance of valid interactions; and
  - evidence showing rejection or non-processing of altered or injected interactions.

R.4.1.5 REQ-RDPS-L-INTEG-002

- **Assessment objective:**

  The objective of the assessment is to verify that the local product side verifies the integrity of relevant received interactions using a cryptographic integrity-protection mechanism.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the relevant received interactions; and
  - the cryptographic integrity-protection mechanism used for those interactions.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the cryptographic integrity-protection mechanism;
  - verify that integrity verification takes place before the interaction is acted upon;
  - verify that a valid cryptographically protected interaction is accepted; and
  - verify that an interaction with missing or invalid integrity protection is rejected or not acted upon.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the local product side uses cryptographic integrity protection for relevant received interactions;
  - integrity verification is performed before the interaction is acted upon; and
  - interactions with invalid or missing integrity protection are rejected or not acted upon**.**

  The assessment is **failed if any of the following occur**:

  - the local product side does not use cryptographic integrity verification where required; or
  - the local product side accepts and acts upon an interaction whose integrity protection is missing or invalid.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for the cryptographic integrity mechanism;
  - evidence showing acceptance of valid protected interactions; and
  - evidence showing rejection of interactions with invalid or missing integrity protection.

R.4.1.6 REQ-RDPS-L-INTEG-003

- **Assessment objective:**

  The objective of the assessment is to verify that the local product side verifies that the relevant received interaction is protected by a cryptographic integrity-protection mechanism bound to the authenticated source of the interaction.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the relevant received interactions;
  - the authenticated source of those interactions; and
  - the mechanism used to bind integrity protection to that authenticated source.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the mechanism that binds integrity protection to the authenticated source;
  - verify that a valid interaction whose integrity protection is correctly bound to the authenticated source is accepted;
  - verify that an interaction with integrity protection not bound to the authenticated source is rejected or not acted upon; and
  - verify that integrity verification and source binding checks are performed before the interaction is acted upon.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the local product side verifies that the interaction is protected by cryptographic integrity protection bound to the authenticated source; and
  - interactions not correctly bound to that source are rejected or not acted upon.

  The assessment is **failed if any of the following occur**:

  - the local product side accepts and acts upon a relevant interaction whose integrity protection is not bound to the authenticated source; or
  - the local product side acts upon the interaction before performing the required checks.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for integrity binding to authenticated source;
  - evidence showing acceptance of correctly bound interactions; and
  - evidence showing rejection of misbound interactions.

R.4.1.7 REQ-RDPS-L-CONF-001

- **Assessment objective:**

  The objective of the assessment is to verify that, where disclosure of exchanged data could lead to unacceptable cybersecurity risk for the RDPS-dependent product function, the local product side protects the confidentiality of that data during exchange with the RDPS side.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;

- the categories of exchanged data for which disclosure could lead to unacceptable cybersecurity risk; and
- the mechanism used by the local product side to protect confidentiality during exchange.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the confidentiality-protection mechanism;
  - verify that the identified data is protected during exchange with the RDPS side; and
  - verify that data designated as requiring confidentiality is not exchanged in an unprotected manner.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the local product side protects the confidentiality of exchanged data where disclosure could lead to unacceptable cybersecurity risk; and
  - such data is not exchanged in an unprotected manner.

  The assessment is **failed if any of the following occur**:

  - the local product side does not protect the confidentiality of exchanged data where such protection is required;
  - identified confidential data is exchanged without appropriate confidentiality protection.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design or configuration evidence identifying the protected data and the means of protection; and
  - evidence showing that relevant exchanged data is protected during exchange.

