ETSI TR 103 621 V1.2.1 (2022-09)

Guide to Cyber Security for Consumer Internet of Things

Reference
RTR/CYBER-0084

Keywords
cybersecurity, IoT

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 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 Web server (https://ipr.etsi.org/).

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™** and **LTE™** 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.

BLUETOOTH® is a trademark registered and owned by Bluetooth SIG, Inc.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Cyber Security (CYBER).

Modal verbs terminology

In the present document "**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 growth of the Internet of Things has spurred the development of security requirements for IoT devices. ETSI EN 303 645 [i.1] provides baseline cyber security provisions for a wide range of Consumer IoT products and remains outcome focused. While ETSI TS 103 701 [i.3] deals with the assessment of conformance of IoT products against the provisions of ETSI EN 303 645 [i.1], the present document has been developed to guide manufacturers on its implementation, by providing non-exhaustive examples of practical solutions that can be used to meet these provisions. Additionally, the examples provided herein are not limitative; it is possible to meet the provisions in ETSI EN 303 645 [i.1] by using other solutions.

1 Scope

The present document serves as guidance to help manufacturers and other stakeholders in meeting the cyber security provisions defined for Consumer IoT devices in ETSI EN 303 645 [i.1] and ETSI TS 103 645 [i.2].

The present document is complementary to ETSI EN 303 645 [i.1] and ETSI TS 103 701 [i.3]. It explains the relationship between these specifications and how they can be used together. It also provides a non-exhaustive set of example implementations that can be used to meet the provisions of ETSI EN 303 645 [i.1] and ETSI TS 103 645 [i.2], noting that not all possible implementations are included. Where relevant, pointers to supporting specifications are provided. Usage by industry players as well as future development of standards, such as specialisation into precise use cases, or certification aspects, are being given consideration.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

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 are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI EN 303 645 (V2.1.1): "CYBER; Cyber Security for Consumer Internet of Things: Baseline Requirements".

[i.2] ETSI TS 103 645 (V2.1.2): "CYBER; Cyber Security for Consumer Internet of Things: Baseline Requirements".

NOTE: The technical content of ETSI TS 103 645 (V2.1.2) is exactly the same as in ETSI EN 303 645 (V2.1.1).

[i.3] ETSI TS 103 701 (V1.1.1): "CYBER; Cyber Security for Consumer Internet of Things: Conformance Assessment of Baseline Requirements".

[i.4] IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".

[i.5] IETF RFC 6347: "Datagram Transport Layer Security Version 1.2".

[i.6] GNU Bison.

NOTE: Available at https://www.gnu.org/software/bison/.

[i.7] W3C® Recommendation: "W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures".

NOTE: Available at https://www.w3.org/TR/xmlschema11-1/.

[i.8] JSON Schema.

NOTE: Available at https://json-schema.org/.

[i.9] Article 29 Working Party: "Guidelines on transparency under Regulation 2016/679".

[i.10] Article 29 Working Party: "Guidelines on consent under Regulation 2016/679".

NOTE: Available at https://edpb.europa.eu.

[i.11] IETF RFC 1034: "DOMAIN NAMES - CONCEPTS AND FACILITIES".

[i.12] Microsoft™ SDL: "Security Development Lifecycle".

NOTE: Available at https://www.microsoft.com/sdl.

[i.13] ISO/IEC 27034-3: "Information technology -- Application security -- Part 3: Application security management process".

[i.14] ISO/IEC 29147: "Information technology -- Security techniques -- Vulnerability disclosure".

[i.15] IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".

[i.16] IETF RFC 7516: "JSON Web Encryption (JWE)".

[i.17] ETSI TR 103 838: "Cyber security; Guide to Coordinated Vulnerability Disclosure".

[i.18] IoT Security Foundation: "Vulnerability Disclosure Best Practice Guidelines".

NOTE: Available at https://www.iotsecurityfoundation.org/best-practice-guidelines/.

[i.19] HackerOne®, vulnerability disclosure service.

NOTE: Available at https://www.hackerone.com/.

[i.20] NCSC: "Setting up two-factor authentication (2FA)".

NOTE: Available at https://www.ncsc.gov.uk/guidance/setting-two-factor-authentication-2fa.

[i.21] NCSC: "Secure development and deployment guidance".

NOTE: Available at https://www.ncsc.gov.uk/collection/developers-collection.

[i.22] ETSI TS 119 312: "Electronic Signatures and Infrastructures (ESI); Cryptographic Suites".

[i.23] NIST SP800-90A: "Recommendation for Random Number Generation Using Deterministic Random Bit Generators".

[i.24] AIS 20/31: "A proposal for: Functionality classes for random number generators".

[i.25] ANSI/ISA-62443: "Security for industrial automation and control systems".

[i.26] IETF RFC 7235: "Hypertext Transfer Protocol (HTTP/1.1): Authentication".

[i.27] IEEE 802.11w™-2009: "IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 4: Protected Management Frames".

[i.28] IEEE 802.11i™-2004: "IEEE Standard for information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 6: Medium Access Control (MAC) Security Enhancements".

[i.29] IETF RFC 6238: "TOTP: Time-Based One-Time Password Algorithm".

[i.30] Directive (EU) 2019/770 of the European Parliament and of the Council of 20 May 2019 on certain aspects concerning contracts for the supply of digital content and digital services.

NOTE: Available at https://eur-lex.europa.eu/.

[i.31] Mayhew, Joe, and Hamid Jahankhani: "Current Challenges of Modern-Day Domestic Abuse", Policing in the Era of AI and Smart Societies. Springer, Cham, 2020. 267-282.

[i.32] Datta Burton, S. et al.: "The UK Code of Practice for Consumer IoT Cybersecurity: where we are and what next" (2021).

[i.33] Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation).

[i.34] FIPS PUB 140-2 (2001): "Security Requirements for Cryptographic Modules".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI EN 303 645 [i.1] apply.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI EN 303 645 [i.1] and the following apply:

ACM	Agreed Cryptographic Mechanisms
AES	Advanced Encryption Standard
AIS	Application notes and Interpretation of the Scheme
ANSI	American National Standards Institute
ARM	Advanced RISC Machine
BLE	Bluetooth® Low Energy
CBC	Cipher Block Chaining
CCM	Counter with CBC-MAC
CPU	Central Processing Unit
CSA	Coordination and Support Action
CSPRNG	Cryptographically Secure Pseudorandom Number Generator
DNS	Domain Name System
DTLS	Datagram Transport Layer Security
ECDH	Elliptic Curve Diffie-Hellman
ECDHE	Elliptic Curve Diffie-Hellman Ephemeral
ECDSA	Elliptic Curve Digital Signature Algorithm
ECRYPT	European Network of Excellence for Cryptology
EN	European Standard
FAQ	Frequently Asked Question
FIPS	Federal Information Processing Standards
GCM	Galois/Counter Mode
GHz	GigaHertz
HTTP	Hypertext Transfer Protocol
IEC	International Electrotechnical Commission
IETF	Internet Engineering Task Force
IoT	Internet of Things
ISA	International Society of Automation
IV	Initialization Vector
JSON	JavaScript Object Notation
KDF	Key Derivation Function
MAC	Message Authentication Code
PIN	Personal Identification Number
PKI	Public Key Infrastructure
PSK	Pre-Shared Key

RAM	Random-Access Memory
RFC	Request for Comments
RSA	Rivest, Shamir and Adleman
SAE	Simultaneous Authentication of Equals
SHA	Securing Hash Algorithm
SMS	Short Message Service
SOGIS	Senior Officials Group on Information Security
TOTP	Time-based One-Time-Password
TPM	Trusted Platform Module
WAN	Wide Area Network
WLAN	Wireless LAN
WPA	Wi-Fi® Protected Access
XOR	exclusive OR
XTS	XOR-Encrypt-XOR-Based Tweaked-Codebook Mode with Ciphertext Stealing

4 Using the present document

4.1 Purpose

The present document provides guidance to implement the provisions in ETSI EN 303 645 [i.1] in the form of examples illustrating possible solutions. The intent is to help implementers better understand how each provision can be met. It is reminded that ETSI EN 303 645 [i.1] and ETSI TS 103 645 [i.2] both provide guidance text and examples. These can be referred to when considering the examples provided herein. These examples are provided in clauses 6 and 7.

In ETSI EN 303 645 [i.1] and ETSI TS 103 645 [i.2] recommendations are expected to be followed by manufacturers unless there exists a justification for not doing so. Examples of situation where it might be difficult to follow a given recommendation are provided in clause 8.

4.2 Relationship to ETSI EN 303 645

The examples provided in the present document are tailored to the outcome-focused nature of ETSI EN 303 645 [i.1]. It is acknowledged that ETSI EN 303 645 [i.1] can be specialised into more precise domains of applicability, for example smart locks. In such case, the example solutions to meet the new set of provisions can be better tailored to this specific IoT domain. It is expected that these examples would be included in an update to the present document, or to a future, dedicated guidance document.

4.3 Relationship to ETSI TS 103 701

ETSI TS 103 701 [i.3] provides a framework for the assessment of the provisions defined in ETSI EN 303 645 [i.1]. As such, they can be used (when implemented) to inform the definition of test scenarios and the development of a test plan based on ETSI TS 103 701 [i.3]. As described in clauses 4.1 and 4.2, a specialisation of ETSI EN 303 645 [i.1] for a specific application domain can allow more precise examples to be provided which, when implemented, would allow more precise test scenarios and test plan, leading to stronger certainty on the test expectations and outcomes.

5 Guidance on implementation

Clauses 6 and 7 provide examples for implementing the provisions laid out in ETSI EN 303 645 [i.1]. The examples provided herein are not exhaustive or limitative; it is possible to meet the provisions in ETSI EN 303 645 [i.1] by using other solutions or variants of the provided examples.

For each provision, the examples are not given in a specific qualitative order (therefore, example 1 is not necessarily qualitatively better than example(s) 2 or 3) but are ordered by usage, from the most widely applicable examples to the more specific ones. While most examples are meant to cover a large spectrum of IoT cases, those that relate to a specific type of IoT device, constrained device, or a given use case are indicated as such.

