Consumer Technology Association™

ANSI/CTA Standard

Baseline Cybersecurity Standard for Devices and Device Systems

ANSI/CTA-2088-A

Approved American National Standard ANSI

May 2022

NOTICE

Consumer Technology Association (CTA)™ Standards, Bulletins and other technical publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for his particular need. Existence of such Standards, Bulletins and other technical publications shall not in any respect preclude any member or nonmember of the Consumer Technology Association from manufacturing or selling products not conforming to such Standards, Bulletins or other technical publications, nor shall the existence of such Standards, Bulletins and other technical publications preclude their voluntary use by those other than Consumer Technology Association members, whether the standard is to be used either domestically or internationally.

Standards, Bulletins and other technical publications are adopted by the Consumer Technology Association in accordance with the American National Standards Institute (ANSI) patent policy. By such action, the Consumer Technology Association does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the Standard, Bulletin or other technical publication.

This document does not purport to address all safety problems associated with its use or all applicable regulatory requirements. It is the responsibility of the user of this document to establish appropriate safety and health practices and to determine the applicability of regulatory limitations before its use.

(Formulated under the cognizance of the CTA **R14 Cybersecurity and Privacy Management Committee**.)

Published by
©CONSUMER TECHNOLOGY ASSOCIATION 2022
Technology & Standards Department
www.cta.tech

FOREWORD

This standard was developed by the Consumer Technology Association under the auspices of the R14 Cybersecurity and Privacy Management Committee.

(This page intentionally left blank.)

TABLE OF CONTENTS

Baseline Cybersecurity Standard for Devices and Device Systems

1 SCOPE

This standard specifies baseline Device Security capabilities and related organizational Security capabilities and recommendations for Devices and Device systems, including for individual connected Devices, Endpoint Devices, components, hardware modules, chips, software, sensors or other operating components.

2 DEFINITIONS AND ABBREVIATIONS

For the purposes of this document, the following definitions apply.

Authenticating	see Authentication.
Authentication	The process of verifying the Identity of an Entity.
Authenticator	Something the Claimant possesses and controls (typically a cryptographic module or password) that is used to authenticate the Claimant's Identity. See Authenticator in NIST SP 800-63-3 [14].
Authorization	The process of verifying that a requested action or service is approved for a specific Entity. See Authorization in NIST SP 800-152 [27].
Authorized User	An Entity permitted to access or manipulate the product.
Authorizing	see Authorization.
Claimant	A party whose Identity is to be verified using an Authentication protocol. See Claimant in NIST SP 800-63-3 [14].
Commonly Available Tools	Widely-available, inexpensive tools (such as common screwdrivers) and free or inexpensive software.
Configuration	The specific hardware and software details, capacity or capability, and exactly what the system consists of, not including normal user-accessible Device functions or controls.
Console Port	A local (i.e., not remote) physical, dedicated interface (whether wired or wireless) that provides capabilities of Remote Management, Remote Monitoring or both. Note that a virtualized physical interface made available over a network (e.g., via IPMI) is also a Console Port.
Credential	An object or data structure that authoritatively binds an Identity — via an identifier or identifiers — and (optionally) additional attributes to at least one Authenticator possessed and controlled by a Claimant. See Credential in NIST SP 800-63-3 [14].
CWE	Common Weakness Enumeration. CWE™ is a community-developed list of common software Security weaknesses. It

	serves as a common language, a measuring stick for software Security tools, and a baseline for weakness identification, mitigation and prevention efforts.
Deprovision	To force the Device back to a state that is as secure as the original as-manufactured state and with data deletion that protects security information such as Device Credentials and personally identifiable information (PII).
Device	A finished product, available to end-users in the last point in the Pre-Market activity. A Device is usable for its intended functions without being embedded or integrated into any other product and is not a component.
Device Credential	A Credential where the Claimant is a Device.
Diagnostic Port	A local, dedicated interface used for direct access to hardware ports, resources, configurations, etc., that provides capabilities that are not ordinarily provided to anyone except the manufacturer. For example, JTAG, I^2C, and SPI.
Endpoint	An Entity comprised of one or more components, addressable on a network.
Entity	An item with a recognizably distinct existence.[1]
EoL	End of Life (of an IoT Device).
EoS	End of Service (of an IoT Device).
Field Deprovisioning	Deprovisioning performed Post-Market and without physical return of the Device to the manufacturer, such as by a user in possession of the Device or via remote means. See Deprovision.
Identity	An inherent property of an Entity that distinguishes it from all other entities. An Identity needs to exist in a namespace to allow it to be referred to without ambiguity.[2]
Insecure	see Security.
Integrity	(Also referred to as "Data Integrity") A property whereby data has not been altered in an unauthorized manner since it was created, transmitted, or stored. See [9].
Manufacturer	The Entity responsible for the creation of a Device.
Port	A physical or logical interface which provides access to the device. In this document, Port does not refer to TCP or UDP "ports". Those will be referred to as TCP Ports or UDP Ports.

[1] See [30].

[2] ISO/IEC/IEEE 31320-2:2012, *Information technology -- Modeling Languages -- Part 2: Syntax and Semantics for IDEF1X97 (IDEFobject),* September 2012, https://www.iso.org/standard/60614.html.