R.4.1.8 REQ-RDPS-L-CONF-002

- **Assessment objective:**

  The objective of the assessment is to verify that the local product side protects the confidentiality of relevant exchanged data using cryptographic confidentiality protection.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the exchanged data requiring confidentiality protection; and
  - the cryptographic confidentiality-protection mechanism used during exchange.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the cryptographic confidentiality-protection mechanism;
  - verify that the identified exchanged data is protected by that mechanism during exchange; and
  - verify that exchange of such data without the required cryptographic confidentiality protection is prevented.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - relevant exchanged data is protected using cryptographic confidentiality protection; and
  - exchange without the required cryptographic protection is prevented.

  The assessment is **failed if any of the following occur**:

  - the local product side exchanges relevant confidential data without cryptographic confidentiality protection; or
  - the configured confidentiality mechanism is not effectively applied to the relevant data exchange.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for the cryptographic confidentiality mechanism; and
  - evidence showing protected exchange of the relevant data.

R.4.1.9 REQ-RDPS-L-CONF-003

- **Assessment objective:**

  The objective of the assessment is to verify that the local product side uses cryptographic confidentiality protection established with the intended RDPS-side endpoint and preventing disclosure of exchanged data to endpoints other than that intended endpoint.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the exchanged data requiring stronger confidentiality protection;
  - the intended RDPS-side endpoint; and
  - the cryptographic mechanism used to establish confidentiality protection with that endpoint.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the cryptographic confidentiality mechanism and its association with the intended RDPS-side endpoint;
  - verify that protected exchange is established with the intended RDPS-side endpoint;
  - verify that the local product side does not disclose the protected data to other endpoints under the same exchange context; and
  - verify that exchanges not established with the intended RDPS-side endpoint are rejected or do not result in disclosure of the protected data.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - confidentiality protection is established with the intended RDPS-side endpoint;
  - protected data is disclosed only to that intended endpoint; and
  - exchanges with other endpoints do not result in disclosure of the protected data.

The assessment is **failed if any of the following occur**:

- the local product side discloses protected data without establishing confidentiality protection with the intended RDPS-side endpoint; or
- the local product side discloses the protected data to endpoints other than the intended RDPS-side endpoint.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for endpoint-associated confidentiality protection;
  - evidence showing protected disclosure to the intended RDPS-side endpoint; and
  - evidence showing prevention of disclosure to other endpoints.

R.4.1.10 REQ-RDPS-L-AVAIL-001

- **Assessment objective:**

  The objective of the assessment is to verify that the local product side detects unavailability, unacceptable delay, or intolerable degradation of required RDPS-side interactions, uses timeout or equivalent timeliness controls, and applies a defined behaviour for the RDPS-dependent product function.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the required RDPS-side interactions;
  - the acceptable timing or service conditions for those interactions; and
  - the defined behaviour to be applied when the required interactions are unavailable, delayed, or degraded.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the timeout, timeliness, or equivalent detection controls;
  - verify that unavailability, unacceptable delay, or intolerable degradation of required interactions is detected;
  - verify that a defined behaviour is applied when such conditions occur; and
  - verify that, under normal conditions, expected interactions are processed without triggering the fault-handling behaviour.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the local product side detects the specified failure, delay, or degradation conditions;
  - timeout or equivalent timeliness controls are used; and
  - a defined behaviour is applied when the identified conditions occur.

  The assessment is **failed if any of the following occur**:

  - the local product side does not detect the relevant failure, delay, or degradation condition; or
  - the required defined behaviour is not applied when such a condition occurs.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for detection and timeliness controls;
  - evidence showing detection of failure, delay, or degradation; and
  - evidence showing application of the defined behaviour.

R.4.1.11 REQ-RDPS-L-AVAIL-002

- **Assessment objective:**

  The objective of the assessment is to verify that the local product side applies a defined degraded behaviour or secure state when required RDPS-side interactions are unavailable, delayed, or degraded, and supports recovery of the RDPS-dependent product function where needed.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the required RDPS-side interactions;
  - the degraded behaviour or secure state defined for failure conditions; and
  - any locally maintained state, configuration, or other recovery information required to restore secure operation.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the degraded behaviour or secure-state mechanism;
  - inspect the design and configuration of the recovery mechanism, where applicable;
  - verify that when required interactions become unavailable, delayed beyond the acceptable time, or degraded beyond acceptable limits, the local product side applies the defined degraded behaviour or secure state; and
  - verify that recovery support is available and can be used to restore secure operation where such support is required.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the local product side applies the defined degraded behaviour or secure state under the relevant failure conditions; and
  - recovery support is available and usable where necessary to restore secure operation.