The examples in clauses 6 and 7 of the present document provide guidance on the implementation of cybersecurity mainly protecting users from unknown other users. However, consumer IoT devices can be misused, e.g. by intimate partners, which makes it even more difficult to find appropriate security measures.

The following list provides examples of related threats using consumer IoT devices:

- audio control (i.e. recording and/or replying);
- video control (i.e. recording and/or displaying);
- data control (i.e. collection, manipulation, unintended disclosure);
- access to shared accounts linked to the consumer IoT device and therefore providing the possibility for social stalking (e.g. social media);
- other remote control threats (e.g. heating control, door lock control).

The exploitation of the aforementioned threats might result in coercive control (e.g. isolation from friends and family, spying, deprivation of vital and basic means such as medical services and food, controlling finances, etc.).

In this regard, the following measures could help to mitigate domestic abuse by using consumer IoT devices [i.31]:

- introduction of legal policy to prosecute abusers and protect victims of domestic abuse in cases of digital coercive control;
- development of technology that at least provides evidence of activity in cases of domestic abuse, however, without violating data protection/privacy regulations;
- creation of awareness to improve prevention, and establish contact possibilities in cases of domestic abuse to support the victims by providing appropriate advice.

Privacy and data protection related legislation only cover a small part of the first of the aforementioned measures as requirements are set against personal data in terms of transparency in processing as well as purpose limitation, accuracy, integrity and confidentiality.

The second item referring to technology depends on the specific application scenario and can be considered during the development of corresponding solutions or verticals based on ETSI EN 303 645 [i.1] appropriate to the properties of the technology, risk, benefit and usage. Designing ways to prevent the misuse of the device intends to mitigate the risk of its misuse, but it is understood that it might not eliminate the risk entirely.

The item regarding awareness can be realized by training and by reaching out to all stakeholders of consumer IoT products, for instance by advertisements or by including the domestic abuse matter into guidelines as in the present document.

NOTE: The UK Code of Practice for Consumer IoT Security [i.32] provides statistics and references to the current state of the art regarding research in consumer IoT tech abuse. It also references to industry actors providing guidelines to prevent that technology is being used for domestic abuse.

6 Examples to meet cyber security provisions for consumer IoT

6.1 Provision 5.1-1

"Where passwords are used and in any state other than the factory default, all consumer IoT device passwords shall be unique per device or defined by the user". (ETSI EN 303 645 [i.1])

NOTE 1: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The consumer IoT device password for the factory default state is printed on a sticker under the device casing. During the initialization phase, the user is requested to provide a new password and the procedure cannot complete without the new password being different from the default state password.

EXAMPLE 2: The consumer IoT device has no password in the factory default state and generates a password for the user during the initialization phase. The device is not constrained and the password generation process is based on a cryptographically secure pseudorandom number generator where the entropy source is an on-device ring oscillator.

NOTE 2: For simpler and constrained devices, a less advanced MCU does not provide the same cryptographic acceleration and entropy.

EXAMPLE 3: The consumer IoT device prompts a user to create a password, choosing any complexity requirements such that a user can create a memorable password with a strength appropriate to the device's capabilities and the security necessary for the application.

6.2 Provision 5.1-2

"Where pre-installed unique per device passwords are used, these shall be generated with a mechanism that reduces the risk of automated attacks against a class or type of device". (ETSI EN 303 645 [i.1])

NOTE 1: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The password is generated using a cryptographically secure pseudorandom number generator present on a chip in the device. The password complexity is such that the password cannot be guessed through an exhaustive search attack (including optimized variants such as dictionary attacks) via the quickest authentication method available on the device, at least during its expected lifetime. The password is concatenated with a per-device salt and hashed. The device uses a well-known widely implemented hashing algorithm with no known weaknesses, appropriate to the device's capabilities and the security necessary for the application.

NOTE 2: A guidance on the choice of hashing algorithms is given in ETSI TS 119 312 [i.22].

EXAMPLE 2: The manufacturer generates pre-installed passwords within the factory environment using a critical security parameter, stored within a hardware security module, concatenated with either the serial number or MAC address of the device. The resultant string is hashed and the last 8 bytes, as represented in hexadecimal to be human-readable, are taken to be the default password for that device.

EXAMPLE 3: The device is provisioned with a private-public key pair. Upon initialization, a critical security parameter is sent from the factory environment to the device, encrypted with the public key of the device. The device decrypts this message to get the critical security parameter. The device concatenates the critical security parameter with either the serial number or MAC address of the device. The resultant string is hashed and the last 8 bytes, as represented in hexadecimal to be human-readable, are taken to be the device password. This allows a manufacturer to update the device's password throughout its lifetime using a cryptographically secure generation method which would not be available on the device.

6.3 Provision 5.1-3

"Authentication mechanisms used to authenticate users against a device shall use best practice cryptography, appropriate to the properties of the technology, risk and usage". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The authentication protocol between the device control application and the device is protected by TLS 1.2 [i.15] or a higher version, using cipher suites and other security parameters in recommendations issued by a governmental agency for security or by prominent industry crypto-catalogues adapted to the usage context of the device, considering the date of implementation and the expected lifecycle of the device.

EXAMPLE 2: The consumer IoT device provides a password-based user authentication mechanism, so that the user can login and connect to the consumer IoT device using a mobile application. The authentication mechanism is IP-based and uses the HTTP authentication framework (IETF RFC 7235 [i.26]), and TLS 1.2 with a PKI-based authentication is implemented. The consumer IoT device stores two certificates imported during the manufacturing process; one is an individual client certificate (Signature Algorithm ECDSA, key length 224, SHA-224) and the other is the root certificate (Signature Algorithm ECDSA, key length 224, SHA-224) of the PKI. The TLS 1.2 cryptographic configuration for the connection is TLS_ECDHE_RSA_WITH_AES_128_ GCM_SHA256. Within the connection establishment, the IoT device and the mobile application exchange their certificates and verify them against the root certificate, ensuring the authenticity of the connection. The AES session encryption with 128 bits key length in GCM mode ensures the confidentiality and integrity of the data exchange. The cryptography is suitable for the corresponding use case regarding the needed security guarantees. There are no published feasible attacks with regard to current readily available techniques (State 17 February 2021). The used cryptography is part of the public cryptographic catalogue SOGIS-ACM (State 17 February 2021).

EXAMPLE 3: The consumer IoT device provides a password-based user authentication mechanism via an IP-based web interface. The IP connection is offered wireless (Wi-Fi® protocol per default) or wired (Ethernet). The underlying use case is that the user connects to the consumer IoT device, located in the home, using a web browser for the login. The authentication mechanism is implemented via HTTP authentication framework (IETF RFC 7235 [i.26]) and WPA2 (IEEE 802.11i [i.28]) with Protected Management Frames support (IEEE 802.11w™ [i.27]). The WPA2 implementation uses AES/CCMP. For wireless connection, WPA2 with the out-of-band exchange of the network key ensures that the communication is authenticated. The AES session encryption with 128 bits key size ensures the security guarantees, confidentiality and integrity of the password transmission. The cryptography is suitable for the corresponding use case regarding the needed security guarantees. There are no published feasible attacks with regard to current readily available techniques (State 17 February 2021). The used cryptography is part of the public cryptographic catalogue ECRYPT CSA and compliant to FIPS PUB 140-2 [i.34] (State 17 February 2021). For a wired connection, the authenticity and confidentiality are ensured by physical measures and network separation (LAN/WAN), and the risk assessment model that only trustworthy devices operate in the home environment. Therefore, the cryptography is suitable for the corresponding use case.

NOTE: Example 3 bases on the assumption of a trustworthy home environment where the connection to the consumer IoT device using a physical interface is considered to be secure.

EXAMPLE 4: Password-based authentication schemes for connection to a wireless network rely on cryptographic schemes ensuring forward secrecy to prevent offline brute-force attacks of the password based on wireless traffic that has been eavesdropped upon then recorded (e.g. WPA3-SAE was introduced to prevent offline attacks on a recorded Wi-Fi® exchange relying on WPA2-PSK).

6.4 Provision 5.1-4

"Where a user can authenticate against a device, the device shall provide to the user or an administrator a simple mechanism to change the authentication value used".(ETSI EN 303 645 [i.1])

NOTE: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: During the initialization process, a user is asked to provide a PIN for password reset purposes. A user can reset the password for their consumer IoT device using this PIN through one or more of the following methods: online, through an application, via email or a dedicated phone hotline. Each of these methods triggers an email or SMS sent to the user's registered details, which contains a link to reset the password.

EXAMPLE 2: The device's Graphical User Interface provides each user a prominent widget on the home screen to manage user authentication values. The widget takes the user to a management process, which is limited to the steps required to change the value, including re-authentication if necessary. When the user is an administrator, they can go through the same process to change their own authentication value or that of another user.

EXAMPLE 3: There is a reset button on the device that, if a user is unable to authenticate with the device, can be pressed to revert the device to its default factory-set authentication value. The user is then prompted to re-perform the initialization phase and change this authentication value before the device will work.

6.5 Provision 5.1-5

"When the device is not a constrained device, it shall have a mechanism available which makes brute-force attacks on authentication mechanisms via network interfaces impracticable".(ETSI EN 303 645 [i.1])

NOTE: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The device implements a throttling mechanism which, starting from a certain number of authentication failures, introduces a random delay between allowed authentication attempts over the network interface protocol, where the minimal duration of the random delay renders brute-force attacks impracticable (considering the complexity offered by the authentication factor), and the maximum value of the random delay is such that usability is not negatively impacted.

EXAMPLE 2: The device implements two-factor authentication [i.20], which makes authentication attempts via network interface impossible even if the correct credentials are brute-forced.

6.6 Provision 5.2-1

"The manufacturer shall make a vulnerability disclosure policy publicly available. This policy shall include, at a minimum:

- *contact information for the reporting of issues; and*
- *information on timelines for:*

1) initial acknowledgement of receipt; and

2) status updates until the resolution of the reported issues". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The manufacturer's website has a separate page for its vulnerability disclosure policy. This policy contains the information required in Provision 5.2-1, following the ETSI TR 103 838 [i.17], guide to vulnerability disclosure, or following ISO/IEC 29147 [i.14]. The web page is referenced in product documentation and the policy is directly linked within the support section of the manufacturer's website.