  The assessment is **failed if any of the following occur**:

  - the local product side does not apply the defined degraded behaviour or secure state under the relevant failure conditions; or
  - required recovery support is absent or ineffective.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for degraded behaviour, secure state, and recovery support; and

- evidence showing activation of degraded behaviour / secure state and restoration of secure operation where applicable.

R.4.1.12 REQ-RDPS-L-AVAIL-003

- **Assessment objective:**

  The objective of the assessment is to verify that the local product side supports higher resilience, including continuity of operation through failover, redundancy, or equivalent resilience mechanisms where necessary.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the required RDPS-side interactions;
  - the defined degraded behaviour or secure state;
  - the recovery support; and
  - the failover, redundancy, or equivalent resilience mechanism, where required.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the resilience mechanism used to support continuity of operation;
  - verify that the defined degraded behaviour or secure state is applied under the relevant failure conditions;
  - verify that recovery support is available where required;
  - where failover, redundancy, or equivalent resilience is required, verify that continuity of operation is supported when the primary interaction path becomes unavailable, delayed, or degraded.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the local product side supports the required resilience mechanism where necessary;
  - defined degraded behaviour or secure state is applied under the relevant conditions; and
  - continuity of operation is supported in accordance with the design where resilience is required.

  The assessment is **failed if any of the following occur**:

  - a required resilience mechanism is absent or ineffective; or
  - the local product side fails to support continuity of operation where such support is required.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for resilience, degraded behaviour, and recovery support; and
  - evidence showing failover, redundancy, or equivalent continuity support where required.

R.4.2 RDPS side requirements

R.4.2.1 REQ-RDPS-R-AUTH-001

- **Assessment objective:**

  The objective of the assessment is to verify that the RDPS side verifies, before acting upon a relevant received interaction, that the interaction originates from the intended local product-side endpoint.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the interactions received from the local product side that can influence the behaviour, state, configuration, or security of that function;
  - the intended local product-side endpoint for those interactions; and
  - the mechanism used by the RDPS side to determine the origin of the received interaction.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the mechanism used by the RDPS side to determine the origin of the relevant interaction;
  - verify that the origin of the interaction is checked before the interaction is acted upon;
  - verify that interactions from the intended local product-side endpoint are accepted; and
  - verify that interactions not originating from the intended local product-side endpoint are rejected or not acted upon.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the RDPS side verifies the origin of the relevant interaction before acting upon it; and
  - interactions from endpoints other than the intended local product-side endpoint are rejected or not acted upon.

  The assessment is **failed if any of the following occur**:

  - the RDPS side acts upon a relevant interaction without verifying its origin; or
  - the RDPS side accepts and acts upon a relevant interaction not originating from the intended local product-side endpoint.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design or configuration evidence describing how origin of the interaction is determined;
  - evidence showing acceptance of interactions from the intended local product-side endpoint; and
  - evidence showing rejection or non-processing of interactions from other endpoints.

R.4.2.2 REQ-RDPS-R-AUTH-002

- **Assessment objective:**

  The objective of the assessment is to verify that the RDPS side verifies the data authenticity of relevant received interactions using cryptographic endpoint authentication, an authenticated communication channel, or an equivalent cryptographic mechanism providing assurance of origin.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the relevant received interactions;
  - the cryptographic mechanism used to provide assurance of origin; and
  - the conditions under which the RDPS side accepts or rejects those interactions.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the cryptographic mechanism used to provide assurance of origin;
  - verify that the RDPS side checks the authenticity-protection status of the interaction before acting upon it;
  - verify that a valid interaction protected by the intended mechanism is accepted; and
  - verify that an interaction lacking valid cryptographic assurance of origin is rejected or not acted upon.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the RDPS side uses a cryptographic mechanism providing assurance of origin for relevant received interactions;
  - the RDPS side verifies that protection before acting upon the interaction; and
  - interactions without valid assurance of origin are rejected or not acted upon.

  The assessment is **failed if any of the following occur**:

  - the RDPS side acts upon a relevant interaction without verifying cryptographic assurance of origin; or
  - the RDPS side accepts and acts upon an interaction for which the required authenticity mechanism is absent or invalid.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for the cryptographic authenticity mechanism;
  - traces or logs showing successful acceptance of valid protected interactions; and
  - traces or logs showing rejection or non-processing of interactions without valid assurance of origin.

R.4.2.3 REQ-RDPS-R-AUTH-003

- **Assessment objective:**

  The objective of the assessment is to verify that the RDPS side verifies that the cryptographic authenticity of the received interaction is bound to the intended local product-side endpoint and that replayed or stale interactions are detected and rejected where replay or reuse could affect secure operation.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the relevant received interactions;
  - the mechanism used to bind authenticity to the intended local product-side endpoint; and
  - the replay or freshness protection mechanism, where applicable.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the mechanism that binds the authenticity of the interaction to the intended local product-side endpoint;
  - inspect the design and configuration of the freshness, anti-replay, or equivalent mechanism used;
  - verify that a valid interaction bound to the intended local product-side endpoint is accepted;
  - verify that an interaction with valid-looking authenticity protection but not bound to the intended local product-side endpoint is rejected or not acted upon; and
  - where replay or reuse could affect secure operation, verify that replayed or stale interactions are detected and rejected.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the RDPS side verifies that authenticity is bound to the intended local product-side endpoint; and
  - replayed or stale interactions are rejected where replay or reuse could affect secure operation.

  The assessment is **failed if any of the following occur**:

  - the RDPS side accepts and acts upon an interaction whose cryptographic authenticity is not bound to the intended local product-side endpoint; or
  - the RDPS side fails to detect and reject replayed or stale interactions where such protection is required.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for source binding and anti-replay/freshness controls;
  - evidence showing acceptance of valid bound interactions; and
  - evidence showing rejection of misbound, replayed, or stale interactions.

R.4.2.4 REQ-RDPS-R-INTEG-001

- **Assessment objective:**

  The objective of the assessment is to verify that the RDPS side checks that a relevant received interaction has not been modified, substituted, corrupted, or injected before acting upon it.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the relevant received interactions; and
  - the mechanism used by the RDPS side to detect modification, substitution, corruption, or injection.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the integrity-checking mechanism;
  - verify that integrity is checked before the interaction is acted upon;
  - verify that an unmodified valid interaction is accepted; and
  - verify that modified, substituted, corrupted, or injected interactions are rejected or not acted upon.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the RDPS side checks the integrity of relevant received interactions before acting upon them; and
  - modified, substituted, corrupted, or injected interactions are rejected or not acted upon.

  The assessment is **failed if any of the following occur**:

  - the RDPS side acts upon a relevant interaction without verifying its integrity; or
  - the RDPS side accepts and acts upon an altered, substituted, corrupted, or injected interaction.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design or configuration evidence of the integrity-checking mechanism;
  - evidence showing acceptance of valid interactions; and
  - evidence showing rejection or non-processing of altered or injected interactions.