EXAMPLE 2: The manufacturer uses a third-party vulnerability reporting service to manage vulnerabilities, either specific to IoT (such as IoT Security Foundation [i.18]) or more general (such as HackerOne® [i.19]). The manufacturer chooses this professional offering to remain up-to-date on CVD best practice and avoid the resource needed to create a bespoke policy. The manufacturer's website contains a web page, discoverable under "security" or "support", that contains a link to the details of this third-party offering. The web page is also referenced in product documentation.

6.7 Provision 5.2-2

"Disclosed vulnerabilities should be acted on in a timely manner". (ETSI EN 303 645 [i.1])

EXAMPLE: A manufacturer's vulnerability handling process stipulates the timeframe in which vulnerabilities will be acknowledged and assessed, to allow resolution of the vulnerability to begin. A 90-day period, from receiving the disclosure to resolution for simple software vulnerabilities, is industry standard, but this can be longer depending on the vulnerability reported and its provenance.

NOTE: Vulnerabilities in hardware usually take longer to resolve than software vulnerabilities. Vulnerabilities in open-source libraries can affect many products and in complicated cases a responsible co-ordinated disclosure can take longer. However, a 90-day resolution period is attainable for most simple software vulnerabilities.

6.8 Provision 5.2-3

"Manufacturers should continually monitor for, identify and rectify security vulnerabilities within products and services they sell, produce, have produced and services they operate during the defined support period".
(ETSI EN 303 645 [i.1].)

EXAMPLE 1: With subject to user's consent as described in clause 6 of ETSI EN 303 645 [i.1], a manufacturer connects to the device to check diagnostic information and help to maintain products. Version information is available so the manufacturer can identify when devices require patching. The manufacturer can rectify security vulnerabilities by pushing out updates to devices remotely.

EXAMPLE 2: Security audits and analyses of device telemetry are carried out by the manufacturer regularly. Analysis of bug reports is integrated with the software development process (see clause 6.45 of the present document or Provision 5.6-9 in ETSI EN 303 645 [i.1]).

6.9 Provision 5.3-1

"All software components in consumer IoT devices should be securely updateable". (ETSI EN 303 645 [i.1])

NOTE: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The secure update mechanism supports the device firmware and all third-party applications. The device firmware encompasses code running electronic components such as baseband processors, interface and networking chipsets, and sensors.

EXAMPLE 2: The device or hub accepts trusted updates that are signed by the manufacturer, which cover all device software components. The manufacturer is able to push updates to the device.

6.10 Provision 5.3-2

"When the device is not a constrained device, it shall have an update mechanism for the secure installation of updates".
(ETSI EN 303 645 [i.1])

NOTE: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The device update mechanism relies on a firmware distribution server. The update and firmware delivery protocol between the device and the firmware distribution server is protected using TLS 1.2 or higher and both the server and device are authenticated by means of server and client certificates. The firmware packages are versioned and digitally signed. Conditional to installing a firmware package, the device verifies that the digital signature is valid.

EXAMPLE 2: The manufacturer provides a computer tool so a user can prepare a USB stick as the update medium. The device or hub accepts signed and trusted updates from the USB stick.

6.11 Provision 5.3-3

"An update shall be simple for the user to apply". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The update is unobtrusive and is applied by the device without any user interaction, once configured so by the user.

EXAMPLE 2: The only action required from the user is to decide on a convenient time to install the update. Download, installation, pre- and post-install configuration, reboot (if necessary), and notification that the update successfully completed all happen automatically.

EXAMPLE 3: The manufacturer provides a computer tool to automatically prepare a USB stick as update medium for devices without a user interface, e.g. devices without a graphical user interface. Once the USB stick is inserted into such a device, the update process starts automatically.

6.12 Provision 5.3-4

"Automatic mechanisms should be used for software updates". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: Checking for availability of, downloading, scheduling and applying unobtrusive updates that do not require a device reboot is done without any user interaction by the device.

EXAMPLE 2: Checking for availability of, downloading and applying of updates that are potentially disruptive to the functioning of the device is done without any user interaction at a time that is determined to be convenient for the user.

6.13 Provision 5.3-5

"The device should check after initialization, and then periodically, whether security updates are available". (ETSI EN 303 645 [i.1])

NOTE: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: After initialization the device connects to the update servers and checks for the availability of updates. This process repeats daily at a randomized time to avoid all devices connecting to the server at the same time and overloading it.

EXAMPLE 2: After initialization the device connects to the update servers and checks for the availability of updates. This process repeats at intervals configured by the device user.

EXAMPLE 3: After the initial configuration of the device by the user, the device performs an update check and notifies the user if an update is available. From then on, the device checks for updates every day between 2 and 5 am without user interaction.

6.14 Provision 5.3-6

"If the device supports automatic updates and/or update notifications, these should be enabled in the initialized state and configurable so that the user can enable, disable, or postpone installation of security updates and/or update notifications". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The device is configured in the initialized state to automatically check for the availability of software updates. Once an update is available, the user receives a notification that contains options on how to handle the update and a pointer to the configuration panel for security updates.

EXAMPLE 2: The device default configuration in the initialized state is to automatically check for the availability of software updates, download, and install them. During device initialization, the user is given the opportunity to review and modify the options for handling updates.

EXAMPLE 3: The device supports automatic updates, which are enabled in the default factory settings. The user can deactivate automatic updates, so updates will only be installed with user consent. The user can reactivate automatic updates through the device's web interface. The user can also postpone installation of device updates to a specific time.

EXAMPLE 4: The device supports update notifications, which is enabled in the default factory settings. The user can deactivate and reactivate update notifications through the device's web interface. The user can also postpone the update notifications to a specific time.

6.15 Provision 5.3-7

"The device shall use best practice cryptography to facilitate secure update mechanisms". (ETSI EN 303 645 [i.1])

EXAMPLE 1: An update of the device is integrity protected and signed by the manufacturer, using recommendations issued by a governmental agency for security or by prominent industry crypto-catalogues adapted to the usage context of the device, considering the date of implementation and the expected lifecycle of the device.

EXAMPLE 2: The update delivery protocol between the device and the update server is protected using TLS 1.2 or higher, recommendations issued by a governmental agency for security or by prominent industry crypto-catalogues adapted to the usage context of the device, considering the date of implementation and the expected lifecycle of the device.

6.16 Provision 5.3-8

"Security updates shall be timely". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: Patches to address a security vulnerability are provided to the device in a security update within 90 days of the vulnerability having been notified to the manufacturer.

EXAMPLE 2: Solutions and patches to address a security vulnerability within a widely used software stack or in hardware are actively discussed and validated across stakeholders such that a security update can be provided without undue delay.

EXAMPLE 3: The manufacturer implements a process for providing security updates using pre-defined communication channels, responsibilities and involved parties. In this process the security update creation is triggered within 14 days and the update is published within 90 days.

6.17 Provision 5.3-9

"The device should verify the authenticity and integrity of software updates". (ETSI EN 303 645 [i.1])

NOTE 1: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: Software updates are digitally signed by a signing server before being delivered to the device. Before installation, the device checks the authenticity and integrity of the update using the digital signature and the public key of the signing server. Only updates with successful validation are installed. Otherwise, the software update is not installed and the user is notified about the invalid update.

NOTE 2: Specific security measures can apply to the signing server, such as perimeter protection within a private network.

EXAMPLE 2: The device obtains updates from a hub on a local network. The hub appends a keyed message authentication code to each update, the key being shared by the hub and the device. The device verifies that the message authentication code matches the update it applies to.

EXAMPLE 3: Software updates are versioned, with the version number being part of the signed package. The device checks that the software update received has a higher version number than the one currently installed before flashing it. This is meant to prevent installation of an older version with a valid signature but containing security vulnerabilities.

NOTE 3: There can be valid reasons to allow reverting to an older version of the code base in specific cases. For instance, when functional issues are reported by end-users, after a new firmware has been deployed on the field. To address this point, a process to quickly resign an existing software package with an incremented version number needs to be put in place.

6.18 Provision 5.3-10

"Where updates are delivered over a network interface, the device shall verify the authenticity and integrity of each update via a trust relationship". (ETSI EN 303 645 [i.1])

EXAMPLE 1: Updates are digitally signed by a signing server before being delivered to the device. The device checks the authenticity and integrity of the update using the digital signature and the public key of the signing server.

NOTE 1: The trust relationship is from the device to the signing server, with trust that the manufacturer remains in control of the signing server and will only sign legitimate updates. The device uses the public key of the signing server to verify the digital signature of the update. It is expected that the public key is integrity protected on the device.

NOTE 2: Specific security measures can apply to the signing server, such as perimeter protection within a private network.

EXAMPLE 2: The device obtains updates from a hub on a local network. The hub appends a keyed message authentication code to each update, the key having previously been shared between the hub and the device. The device verifies that the message authentication code matches the update it applies to.

EXAMPLE 3: The device obtains updates from a hub on a local network that is responsible for verifying the authenticity and integrity of the update. Access on the local network requires a shared secret (as with a Zigbee™ group key or Wi-Fi® WPA2-PSK). Through this shared secret, the device assumes that a trusted hub exists on the local network.

NOTE 3: In both examples 2 and 3, the trust relationship is from the device to the hub, where the core responsibility of verifying the authenticity and integrity of the update is that of the hub. The shared secret is a critical security parameter. A secure mechanism is needed to provision the shared secret. Example 3 assumes that other devices on the local network will not tamper with the delivery of the update or impersonate the hub. This implies a strong assumption that other devices (possibly provided by other parties) can be trusted and will not be compromised. Wherever possible, when the device is constrained in such a way that it cannot by itself implement strong authenticity and integrity verification mechanisms, this is expected to be handled by trusted third party that is reachable over a trusted communication path.