R.4.2.5 REQ-RDPS-R-INTEG-002

- **Assessment objective:**

  The objective of the assessment is to verify that the RDPS side verifies the integrity of relevant received interactions using a cryptographic integrity-protection mechanism.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the relevant received interactions; and

- the cryptographic integrity-protection mechanism used for those interactions.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the cryptographic integrity-protection mechanism;
  - verify that integrity verification takes place before the interaction is acted upon;
  - verify that a valid cryptographically protected interaction is accepted; and
  - verify that an interaction with missing or invalid integrity protection is rejected or not acted upon.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the RDPS side uses cryptographic integrity protection for relevant received interactions;
  - integrity verification is performed before the interaction is acted upon; and
  - interactions with invalid or missing integrity protection are rejected or not acted upon.

  The assessment is **failed if any of the following occur**:

  - the RDPS side does not use cryptographic integrity verification where required; or
  - the RDPS side accepts and acts upon an interaction whose integrity protection is missing or invalid.

- Assessment evidence:

  The following evidence may be used to support the assessment:

  - design and configuration evidence for the cryptographic integrity mechanism;
  - evidence showing acceptance of valid protected interactions; and
  - evidence showing rejection of interactions with invalid or missing integrity protection.

R.4.2.6 REQ-RDPS-R-INTEG-003

- **Assessment objective:**

  The objective of the assessment is to verify that the RDPS side verifies that the relevant received interaction is protected by a cryptographic integrity-protection mechanism bound to the authenticated source of the interaction.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the relevant received interactions;
  - the authenticated source of those interactions; and
  - the mechanism used to bind integrity protection to that authenticated source.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the mechanism that binds integrity protection to the authenticated source;

- verify that a valid interaction whose integrity protection is correctly bound to the authenticated source is accepted;
- verify that an interaction with integrity protection not bound to the authenticated source is rejected or not acted upon; and
- verify that integrity verification and source binding checks are performed before the interaction is acted upon.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the RDPS side verifies that the interaction is protected by cryptographic integrity protection bound to the authenticated source; and
  - interactions not correctly bound to that source are rejected or not acted upon.

  The assessment is **failed if any of the following occur**:

  - the RDPS side accepts and acts upon a relevant interaction whose integrity protection is not bound to the authenticated source; or
  - the RDPS side acts upon the interaction before performing the required checks.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for integrity binding to authenticated source;
  - evidence showing acceptance of correctly bound interactions; and
  - evidence showing rejection of misbound interactions.

R.4.2.7 REQ-RDPS-R-CONF-001

- **Assessment objective:**

  The objective of the assessment is to verify that, where disclosure of exchanged data could lead to unacceptable cybersecurity risk for the RDPS-dependent product function, the RDPS side protects the confidentiality of that data during exchange with the local product side.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the categories of exchanged data for which disclosure could lead to unacceptable cybersecurity risk; and
  - the mechanism used by the RDPS side to protect confidentiality during exchange.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the confidentiality-protection mechanism;
  - verify that the identified data is protected during exchange with the local product side; and
  - verify that data designated as requiring confidentiality is not exchanged in an unprotected manner.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the RDPS side protects the confidentiality of exchanged data where disclosure could lead to unacceptable cybersecurity risk; and
  - such data is not exchanged in an unprotected manner.

  The assessment is **failed if any of the following occur**:

  - the RDPS side does not protect the confidentiality of exchanged data where such protection is required; or
  - identified confidential data is exchanged without appropriate confidentiality protection.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design or configuration evidence identifying the protected data and the means of protection; and
  - evidence showing that relevant exchanged data is protected during exchange.

R.4.2.8 REQ-RDPS-R-CONF-002

- **Assessment objective:**

  The objective of the assessment is to verify that the RDPS side protects the confidentiality of relevant exchanged data using cryptographic confidentiality protection.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the exchanged data requiring confidentiality protection; and
  - the cryptographic confidentiality-protection mechanism used during exchange.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the cryptographic confidentiality-protection mechanism;
  - verify that the identified exchanged data is protected by that mechanism during exchange; and
  - verify that exchange of such data without the required cryptographic confidentiality protection is prevented.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - relevant exchanged data is protected using cryptographic confidentiality protection; and
  - exchange without the required cryptographic protection is prevented.

  The assessment is **failed if any of the following occur**:

  - the RDPS side exchanges relevant confidential data without cryptographic confidentiality protection; or
  - the configured confidentiality mechanism is not effectively applied to the relevant data exchange.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for the cryptographic confidentiality mechanism; and
  - evidence showing protected exchange of the relevant data.

R.4.2.9 REQ-RDPS-R-CONF-003

- **Assessment objective:**

  The objective of the assessment is to verify that the RDPS side uses cryptographic confidentiality protection established with the intended local product-side endpoint and preventing disclosure of exchanged data to endpoints other than that intended endpoint.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the exchanged data requiring stronger confidentiality protection;
  - the intended local product-side endpoint; and
  - the cryptographic mechanism used to establish confidentiality protection with that endpoint.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the cryptographic confidentiality mechanism and its association with the intended local product-side endpoint;
  - verify that protected exchange is established with the intended local product-side endpoint;
  - verify that the RDPS side does not disclose the protected data to other endpoints under the same exchange context; and
  - verify that exchanges not established with the intended local product-side endpoint are rejected or do not result in disclosure of the protected data.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - confidentiality protection is established with the intended local product-side endpoint;
  - protected data is disclosed only to that intended endpoint; and
  - exchanges with other endpoints do not result in disclosure of the protected data.

  The assessment is **failed if any of the following occur**:

  - the RDPS side discloses protected data without establishing confidentiality protection with the intended local product-side endpoint; or
  - the RDPS side discloses the protected data to endpoints other than the intended local product-side endpoint.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for endpoint-associated confidentiality protection;

- evidence showing protected disclosure to the intended local product-side endpoint; and
- evidence showing prevention of disclosure to other endpoints.

R.4.2.10 REQ-RDPS-R-AVAIL-001

- **Assessment objective:**

  The objective of the assessment is to verify that the RDPS side detects unavailability, unacceptable delay, or intolerable degradation of required local product-side interactions, uses timeout or equivalent timeliness controls, and applies a defined behaviour for the RDPS-dependent product function.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the required local product-side interactions;
  - the acceptable timing or service conditions for those interactions; and
  - the defined behaviour to be applied when the required interactions are unavailable, delayed, or degraded.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the timeout, timeliness, or equivalent detection controls;
  - verify that unavailability, unacceptable delay, or intolerable degradation of required interactions is detected;
  - verify that a defined behaviour is applied when such conditions occur; and
  - verify that, under normal conditions, expected interactions are processed without triggering the fault-handling behaviour.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the RDPS side detects the specified failure, delay, or degradation conditions;
  - timeout or equivalent timeliness controls are used; and
  - a defined behaviour is applied when the identified conditions occur.

  The assessment is **failed if any of the following occur**:

  - the RDPS side does not detect the relevant failure, delay, or degradation condition; or
  - the required defined behaviour is not applied when such a condition occurs.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for detection and timeliness controls;
  - evidence showing detection of failure, delay, or degradation; and
  - evidence showing application of the defined behaviour.

R.4.2.11 REQ-RDPS-R-AVAIL-002

- **Assessment objective:**

  The objective of the assessment is to verify that the RDPS side applies a defined degraded behaviour or secure state when required local product-side interactions are unavailable, delayed, or degraded, and supports recovery of the RDPS-dependent product function where needed.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the required local product-side interactions;
  - the degraded behaviour or secure state defined for failure conditions; and
  - any maintained state, configuration, or other recovery information required to restore secure operation.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the degraded behaviour or secure-state mechanism;
  - inspect the design and configuration of the recovery mechanism, where applicable;
  - verify that when required interactions become unavailable, delayed beyond the acceptable time, or degraded beyond acceptable limits, the RDPS side applies the defined degraded behaviour or secure state; and
  - verify that recovery support is available and can be used to restore secure operation where such support is required.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the RDPS side applies the defined degraded behaviour or secure state under the relevant failure conditions; and
  - recovery support is available and usable where necessary to restore secure operation.