EXAMPLE 4: In an IoT setup consisting of a smart home controller and some sensors and actuators, the check for software updates is performed by the controller. Integrity and authenticity of software updates are guaranteed by a cryptographic signature. Before installation, integrity and authenticity are validated by the controller, and only updates with successful validation are transmitted to the sensors and actors within the infrastructure. For this transmission an existing trustworthy channel (that ensures at least integrity and authenticity) is used. The sensors and actors will then install the update without revalidation due to the trust relationship to the controller.

EXAMPLE 5: Integrity and authenticity of software updates are guaranteed by a cryptographic signature. Before the installation, integrity and authenticity are validated by the device, and only updates with successful validation are installed. Otherwise, the update is not installed and the user is notified about the lack of trust in the update. In this case the trust relationship does not involve further devices, but is directly between the device and the provider of software updates.

6.19 Provision 5.3-11

"The manufacturer should inform the user in a recognizable and apparent manner that a security update is required together with information on the risks mitigated by that update". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The manufacturer informs the user that a security update is required on the device via a device companion application on the user's smartphone. A notification is displayed on the smartphone user interface.

EXAMPLE 2: After login on the device's web interface, an obvious notification appears, which informs the user about the availability of a security update. There is a button the user can click to receive more information about the risks mitigated by the update.

EXAMPLE 3: Once an update check reveals the availability of a security update, the user is informed about this via email notification, which also contains information about the risks mitigated by the update. The email address of the user is configured by the user in the initial device setup.

6.20 Provision 5.3-12

"The device should notify the user when the application of a software update will disrupt the basic functioning of the device". (ETSI EN 303 645 [i.1])

NOTE: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE: Before the user confirms the installation of the software update, a warning, which the user acknowledges first, is displayed on the same interface as the confirmation prompt.

6.21 Provision 5.3-13

"The manufacturer shall publish, in an accessible way that is clear and transparent to the user, the defined support period". (ETSI EN 303 645 [i.1]).

EXAMPLE 1: Information about the defined support period is provided prominently in the device documentation.

EXAMPLE 2: The manufacturer publishes that for a new product, released in 2021, the device will receive security updates until the end of 2025. This support period is displayed on the manufacturer website on the product page. In 2023, the manufacturer makes the decision that they will support the device longer, and so change the date at which security updates will cease to the end of 2027. The manufacturer informs their users of this change by updating the online policy document, via a notification within a companion app and/or via email or SMS.

6.22 Provision 5.3-14

"For constrained devices that cannot have their software updated, the rationale for the absence of software updates, the period and method of hardware replacement support and a defined support period should be published by the manufacturer in an accessible way that is clear and transparent to the user". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The information is provided prominently in the device documentation as part of the installation instructions.

EXAMPLE 2: The manufacturer publishes that for a new product, released in 2021, the device will be supported until the end of 2025. This support period and method of hardware replacement is displayed on the manufacturer website on the product page. The manufacturer informs their users of any extension of support period by updating the online policy document, via a notification within a companion app and/or via email or SMS.

6.23 Provision 5.3-15

"For constrained devices that cannot have their software updated, the product should be isolable and the hardware replaceable". (ETSI EN 303 645 [i.1]).

EXAMPLE 1: A group of temperature sensors is networked through a smart hub that provides network and application layer isolation. The interface between the sensor and the smart hub allows for replacement of the sensor, including with newer models.

EXAMPLE 2: A non-updatable sensor records and transmits a certain value every 10 seconds (e.g. temperature value) to a smart home controller. If the sensor is affected by a vulnerability and the sensor is non-updateable, the network connection is detached. The sensor is nevertheless still recording the values, but no longer transmitting them to the controller. Instead, the user can retrieve the values using a physical interface on the device.

6.24 Provision 5.3-16

"The model designation of the consumer IoT device shall be clearly recognizable, either by labelling on the device or via a physical interface". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: A sticker is attached to the back of the device that provides the model designation.

EXAMPLE 2: An administrative command line interface is available over a dedicated physical port. The model designation is displayed prior to authentication on the welcome banner of this command line interface.

6.25 Provision 5.4-1

"Sensitive security parameters in persistent storage shall be stored securely by the device". (ETSI EN 303 645 [i.1])

NOTE 1: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: A public key that is otherwise recoverable is protected in confidentiality and integrity via a key wrapping mechanism on the device.

NOTE 2: This does not protect the public key from deletion on the device. The confidentiality protection of the public key is a by-product of using the key-wrapping mechanism.

EXAMPLE 2: The device has an internal key that encrypts sensitive data at rest. This internal key is kept in a secure volatile memory.

NOTE 3: In example 2 the sensitive data is meant to be stored securely in persistent storage.

NOTE 4: Both - the sensitive data and the internal key - need to be stored securely.

EXAMPLE 3: The device's non-volatile memory that stores sensitive security parameters is completely encrypted with AES-128-XTS and en-/decrypted on the fly during its operation. The encryption key is loaded via the Trusted Board Boot Sequence of the underlying ARM chip.

EXAMPLE 4: A secure element is used to store keys and to use these keys further on for cryptographic operations, e.g. for encryption of sensitive data.

6.26 Provision 5.4-2

"Where a hard-coded unique per device identity is used in a device for security purposes, it shall be implemented in such a way that it resists tampering by means such as physical, electrical or software". (ETSI EN 303 645 [i.1])

NOTE 1: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: A smart consumer device for use within a residential property is programmed by the manufacturer to have a serial number to use as an identity. This identity is printed inside the device casing and is protected from physical access, as the device is inside a residential property.

This identity is displayed to the user when granting access to cloud services associated with the user's account, and it is stored in hardware that cannot be reprogrammed, such as by blowing fuses during or after writing. This storage prevents a cloned device being added to the user's account and prevents the device being renamed to one associated with an attacker's account.

EXAMPLE 2: The device identity is hashed and signed by the manufacturer. This signature is verified prior to using the device identity.

NOTE 2: This does not protect the device identity from deletion.

EXAMPLE 3: A hard-coded device identity is stored in a secure element of the device, not in software, and is accessed through a dedicated read-only interface. After being retrieved and used for authorization, it is deleted from temporary memory.

EXAMPLE 4: The device is enrolled with a local manager. At enrolment, in addition to the hard-coded identity, the local manager accepts a public key generated by the device. The local manager provides the device with a certificate to use for security purposes containing the enrolled device identity and public key.

NOTE 3: The private key that corresponds to the enrolled public key is a critical security parameter and falls in the scope of Provision 5.4-1.

6.27 Provision 5.4-3

"Hard-coded critical security parameters in device software source code shall not be used". (ETSI EN 303 645 [i.1])

EXAMPLE 1: Critical security parameters are stored on a secure element of the device and are accessible from the software through a dedicated interface. They are not, and never have been, hard-coded into the device software source code.

EXAMPLE 2: In the absence of a secure element, critical security parameters are stored in flash encrypted with a key derived in RAM when needed from a combination of device-unique hardware parameters that are private (not exposed through network services, not printed on a sticker).

6.28 Provision 5.4-4

"Any critical security parameters used for integrity and authenticity checks of software updates and for protection of communication with associated services in device software shall be unique per device and shall be produced with a mechanism that reduces the risk of automated attacks against classes of devices". (ETSI EN 303 645 [i.1])

NOTE 1: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: A manufacturer produces critical security parameters for its devices using a mechanism that reduces the risk of automated attacks. Random numbers generated by the manufacturer are concatenated with either the serial number or MAC address of the device. This string is then hashed before being used as input to a Key Derivation Function (KDF).

EXAMPLE 2: A different random string is deployed on each device of the same product class for the initial seeding of the device CSPRNG. The random string is produced according to the requirements of the CSPRNG as described in NIST SP800-90A [i.23] and AIS 20/31 [i.24]. This ensures parameters generated on the device are unique per device and are produced with a mechanism that reduces the risk of automated attacks against classes of devices.

NOTE 2: The initial seeding of the device CSPRNG is meant to provide an adequate level of entropy for the device CSPRNG to produce random numbers from the very first start of the device. The CSPRNG seed is then updated through secure entropy gathering mechanisms. Leaving the CSPRNG seed static introduces a security vulnerability.

EXAMPLE 3: Each device has a unique secret key, or a unique private key, assigned to it by the manufacturer. The key is stored in a dedicated cryptographic microprocessor or a TPM. The secret key is used to validate the Message Authentication Codes (MACs) that are appended to software updates by the manufacturer. The private key is used by the device to attest its identity to an update service provided by the manufacturer.

NOTE 3: ETSI EN 303 645 [i.1] describes when the device uses the manufacturer's public key to verify a software update. In this case, this public key is not a critical security parameter and does not need to be unique per device.

6.29 Provision 5.5-1

"The consumer IoT device shall use best practice cryptography to communicate securely". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The device uses TLS and/or DTLS to communicate with associated services, and the software library providing these secure protocols is configured with cipher suites and other security parameters according to the recommendations issued by a governmental agency for security or by prominent industry crypto-catalogues adapted to the usage context of the device.

EXAMPLE 2: The consumer IoT device provides an IP-based http connection to an associated service from the manufacturer. The underlying use case is that the associated service enables the user to control functionality of the IoT device via a web-service. The connection is implemented via http with underlying TLS 1.2 using PKI-based authentication. The consumer IoT device stores two certificates imported within the manufacturing process. An individual client certificate (Signature Algorithm ECDSA, key length 224 bit, SHA-224) and the root certificate (Signature Algorithm ECDSA, key length 224 bit, SHA-224) of the used PKI. The TLS 1.2 cryptographic configuration for the connection is TLS_ECDHE_RSA_WITH_AES_128 _GCM_SHA256. Within the connection establishment, both parties (IoT device and associated services) exchange their certificates and verify it against the root certificate. This ensures the authenticity of the connection. The AES session encryption wit 128 bits key length in GCM Mode ensures the confidentiality and integrity of the data exchange. The cryptography is suitable for the corresponding use case regarding the needed security guarantees. There are no published feasible attacks with regard to current readily available techniques (State 17 February 2021). The used cryptography is part of the public cryptographic catalogue SOGIS-ACM (State 17 February 2021).