  The assessment is **failed if any of the following occur**:

  - the RDPS side does not apply the defined degraded behaviour or secure state under the relevant failure conditions; or
  - required recovery support is absent or ineffective.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for degraded behaviour, secure state, and recovery support; and
  - evidence showing activation of degraded behaviour / secure state and restoration of secure operation where applicable.

R.4.2.12 REQ-RDPS-R-AVAIL-003

- **Assessment objective:**

  The objective of the assessment is to verify that the RDPS side supports higher resilience, including continuity of operation through failover, redundancy, or equivalent resilience mechanisms where necessary.

- **Assessment preparation:**

  Before starting the assessment, the following shall be identified:

  - the RDPS-dependent product function;
  - the required local product-side interactions;
  - the defined degraded behaviour or secure state;
  - the recovery support; and
  - the failover, redundancy, or equivalent resilience mechanism, where required.

- **Assessment activities:**

  The assessment shall include the following activities:

  - inspect the design and configuration of the resilience mechanism used to support continuity of operation;
  - verify that the defined degraded behaviour or secure state is applied under the relevant failure conditions;
  - verify that recovery support is available where required;
  - where failover, redundancy, or equivalent resilience is required, verify that continuity of operation is supported when the primary interaction path becomes unavailable, delayed, or degraded.

- **Assessment verdict:**

  The assessment is **passed if all of the following occur**:

  - the RDPS side supports the required resilience mechanism where necessary;
  - defined degraded behaviour or secure state is applied under the relevant conditions; and
  - continuity of operation is supported in accordance with the design where resilience is required.

  The assessment is **failed if any of the following occur**:

  - a required resilience mechanism is absent or ineffective; or
  - the RDPS side fails to support continuity of operation where such support is required.

- **Assessment evidence:**

  The following evidence may be used to support the assessment:

  - design and configuration evidence for resilience, degraded behaviour, and recovery support; and
  - evidence showing failover, redundancy, or equivalent continuity support where required.

Annex S (informative): Change history

Version	Information about changes
V0.0.1	• Defined initial structure of the present document. • Defined the scope of the present document. • Added initial content in clause 4 on: – General information about the product. – Essential Functionalities. – Cybersecurity Levels – Identified Threats – Use cases – Drafted a generic architecture. – Users • Placeholder Annex A: Mapping of requirements in Clause 5 and CRA essential requirements. • Placeholder Annex B: Security analysis methodology
V0.0.2	• Redefined content in clause 4 on: – Essential Functionalities. – Cybersecurity Levels – Identified Threats – Use cases • Added initial content in clause 5 with a first set of cybersecurity requirements.
V0.0.3	• Redefined content in clause 4 on: – General information about the product. – Cybersecurity Levels – Identified Threats – Updated generic architecture figure and added description. • Updated content in clause 5 with new cybersecurity requirements.
V0.0.4	• Redefined content in clause 4 on: – General information about the product. – Cybersecurity Levels description – Identified Threats – Use Cases introductions – Users' introduction – Updated generic architecture figure and added description. – Removed clause 4.7.1 and converted it into cybersecurity requirements in clause 5. • Updated content in clause 5 on: – Added new cybersecurity requirements. • Added content in ANNEX B: Risk analysis methodology
V0.0.5	• Updated content in clause 4 on: – Cybersecurity Levels description – New Use Cases • Updated content in clause 5 on: – New structure pre requirements to map UCs relationship. • Improved content in ANNEX B: Risk analysis methodology
V0.0.6	• Updated overall structure as agreed with all other parallel CRA-related vertical standards. • Updated clause 5 content: - Removed and merged requirements that are equal or overlap with others, respectively. - Added tables to assist product vendors to understand which requirements to apply based on UC or SP • Added content in clause 6 for requirements until clause 5.2.6. • Updated the Annexes distribution and their content.
V0.0.7	• Updated clause 4 content by reopening the discussion about the users and UCs to improve the content as some doubts raised with the previous version. • New content in clause 6 regarding the requirements assessment.
V0.0.8	• Text updated with the final decision on the Users and UCs in clause 4. • Requirements in clause 5 has been updated to make them clearer and/or granular.