EXAMPLE 3: The consumer IoT device provides a wireless connection to the control hub via a smart home protocol. The underlying use case is that the IoT device is a thermostat controlled by a hub, e.g. to set a temperature. The smart home protocol is the base for the communication. The cryptographic configuration ECDH (256 bits key length) for key exchange and AES with the key length of 128 bits for the payload encryption. The thermostat sends a public key to the hub starting to build a trust relation. The thermostat masks two byte of its public key with zeros. The masked bytes are displayed on the thermostat. The user enters this value (PIN) into the hub manually to reconstruct the complete public key as base for the trust relation. This out-of-band procedure fulfils the security guarantee authenticity. The AES encryption of the payload with 128 bits key length in CCM Mode ensures the security guarantees confidentiality and integrity. A message counter with sufficient size is increased monotonically and also used as an IV for the AES CCM encryption, which ensures that both replay of older messages and message decryption due to IV reuse are not possible. The cryptography is suitable for the corresponding use case regarding the needed security guarantees. There are no published feasible attacks with regard to current readily available techniques (State 17 February 2021). The used cryptography is part of the public cryptographic catalogue SOGIS-ACM (State 17 February 2021).

6.30 Provision 5.5-2

"The consumer IoT device should use reviewed or evaluated implementations to deliver network and security functionalities, particularly in the field of cryptography". (ETSI EN 303 645 [i.1])

NOTE 1: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The device TLS library uses cipher suites and other security parameters according to the recommendations issued by a governmental agency for security or by prominent industry crypto-catalogues to deliver network and security functionalities. The device TLS library is open-sourced, with many contributors that have reviewed it, and is widely used.

EXAMPLE 2: The device's code makes calls to evaluated built-in cryptographic libraries to fulfil its cryptographic operations. It does not use functions developed in-house.

EXAMPLE 3: Following an initial review of the implementation, the manufacturer publishes the documentation and software libraries for a device API and puts out a bug bounty programme for security analysts and advanced users to conduct vulnerability research.

NOTE 2: Example 3 is about performing continuous review of the code implementation by utilizing vulnerability disclosures. Hence, example 3 also conforms to Provision 5.2-1 of the ETSI EN 303 645 [i.1].

EXAMPLE 4: The manufacturer makes use of in-house developed cryptographic functions in an IoT product. These cryptographic functions follow state of the art and industry best practice, appropriate to the properties of the technology, risk and usage.

6.31 Provision 5.5-3

"Cryptographic algorithms and primitives should be updateable". (ETSI EN 303 645 [i.1])

EXAMPLE 1: Software libraries providing cryptographic algorithms and primitives are part of the device firmware image, which can be securely updated. A device can update its algorithms and primitives as part of its regular update cycle.

EXAMPLE 2: The software components using cryptographic algorithms and primitives are developed with crypto agility in mind and are updateable. The manufacturer can provide an update, which changes the cryptographic algorithms, key lengths, or the primitives under consideration, without any user interaction.

6.32 Provision 5.5-4

"Access to device functionality via a network interface in the initialized state should only be possible after authentication on that interface". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The manufacturer issues client certificates to peers that require access to device functionalities over TLS. The certificates are unique per peer and are based on unique information about the peer such as an identity. The on-device server authenticates the client certificate and device functionality is available after the TLS handshake completes successfully.

EXAMPLE 2: Device functionality is only accessible after successful authentication against a Wi-Fi® network provided and controlled by the device.

NOTE: When the device itself is in control of the provided Wi-Fi® network, in example 2, the use of the Wi-Fi® credentials is to be understood as authentication method with a low level of security. This example does not refer to devices such as home gateways, it refers to devices such as wireless non-medical thermometers, that might not require a high level of security. An additional web user interface log in for authentication on such IoT devices would reflect a too low effort-benefit as such IoT devices are not security critical.

EXAMPLE 3: Device functionality is only accessible after successful authentication, which can be password-based, public-key-based or PSK-based.

6.33 Provision 5.5-5

"Device functionality that allows security-relevant changes in configuration via a network interface shall only be accessible after authentication. The exception is for network service protocols that are relied upon by the device and where the manufacturer cannot guarantee what configuration will be required for the device to operate". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The device configuration protocol requires authentication of the configuration peer by the device, and of the device by the configuration peer.

EXAMPLE 2: The device allows changing the network settings using a web interface. For instance, the user can choose the WLAN the device is connected to, which is a security-relevant change. The execution of the change requires the user to authenticate with their credentials.

6.34 Provision 5.5-6

"Critical security parameters should be encrypted in transit, with such encryption appropriate to the properties of the technology, risk and usage". (ETSI EN 303 645 [i.1])

EXAMPLE 1: Critical security parameters exchanged between the device and a management server are protected end-to-end using JSON Web Encryption [i.16].

EXAMPLE 2: A pre-shared key is provisioned onto the device during manufacture, which is subsequently used to encrypt critical security parameters in transit.

6.35 Provision 5.5-7

"The consumer IoT device shall protect the confidentiality of critical security parameters that are communicated via remotely accessible network interfaces". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The communication of critical security parameters between remotely accessible network interfaces and the device is protected by TLS 1.2 or higher, using cipher suites and other security parameters according to the recommendations issued by a governmental agency for security or by prominent industry crypto-catalogues adapted to the usage context of the device.

EXAMPLE 2: Where the device allows the user to update their password via an app on their smartphone connected over a Wi-Fi® connection, the updated value is protected by TLS 1.2 or higher when sent to the device.

EXAMPLE 3: At configuration time a mobile phone is used to send the Wi-Fi® key of a home gateway to a device using BLE. To protect against intrusion on the LAN by someone eavesdropping on the BLE exchanges, the Wi-Fi® key is encrypted at the application layer using a public encryption key extracted from a valid certificate that was presented by the device.

6.36 Provision 5.5-8

"The manufacturer shall follow secure management processes for critical security parameters that relate to the device". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The manufacturer implements a secure key management process that covers key generation, key provisioning, storage and updates, key decommissioning, archival and destruction, and processes to handle key expiration and compromise. Where relevant, critical security parameters are protected using cryptographic algorithms and other security parameters according to the recommendations issued by a governmental agency for security or by prominent industry crypto-catalogues adapted to the usage context of the device. For instance, any managed keys that are stored are protected using an algorithm appropriate to the risk profile. This includes encryption or only storing hashes of passwords, rather than the passwords themselves.

EXAMPLE 2: Certificates residing on the device are signed by a Certificate Authority managed by the manufacturer, who has a secure management policy to operate the Certificate Authority including restricting access to signing keys. For instance, the manufacturer utilizes the standard ANSI/ISA-62443 [i.25] for management processes, if applicable to the consumer IoT device. This covers the secure generation of parameters (including coding standards needed for this), the environment security in context of these parameters, the storing and update management, as well as the defined testing environment and test procedures applied to the security parameters.

6.37 Provision 5.6-1

"All unused network and logical interfaces shall be disabled". (ETSI EN 303 645 [i.1])

NOTE: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The status of network interfaces, configuration of logical interfaces provided by software services, and configuration of filtering mechanisms are all the result of an exhaustive analysis conducted to identify the connectivity requirements of each software service present on the device. Based on this analysis, all unused interfaces are disabled.

EXAMPLE 2: Network interfaces, services, options in protocol stacks are activated, and additional packet filtering policies added, in accordance with the expected services provided by the device and with connectivity requirements of each service. By default, these are disabled, and the default packet filtering policy is dropped. The user can manually reactivate any interface.

EXAMPLE 3: The device manufacturer identifies all available network and logical interfaces and decides on those that are necessary for the operation of the device. The remaining interfaces are disabled, by default. The user can manually reactivate any interface.

6.38 Provision 5.6-2

"In the initialized state, the network interfaces of the device shall minimize the unauthenticated disclosure of security-relevant information". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The device welcome banner and welcome web page only provide generic information that cannot be traced back to software or hardware versions used by the device, a pointer to publicly available information hosted by the device, and a function to authenticate.

EXAMPLE 2: The device manufacturer identifies all security-relevant information held on the device, and what comprises the minimum amount to be disclosed in the initialized state to allow functionality and authentication. By default, only this minimum amount of information is disclosed until authentication has taken place. The user can, after authentication, manually customize what security-relevant information can be disclosed.

EXAMPLE 3: Data denoted as security-relevant is flagged and is not sent on a network interface if authentication has not occurred, unless deemed essential by the device manufacturer or a previously authenticated user.

6.39 Provision 5.6-3

"Device hardware should not unnecessarily expose physical interfaces to attack". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: Physical interfaces that are unused in regular use of the device are protected by the device casing.

EXAMPLE 2: Physical interfaces and physical air interfaces that are not permanently needed are disabled by default and can be enabled and disabled through a trusted mechanism.

EXAMPLE 3: Multi-purpose interfaces are configured only for the purpose(s) they serve with the device.

EXAMPLE 4: The JTAG interface on the device's main board is deactivated in software.

6.40 Provision 5.6-4

"Where a debug interface is physically accessible, it shall be disabled in software". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: All JTAG and SWD interfaces are disabled in software by configuring the appropriate control bit(s) accordingly.

EXAMPLE 2: During development phase the manufacturer debugs the product using a UART interface that gives access to a bootloader menu accessible through a key press at power on of the device, and a Linux® shell accessible at any time once the system has been started. Before the device is shipped on the field both bootloader and firmware are updated to commercial software that removes those interactive menus.

6.41 Provision 5.6-5

"The manufacturer should only enable software services that are used or required for the intended use or operation of the device". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: All installed software services are registered in a service management framework and disabled by default. On device start, only those software services that are necessary for the operation of the device are activated.

EXAMPLE 2: A manufacturer sends updates to the device that re-enable previously disabled software services, as initial assumptions about device usage and software services needed for its operation change over time.

EXAMPLE 3: The user can re-enable software services throughout a device's lifetime, to allow for new features and software services that improve performance of the device. This mechanism is used in cases where the manufacturer cannot push appropriate updates remotely, or the device cannot process them without user intervention.

6.42 Provision 5.6-6

"Code should be minimized to the functionality necessary for the service/device to operate". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: Software libraries are compiled only with the options necessary for the device operation.

EXAMPLE 2: Chosen security algorithms and parameters, such as key sizes, are appropriate to the device's risk profile and processing requirements; they run in reasonable time and with appropriate memory capacity. The code contains only these algorithms and parameters, no other cryptographic parameters are implemented.

EXAMPLE 3: The complexity of functional mechanisms, such as recovery mechanisms that allow a user to regain access to the device, is limited to the necessary functionality that is appropriate for the device. Before release, the code is checked for unused functions that are not needed for the current mode of operation.

6.43 Provision 5.6-7

"Software should run with least necessary privileges, taking account of both security and functionality". (ETSI EN 303 645 [i.1]).

NOTE: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The software is structured in modules. Only the control module runs under a privileged user, all other modules run under an unprivileged user and are sandboxed such that it is allowed access to a limited set of system calls and other resources such as files, that correspond to the module's requirements.

EXAMPLE 2: For most use cases, user level of privileges is sufficient for the device to run. Admin login grants access to all functionality that has been allowed by the manufacturer.

EXAMPLE 3: On a Linux-based device the Linux capabilities granted to a program run as root are restricted through the use of a Linux Security Module (e.g. AppArmor™, SELinux) or by dropping capabilities in the program source code in order to give only the necessary subset of the full root rights.

6.44 Provision 5.6-8

"The device should include a hardware-level access control mechanism for memory". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The device micro-controller provides a Trusted Execution Environment that is used to store all security relevant data and to perform security relevant processes.

EXAMPLE 2: The device micro-controller provides a Memory Protection Unit.

6.45 Provision 5.6-9

"The manufacturer should follow secure development processes for software deployed on the device". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The manufacturer implements a standard or methodology to develop in-house software, such as the Security Development Lifecycle [i.12] methodology or ISO/IEC 27034-3 [i.13], or Secure development and deployment guidance [i.21].

EXAMPLE 2: The manufacturer uses reviewed code, such as open-sourced or widely available code or, for in-house development, paired-programming and peer-reviewing practices.

EXAMPLE 3: The manufacturer creates tests for software written, including negative testing of code, and automated unit and integration tests.

EXAMPLE 4: The manufacturer relies on a toolchain that includes the gcc compiler. The following security flags, which are supported by that compiler, are applied at build time to mitigate potential vulnerabilities in the source code:
CFLAGS: "-fpie", "-fstack-protector-all", "-D_FORTIFY_SOURCE=2"
LDLAGS: "-Wl,-z, relro", "-Wl,-z, now"
The manufacturer can also automatically check that those security options were applied correctly by running automated tools on the firmware.

6.46 Provision 5.7-1

"The consumer IoT device should verify its software using secure boot mechanisms". (ETSI EN 303 645 [i.1])

EXAMPLE: The device uses an authenticated boot mechanism provided by the CPU. A user can view the health of these boot mechanisms, by displaying the version and the results of the device start-up tests. The firmware digital signature is verified using a public cryptographic key that is stored in protected memory. To preserve the chain of trust, software assets that are not part of the firmware are authenticated by an agent in the firmware.

6.47 Provision 5.7-2

"If an unauthorized change is detected to the software, the device should alert the user and/or administrator to the issue and should not connect to wider networks than those necessary to perform the alerting function". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The manufacturer digitally signs updates and a device is configured to expect updates in this format. The device verifies these updates as follows. The device has a boot mechanism that checks for errors to the updates and, if errors are found, the device recovers safely and remotely by reverting to a known good state, determined by back-ups and telemetry. Additionally, an alert can be sent to the user, dependent on the manufacturer's risk assessment (that the benefit from this alert offsets or outweighs the extra attack surface it creates).

EXAMPLE 2: The device attempts to verify its firmware during the authenticated boot mechanism presented in example 1 of clause 6.46 (Provision 5.7-1). The check completes but the result shows an error because the verification fails. Therefore, the device enters a failsafe mode. One possible fail-safe mode is where the offending software is not activated, and an alert is sent to the management dashboard. Another possible fail-safe mode allows the device to load an alternative firmware from read-only memory that deactivates all its functions apart from a visual indicator of compromise on the device casing.

6.48 Provision 5.8-1

"The confidentiality of personal data transiting between a device and a service, especially associated services, should be protected, with best practice cryptography". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The device runs application protocols to access services on top of a secure transport protocol such as TLS [i.4] or DTLS [i.5]. Security parameters, such as allowed cipher suites, are configured and used regarding data flows that contain personal data according to government or industry security recommendations and are adapted to the usage context of the device.

EXAMPLE 2: All data on the device that is categorized as personal data, such as email addresses and real names, are stored with an *encryptionRecommended* flag, which ensures encryption is used, where supported, when this data transits between a device and a service.

EXAMPLE 3: The device shares an identifier, based on its IP address, with the manufacturer for periodic maintenance purposes. The device and manufacturer use a pre-shared key to protect the identifier.

6.49 Provision 5.8-2

"The confidentiality of sensitive personal data communicated between the device and associated services shall be protected, with cryptography appropriate to the properties of the technology and usage". (ETSI EN 303 645 [i.1])

EXAMPLE 1: An IP camera backs up a video content stream to the cloud; this is sensitive personal data. This video data is sent over a connection between the device and associated services, which is always established using a secure transport protocol such as TLS [i.4] or DTLS [i.5]. Allowed cipher suites and other security parameters are configured according to recommendations issued by a governmental agency for security or by prominent industry crypto-catalogues adapted to the usage context of the device.

EXAMPLE 2: All data on the device categorized as sensitive personal data, such as bank account details, home video surveillance footage and health monitor readings, is stored with an *encryptionRequired* flag, which prevents the data being transitioned to a service without encryption.

6.50 Provision 5.8-3

"All external sensing capabilities of the device shall be documented in an accessible way that is clear and transparent for the user". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The user documentation lists all external sensing capabilities: in the device manual, through the user interface in a prominently visible menu, or on the device packaging.

EXAMPLE 2: The device provides an interface that identifies when each external sensing capability is active, with notifications sent to the user when, for example, a microphone is activated.

6.51 Provision 5.9-1

"Resilience should be built in to consumer IoT devices and services, taking into account the possibility of outages of data networks and power". (ETSI EN 303 645 [i.1])

EXAMPLE 1: A device provides resilience to electrical power outage occurring during the firmware updates by implementing a dual bank flash memory, and a safe firmware update and validation mechanism.

EXAMPLE 2: A temperature sensor has enough memory capacity that it can store results locally and continue measurements while the data collection service is offline.

EXAMPLE 3: A smart speaker has a battery with sufficient capacity to allow continued full functionality and connectivity during short power outages and in relocation of the speaker between power sources. To support this relocation, the speaker also supports quick connection to new networks and scans for new connections (e.g. Bluetooth®, Wi-Fi®) when not connected to a network or device. This functionality allows use and network-based management of multiple speakers and devices throughout a home.

6.52 Provision 5.9-2

"Consumer IoT devices should remain operating and locally functional in the case of a loss of network access and should recover cleanly in the case of restoration of a loss of power". (ETSI EN 303 645 [i.1])

EXAMPLE 1: A household appliance, such as a smart fridge, does not depend on network connectivity to perform its main function of preserving food. A smart fridge will maintain its cooling function as long it has electrical power and will continue to do so during outages of connected network access. Though online features, such as consulting supermarket catalogues and shopping options will be inaccessible during the outage, the fridge can resume these online functions after reconnecting to the network.

EXAMPLE 2: A device acting as an authentication factor uses a method that does not require a network connection (such as a One Time Password) or that relies on proximity detection.

EXAMPLE 3: A device communicates securely with a service to function and caches the credentials to resume a connection as soon as power or network access is restored. To ensure availability, the cryptographic material used is valid for a period that is longer than any expected outage, to avoid mismatched cryptography and optimize reconnection.

6.53 Provision 5.9-3

"The consumer IoT device should connect to networks in an expected, operational and stable state and in an orderly fashion, taking the capability of the infrastructure into consideration". (ETSI EN 303 645 [i.1])

NOTE: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The device uses a lightweight heartbeat protocol against a specialized network host to check for network recovery upon detection of connectivity loss. Upon detecting the restoration of the network connection, the device waits a randomized delay before connecting to their associated services.

EXAMPLE 2: Devices of the same model intended to be used in large numbers within the same routable network implement a local peer-to-peer firmware transfer protocol to limit the load on the delivery infrastructure.

EXAMPLE 3: A device that needs to authenticate to a service or network, and expects frequent outages, can choose to use long-lived keys to avoid repeatedly stressing networks with the extra traffic inherent in an authentication process that requires several round trips.

6.54 Provision 5.10-1

"If telemetry data is collected from consumer IoT devices and services, such as usage and measurement data, it should be examined for security anomalies". (ETSI EN 303 645 [i.1])

NOTE: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: Telemetry data is collated by the manufacturer at a central server, anonymized and aggregated for analytics. This allows a manufacturer to check various metrics, such as the versions of the software running on devices, if updates are successful, and consumer usage of the device.

EXAMPLE 2: Telemetry data describes events of use to the manufacturer. Security anomalies can be represented by a deviation from normal behaviour of the device, as expressed by the monitored indicators, such as an abnormal increase of failed login attempts.

EXAMPLE 3: Telemetry data collected describes the devices status and can be used to identify devices that are configured incorrectly.

6.55 Provision 5.11-1

"The user shall be provided with functionality such that user data can be erased from the device in a simple manner". (ETSI EN 303 645 [i.1])

EXAMPLE: The user interface provides a command which, upon user confirmation, resets the user account to the state of a freshly created account. This includes cancelling device bindings to services and erasing all user data from the device, including personal data, configuration data, and cryptographic material.

6.56 Provision 5.11-2

"The consumer should be provided with functionality on the device such that personal data can be removed from associated services in a simple manner". (ETSI EN 303 645 [i.1])

EXAMPLE 1: For each associated service, the user interface of the consumer IoT device provides a command which, upon user confirmation, requests that the associated service deletes any remotely held user personal data, and performs deletion of any local copy of that data.