Version	Information about changes
V0.0.9	• Added sub-clauses in clause 4 about ZT, cryptography and logging to assist the related requirements. • Annex C completed. • Improved clause 6 content with new assessment text for those requirements that were updated, modified or removed.
V0.0.10	• Tables in clauses 5 to relate sec requirements with UC and SP completed. • Annex A completed.
V0.0.11	• Editorial issues and typos corrected.
V0.0.12	• Editorial issues and typos corrected for the 1st HAS Assessment.
V0.0.13	• HAS and experts comments resolution.
V0.0.14	• HAS and experts comments resolution.
V0.0.15	• Updated harmonised structure agreed among vertical rapporteurs. - Updated clause 4, 5 and 6 structure and titles - Merged Threats, risk and assessment into Annex B - Added Annex K on Cryptography. • HAS and experts comments resolution.
V0.0.16	• Updated clauses 1, 2, 3 and 4 with latest comments from antivirus call experts.
V0.0.17	• Updated clause 5 and Annex B based on received comments from experts and HAS assessment comments..
V0.0.18	• Updated clause 5 with new received comments from experts. Some of the requirements has open options to discuss with experts during the antivirus call.
V0.0.19	• Updated clause 5 requirements and removed clause 6 assessment from related to removed requirements.
V0.0.20	• Update of clauses 4 and 5 with the definition of users and accepted requirements.
V0.0.21	• Update of clause 5 with accepted requirements and updated assessments in clause 6 from those requirements accepted in previous version 0.0.20.
V0.0.22	• Updated clause 5 with new accepted requirements and clause 6 with placeholders.
V0.0.23	• Updated clause 5 with new accepted requirements and clause 6 with placeholders.
V0.0.24	• Updated clause 5 with new accepted requirements and clause 6 assessments.
V0.0.25	• Completed first final set of requirements (clause 5) and updated clause 6 assessments.
V0.0.26	• Clause 5 complete with first set of reviewed requirements. • Clause 6 completed with updated assessments and based on agreed clause 5. • Annex B updated with updated Threats text (B.1.1) and UCs security analysis placeholders (B.5).
V0.0.27	• New requirement and assessment text in clauses 5 and 6 pointing to Annex R. • Annex R added.
V0.0.28	• Updated UCs in clause 4, removing any impact-related content. • Started to fill in the tables with the applicability of requirements per SP.
V0.0.29	• Reviewed Annex B: Threats and Risk Factors to solve comments from 1st HAS Assessment.
V0.0.30	• Completed clause B.5: Security Analysis with all RFs values and resulting SP.
V0.0.31	• Integration with latest cross-vertical topics: skeleton, Annex K and Annex R. • Added extra-ACM cryptography mechanisms based on experts proposal.
V0.1.0	• Final Draft to submit to the CYBER EUSR as ready for Public Enquiry. (1st try)
V0.1.1	• Updated draft with content based on the comments from CYBER EUSR requesting to complete missing threats, the corresponding requirements and assessments.
V0.1.2	• Updated draft with comments from antivirus regular calls experts with updated on the CYBER EUSR comments.
V0.1.3	• Final Draft to submit to the CYBER EUSR as ready for Public Enquiry. (2nd try)

History

Version	Date	Status	
V1.0.0	July 2026	SRdAP process	EV 20261012: 2026-07-20 to 2026-10-12