EXAMPLE 2: The device's user interface provides a menu option to search for associated services that have been sent personal data, and to request deletion of that data. Upon selecting this delete option, the device sends a simple prompt to confirm the request and then completes the deletion.

6.57 Provision 5.11-3

"Users should be given clear instructions on how to delete their personal data". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The user manual includes a clause describing the steps to perform to delete personal data locally as well as the steps to request deletion of personal data from associated services.

EXAMPLE 2: The user manual describes how to configure remote deletion of personal data from a device in the event of the device being lost or stolen. This remote deletion process can be configured from either an administrator account or a user account.

EXAMPLE 3: The user interface guides the user through the deletion process, whereby the user can choose which categories of data are to be deleted.

6.58 Provision 5.11-4

"Users should be provided with clear confirmation that personal data has been deleted from services, devices and applications". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: The user interface provides a visual indication when personal data has been deleted from the device, and another visual indication when the request to delete personal data from an associated service has been successfully processed. To satisfy legal data retention obligations (including data protection laws), a confirmation email is sent to the user-provided email address once all personal data has been deleted.

EXAMPLE 2: An SMS is sent to the user-provided phone number when personal data has been deleted. This includes when: data has been deleted from the device; the request to delete personal data from an associated service has been successfully processed; or all personal data has been completely deleted by the data controller.

EXAMPLE 3: Following the deregistration of the device, and if the user agrees to it, a backup of personal data is kept for three months after deletion, for easy restoration if the user changes their mind or purchases a new compatible device. After three months, the data is permanently deleted, but an SMS reminder is sent to the user-provided phone number one week before this final deletion. An additional SMS is sent after the data has been deleted.

6.59 Provision 5.12-1

"Installation and maintenance of consumer IoT should involve minimal decisions by the user and should follow security best practice on usability". (ETSI EN 303 645 [i.1])

NOTE: An example for this provision is also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: Upon being powered on for the first time the consumer IoT device enters a default network configuration that allows the user to start the installation process as instructed in the user manual. The installation process is linear, presents default options for each available service but gives the user the option to decide case by case or collectively whether they want the service activated or not. Ideally secure configuration of the device and each service is implemented automatically, or the user is presented with reasonable defaults.

EXAMPLE 2: For a consumer IoT device where network connectivity only enhances its intended use, when the device is installed and powered on for the first time, the device can immediately provide services that do not require network connectivity, and the mechanical installation steps (for example plugging in audio cables) are explained in the user manual. The device activates and deactivates network connectivity at the press of a button on the device casing. The steps to follow for further configuration of the device and of additional services are explained in the user manual.

EXAMPLE 3: A device comes with a manufacturer hub that automatically checks for updates and provisions these updates to its connected devices. The manufacturer hub handles revocation, version checking and other security best practices, according to the recommendations issued by a governmental agency for security or by prominent industry crypto-catalogues, adapted to the usage context of the device.

6.60 Provision 5.12-2

"The manufacturer should provide users with guidance on how to securely set up their device".
(ETSI EN 303 645 [i.1])

EXAMPLE 1: The user manual includes a clause covering the device configuration parameters and aspects of the physical setup that are relevant for the security of the device, along with a discussion and recommendations.

EXAMPLE 2: The manufacturer provides an online video tutorial and accessible manual, that contain guidance on how to setup a device and maintain it through its lifecycle. Specifically, this includes how to reset the device password.

EXAMPLE 3: The user is guided through the setup process by a wizard on the user interface and is supported on taking decisions to enable a secure setup.

6.61 Provision 5.12-3

"The manufacturer should provide users with guidance on how to check whether their device is securely set up".
(ETSI EN 303 645 [i.1])

EXAMPLE 1: The user interface of the device allows the user to start an automated configuration verification process. The process checks the version of installed firmware and software and checks the device configuration parameters against up-to-date recommendations by the manufacturer, communicated via TLS and/or DTLS. When these are within acceptable values the user interface informs the user that the device is securely set up. If not, the user is informed on how to secure their set up.

EXAMPLE 2: The interface provided by the device includes a device status that indicates if the device is in a secure state.

EXAMPLE 3: The manufacturer provides details of a source of help, such as a website. This allows a user to confirm they have performed installation correctly.

6.62 Provision 5.13-1

"The consumer IoT device software shall validate data input via user interfaces or transferred via Application Programming Interfaces (APIs) or between networks in services and devices". (ETSI EN 303 645 [i.1])

NOTE 1: Examples for this provision are also provided in ETSI EN 303 645 [i.1].

EXAMPLE 1: An API provided by a smart device, which receives data over a network, is clear, tightly scoped and validates requests and data. Invalid data or malformed requests are rejected by the device and an error logged in the device telemetry. API functions are limited to those needed for the smart device services, removing the risk of overexposing data or providing attackers access to core functionality. The APIs provided by the device are publicly documented and analysed by vulnerability research.

NOTE 2: Not documenting APIs and relying on security through obscurity is poor practice.

EXAMPLE 2: The input data of all APIs go through and are validated by a parser that has been automatically generated using a language grammar of bounded complexity. The grammar is adapted to the API and informs about allowed function calls, allowed data types and structures, allowed data values, and allowed cardinalities and ordering of the data items. It is also adapted to validating the mime type, encoding, and content of binary objects. The parser is generated with GNU Bison [i.6].

EXAMPLE 3: The input data of a POST request to a REST API is instantiated based on XML Schema Definition [i.7] or, when that is not practical, in JSON objects that can be validated against a JSON Schema [i.8].

EXAMPLE 4: The input data of a SOAP interface is validated by different validation rules, each consisting of one or more regular expression that are fit for the complexity of the input data. For every input type, the regular expression defines which values the succeeding parser of the application layer can interpret flawlessly. The regular expressions are predefined using Python's regular expression engine.

EXAMPLE 5: After the firmware has been input, i.e. uploaded via user interface, the device verifies its signature. The manufacturer cryptographically signs the firmware update for the constrained device. Therefore, the device implicitly has validated the incoming firmware as trusted.

7 Examples to meet data protection provisions for consumer IoT

7.1 Provision 6-1

"The manufacturer shall provide consumers with clear and transparent information about what personal data is processed, how it is being used, by whom, and for what purposes, for each device and service. This also applies to third parties that can be involved, including advertisers". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The FAQ on the manufacturer's website provides a precise list of the processed personal data, how it is being used, by whom and for what purposes.

EXAMPLE 2: For consumer IoT devices providing services falling under the scope of the GDPR [i.33], the manufacturer provides consumers with a data protection policy in accordance with the principles laid out in [i.9].

EXAMPLE 3: Whenever sensitive data is leaving the device, the user is informed about its content, destination and protection, and the user confirms that they wish to continue before the action is completed.

EXAMPLE 4: All sensitive personal data that leaves the device, the encryption used to protect it, and the use case of its further processing is logged in a file that the user can access through an interface on the device.

7.2 Provision 6-2

"Where personal data is processed on the basis of consumers' consent, this consent shall be obtained in a valid way". (ETSI EN 303 645 [i.1])

EXAMPLE 1: For consumer IoT devices providing services falling under the scope of the GDPR [i.33], the process by which the consumer's consent is obtained follows the principles laid out in [i.10].

EXAMPLE 2: The user explicitly confirms the processing of personal data via the device's user interface upon entering the data by enabling a check box linked to the manufacturer's general terms and conditions.

7.3 Provision 6-3

"Consumers who gave consent for the processing of their personal data shall have the capability to withdraw it at any time". (ETSI EN 303 645 [i.1]).

EXAMPLE: For each service provided by the consumer IoT device and for which the user gave consent to the processing of personal data, the user interface provides commands to withdraw consent given for each of the data processing activities related to the service.

7.4 Provision 6-4

"If telemetry data is collected from consumer IoT devices and services, the processing of personal data should be kept to the minimum necessary for the intended functionality". (ETSI EN 303 645 [i.1])

EXAMPLE 1: When the collection of telemetry data is intended to allow the manufacturer to act on problems on a per-device basis, the telemetry data is linked to a device pseudonym that does not allow any conclusions to be drawn about individuals.

EXAMPLE 2: When the collection of telemetry data is intended to allow the manufacturer to act on problems on a per-device basis, the telemetry data is anonymized.

7.5 Provision 6-5

"If telemetry data is collected from consumer IoT devices and services, consumers shall be provided with information on what telemetry data is collected, how it is being used, by whom, and for what purposes". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The FAQ on the manufacturer's website provides a precise list of the collected telemetry data, how it is being used, by whom and for what purposes.

EXAMPLE 2: For consumer IoT devices providing services falling under the scope of the GDPR [i.33], the data protection policy is provided in accordance with the principles laid out in [i.9] and includes a clause on telemetry data. Said clause covers both non-personal data and personal data.

8 Handling of recommendations

8.1 Status of recommendations in ETSI EN 303 645

Being able to implement a recommendation depends on many factors including the hardware, software, and communication technology available to the manufacturer to implement a given IoT product. Since the available technology can change over time, these examples are meant as illustration only, not as firm criteria. Also, not supporting a given recommendation has an impact on the security provided by the device. A risk analysis from the manufacturer or a third party could conclude that a given use case should not be implemented in case some recommendations are not supported.

8.2 Example situations where recommendations cannot be followed

8.2.1 Provision 5.2-2

"Disclosed vulnerabilities should be acted on in a timely manner". (ETSI EN 303 645 [i.1])

EXAMPLE: A security update for the microcode of a widely used CPU model requires extended validation to ensure that deploying the update will not cause more damage than the vulnerability it fixes.

8.2.2 Provision 5.2-3

"Manufacturers should continually monitor for, identify and rectify security vulnerabilities within products and services they sell, produce, have produced and services they operate during the defined support period".
(ETSI EN 303 645 [i.1])

EXAMPLE: A device component is a closed system and the device manufacturer relies on the component supplier to perform these activities for said component.

8.2.3 Provision 5.3-1

"All software components in consumer IoT devices should be securely updateable". (ETSI EN 303 645 [i.1])

EXAMPLE: The software of a battery charge controller is not meant to be modified once it has been vetted for safety.

8.2.4 Provision 5.3-4

"Automatic mechanisms should be used for software updates". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The user stays in control of when a device that serves as authentication factor is updated, such that other activities are not impaired by the update.

EXAMPLE 2: The device claims to be compliant to the Directive (EU) 2019/770 [i.30] and follows its recommendation (see article 47 in [i.30]) to let the user decide whether to install a software update or not.

8.2.5 Provision 5.3-5

"The device should check after initialization, and then periodically, whether security updates are available".
(ETSI EN 303 645 [i.1])

EXAMPLE: The device is connected to a notification framework over which the availability of a security update is announced.

8.2.6 Provision 5.3-6

"If the device supports automatic updates and/or update notifications, these should be enabled in the initialized state and configurable so that the user can enable, disable, or postpone installation of security updates and/or update notifications". (ETSI EN 303 645 [i.1])

EXAMPLE: The device connects through a metered network. Automated notifications and downloads are disabled by default to prevent unexpected charges.

8.2.7 Provision 5.3-9

"The device should verify the authenticity and integrity of software updates". (ETSI EN 303 645 [i.1])

EXAMPLE: A smart hub in a smart home network handles this task on behalf of sensors with which it has security associations.

8.2.8 Provision 5.3-11

"The manufacturer should inform the user in a recognizable and apparent manner that a security update is required together with information on the risks mitigated by that update". (ETSI EN 303 645 [i.1])

EXAMPLE: The deployment model is such that there is no direct link between the manufacturer and the user. A third party is responsible for informing the user.

8.2.9 Provision 5.3-12

"The device should notify the user when the application of a software update will disrupt the basic functioning of the device". (ETSI EN 303 645 [i.1])

EXAMPLE: This notification is provided within an external application through which the device is managed.

8.2.10 Provision 5.3-14

"For constrained devices that cannot have their software updated, the rationale for the absence of software updates, the period and method of hardware replacement support and a defined support period should be published by the manufacturer in an accessible way that is clear and transparent to the user". (ETSI EN 303 645 [i.1])

EXAMPLE: The device is designed for one-time use and therefore the lifecycle ends after the usage. Moreover, hardware replacement is not applicable.

NOTE: This example does not constitute an endorsement of consumer IoT devices being used as consumable goods.

8.2.11 Provision 5.3-15

"For constrained devices that cannot have their software updated, the product should be isolable and the hardware replaceable". (ETSI EN 303 645 [i.1])

EXAMPLE 1: An outdoor temperature sensor connected through sub-GHz communication technologies does not benefit from software update and is part of a WAN.

EXAMPLE 2: The device is a small button, which can send yes/no commands to a base station. The button has no power supply, but uses energy harvesting to send a message. Because a connection from outside is not possible, the device offers only a very small attack surface and thus no risk is expected. Therefore, the device does not have to be isolable and the hardware replaceable.

8.2.12 Provision 5.5-2

"The consumer IoT device should use reviewed or evaluated implementations to deliver network and security functionalities, particularly in the field of cryptography". (ETSI EN 303 645 [i.1])

EXAMPLE: The implementation is the first of its kind, addressing a new functional domain, and was thus only subject to first-party review.

8.2.13 Provision 5.5-3

"Cryptographic algorithms and primitives should be updateable". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The device is a low-cost device for which cryptographic capabilities are provided via hardware acceleration, and the device lifetime is within the bounds of the expected lifetime of the cryptographic algorithm(s).

EXAMPLE 2: The device is designed for one-time use and therefore the lifecycle ends after the usage.

NOTE: Example 2 does not constitute an endorsement of consumer IoT devices being used as consumable goods.

8.2.14 Provision 5.5-4

"Access to device functionality via a network interface in the initialized state should only be possible after authentication on that interface". (ETSI EN 303 645 [i.1])

EXAMPLE: The intended use of the device is to provide publicly available information, such as outdoor temperature in a public place.

8.2.15 Provision 5.5-6

"Critical security parameters should be encrypted in transit, with such encryption appropriate to the properties of the technology, risk and usage". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The device is meant to interoperate with another device over a legacy, unprotected protocol, that is the only option for said other device.

EXAMPLE 2: A time-based one-time-password (TOTP, IETF RFC 6238 [i.29]) is used as a second factor for authentication. The disclosure of the TOTP itself does not compromise the device.

8.2.16 Provision 5.6-3

"Device hardware should not unnecessarily expose physical interfaces to attack". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The device provides a hardware bus for extension modules.

EXAMPEL 2: The device is built upon a multi-purpose hardware platform, with several exposed physical interfaces suitable for different kinds of devices. However, the device does not use one of the exposed physical interfaces.

8.2.17 Provision 5.6-5

"The manufacturer should only enable software services that are used or required for the intended use or operation of the device". (ETSI EN 303 645 [i.1])

EXAMPLE: The device embeds security monitoring software.

8.2.18 Provision 5.6-6

"Code should be minimized to the functionality necessary for the service/device to operate". (ETSI EN 303 645 [i.1])

EXAMPLE: The device functionality is built on top of a generic platform that targets various use cases, and the core binaries cannot be tailored further.

8.2.19 Provision 5.6-7

"Software should run with least necessary privileges, taking account of both security and functionality". (ETSI EN 303 645 [i.1])

EXAMPLE: The device is a low-cost constrained device with a single, small memory address space for all application code.

8.2.20 Provision 5.6-8

"The device should include a hardware-level access control mechanism for memory". (ETSI EN 303 645 [i.1])

EXAMPLE: The device is a low-cost constrained device with a single memory address space and no memory management unit.

8.2.21 Provision 5.6-9

"The manufacturer should follow secure development processes for software deployed on the device". (ETSI EN 303 645 [i.1])

EXAMPLE: The manufacturer is rebranding the device, which is obtained from a supplier under white label.

8.2.22 Provision 5.7-1

"The consumer IoT device should verify its software using secure boot mechanisms". (ETSI EN 303 645 [i.1])

EXAMPLE: The device is a constrained, short lifespan device, for which the software is burnt in read-only memory.

8.2.23 Provision 5.7-2

"If an unauthorized change is detected to the software, the device should alert the user and/or administrator to the issue and should not connect to wider networks than those necessary to perform the alerting function". (ETSI EN 303 645 [i.1])

EXAMPLE 1: The device is not administered by the user but by the device manufacturer, the alert is therefore sent to the manufacturer.

EXAMPLE 2: An alert is not sent to the user, as the manufacturer's risk assessment shows the benefit from this alert is not offset by the extra attack surface such an alert creates.

8.2.24 Provision 5.8-1

"The confidentiality of personal data transiting between a device and a service, especially associated services, should be protected, with best practice cryptography". (ETSI EN 303 645 [i.1])

EXAMPLE 1: When no other option is available for hostname resolution, the device performs plain DNS [i.11] requests.

EXAMPLE 2: When the device communicates with third-party services, where security capabilities and configurations are not in control by the manufacturer.

8.2.25 Provision 5.9-1

"Resilience should be built in to consumer IoT devices and services, taking into account the possibility of outages of data networks and power". (ETSI EN 303 645 [i.1])

EXAMPLE: The device is a temperature sensor that only provides on-demand temperature readings; resilience is not an important aspect for this device.

8.2.26 Provision 5.9-2

"Consumer IoT devices should remain operating and locally functional in the case of a loss of network access and should recover cleanly in the case of restoration of a loss of power". (ETSI EN 303 645 [i.1])

EXAMPLE: An internet radio that solely plays audio streams from the internet cannot operate when local network access is lost.

8.2.27 Provision 5.9-3

"The consumer IoT device should connect to networks in an expected, operational and stable state and in an orderly fashion, taking the capability of the infrastructure into consideration". (ETSI EN 303 645 [i.1])

EXAMPLE: The device is a security camera that resumes streaming immediately after network access is restored.

8.2.28 Provision 5.10-1

"If telemetry data is collected from consumer IoT devices and services, such as usage and measurement data, it should be examined for security anomalies". (ETSI EN 303 645 [i.1])

EXAMPLE: The collected data do not allow detection of security anomalies.

8.2.29 Provision 5.11-2

"The consumer should be provided with functionality on the device such that personal data can be removed from associated services in a simple manner". (ETSI EN 303 645 [i.1])

EXAMPLE: The device is a body thermometer. The device control interface is connected to a smartphone application that provides the functionality to remove personal data from associated services.

8.2.30 Provision 5.11-3

"Users should be given clear instructions on how to delete their personal data". (ETSI EN 303 645 [i.1])

EXAMPLE: The device is a room temperature sensor that does not process personal data.

8.2.31 Provision 5.11-4

"Users should be provided with clear confirmation that personal data has been deleted from services, devices and applications". (ETSI EN 303 645 [i.1])

EXAMPLE: The device is a room temperature sensor that does not process personal data.

8.2.32 Provision 5.12-1

"Installation and maintenance of consumer IoT should involve minimal decisions by the user and should follow security best practice on usability". (ETSI EN 303 645 [i.1])

EXAMPLE: The device offers an additional "expert mode" for installation and configuration.

8.2.33 Provision 5.12-2

"The manufacturer should provide users with guidance on how to securely set up their device".
(ETSI EN 303 645 [i.1])

EXAMPLE: All device and service configuration parameters are automatically set or remotely steered to secure values, and there is no requirement from the device intended usage for physical security measures.

8.2.34 Provision 5.12-3

"The manufacturer should provide users with guidance on how to check whether their device is securely set up". (ETSI EN 303 645 [i.1])

EXAMPLE: All device and service configuration parameters are automatically set or remotely steered to secure values.

8.2.35 Provision 6-4

"If telemetry data is collected from consumer IoT devices and services, the processing of personal data should be kept to the minimum necessary for the intended functionality". (ETSI EN 303 645 [i.1])

EXAMPLE: The processing of personal data is also done in accordance with other legal obligations of the manufacturer.

History

Document history		
V1.1.1	March 2022	Publication
V1.2.1	September 2022	Publication