Post-Market	After release of the individual Device to the field (i.e., after it leaves the factory and goes into the distribution channel). Compare to Pre-Market.
Pre-Market	Prior to release of the individual Device to the market (e.g., before it leaves the factory and goes into the distribution channel). Compare to Post-Market.
Proprietary Tools	Tools that are not Commonly Available Tools or Specialized Tools due to restricted sales or use provisions, such as a special-purpose tool sold by a Device Manufacturer only to authorized repair facilities.
Remote Management	The facility or capability of configuring a Device without physical access to the Device.
Remote Monitoring	The facility or capability of monitoring a Device without physical access to the Device. Compare to Remote Management.
Reprovision	A method for Authorized users to securely reconfigure and redeploy a Device Post-Market (especially to return the product to factory defaults or an authorized restore point) and securely remove data collected by the Device (that is not essential to its operation) within a defined period established by the organization.
Secure	see Security.
Security	Protecting information and information systems from unauthorized access, use, disclosure, disruption, modification or destruction in order to provide — (A) integrity, which means guarding against improper information modification or destruction, and includes ensuring information non-repudiation and authenticity; (B) confidentiality, which means preserving authorized restrictions on access and disclosure, including means for protecting personal privacy and proprietary information; and (C) availability, which means ensuring timely and reliable access to and use of information.[3]
Security Support Phase	The period of time when a specific product is receiving security updates from the Manufacturer. When a product is no longer in the Security Support Phase, it no longer receives security updates.
Sensitive Data	Data that, if extracted or observed by a third party, would compromise system security (e.g., Credentials) or user privacy (e.g., personally identifiable information).

[3] 44 USC §3552(b)(3), available at https://uscode.house.gov/view.xhtml?req=granuleid:USC-prelim-title44-section3552&num=0&edition=prelim.

Specialized Tools	Commercially available tools that are not Commonly Available Tools, including logic analyzers, oscilloscopes, debuggers, decompilers, tools for tamper-resistant fasteners and similar tools.
User Authentication	see Authorized User.
User Credential	A Credential where the Claimant is an end-user.
Vulnerability	A software weakness found in the product for which an exploit can exist such that it can be directly used by an attacker.

3 REFERENCES

3.1 Normative References

The following documents contain provisions that, through reference in this text, constitute normative provisions of this standard. At the time of publication, the editions indicated were valid. All documents are subject to revision. Users of this document are cautioned that newer editions of the referenced documents might or might not be compatible.

3.1.1 Normative Reference List

1. National Institute of Standards and Technology (NIST) Special Publication 800-131A Rev. 2, *Transitioning the Use of Cryptographic Algorithms and Key Lengths*, https://doi.org/10.6028/NIST.SP.800-131Ar2.
2. Internet Engineering Task Force (IETF) RFC 5905, *Network Time Protocol Version 4: Protocol and Algorithms Specification*, June 2010, https://tools.ietf.org/html/rfc5905.
3. International Electrical and Electronics Engineers (IEEE) 1588-2008, *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, March 2008, https://standards.ieee.org/standard/1588-2008.html.
4. IETF RFC 8915, *Network Time Security for the Network Time Protocol*, https://tools.ietf.org/html/rfc8915.
5. IETF RFC 6066, *Transport Layer Security (TLS) Extensions: Extension Definitions*, https://tools.ietf.org/html/rfc6066.
6. NIST SP 800-52 Rev. 2, *Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations*, https://doi.org/10.6028/NIST.SP.800-52r2.
7. Open Web Application Security Project (OWASP), *OWASP Top 10 – 2017: The Ten Most Critical Web Application Security Risks*, https://owasp.org/www-project-top-ten/OWASP_Top_Ten_2017/.
8. ICSA Labs, *Firewall Certification Criteria Baseline Module – Version 4.2*, http://www.icsalabs.com/sites/default/files/FW_Baseline_4.2_0.pdf.
9. NIST SP 800-57 Part 1 Revision 5*, Recommendation for Key Management Part 1: General*, https://doi.org/10.6028/NIST.SP.800-57pt1r5.
10. NIST SP 800-90A Revision 1, *Recommendation for Random Number Generation Using Deterministic Random Bit Generators*, http://dx.doi.org/10.6028/NIST.SP.800-90Ar1.

11. NIST FIPS PUB 140-3, *Security Requirements for Cryptographic Modules*, https://doi.org/10.6028/NIST.FIPS.140-3.
12. NIST SP 800-133 Revision 2, *Recommendation for Cryptographic Key Generation*, https://doi.org/10.6028/NIST.SP.800-133r2.
13. NIST SP 800-88 Revision 1, *Guidelines for Media Sanitization*, http://dx.doi.org/10.6028/NIST.SP.800-88r1.

3.2 Informative References

The following documents contain provisions that, through reference in this text, constitute informative provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision. Users of this standard are cautioned that newer editions of the referenced documents might or might not be compatible.

3.2.1 Informative Reference List

14. NIST SP 800-63-3, *Digital Identity Guidelines*, June 2017, https://doi.org/10.6028/NIST.SP.800-63-3.
15. NIST SP 800-37 Rev. 2, *Risk Management Framework for Information Systems and Organizations*, December 2018, https://doi.org/10.6028/NIST.SP.800-37r2.
16. Council to Secure the Digital Economy (CSDE), *The C2 Consensus on IoT Device Security Baseline Capabilities*, https://securingdigitaleconomy.org/projects/c2-consensus/.
17. NIST, *FIPS PUB 186-4, Digital Signature Standard (DSS),* https://doi.org/10.6028/NIST.FIPS.186-4.
18. IETF RFC 8446, *The Transport Layer Security (TLS) Protocol Version 1.3*, August 2018, https://datatracker.ietf.org/doc/html/rfc8446.
19. Internet Assigned Numbers Authority, *Transport Layer Security (TLS) Extensions*, https://www.iana.org/assignments/tls-extensiontype-values/tls-extensiontype-values.xhtml.
20. IETF RFC 6960, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP*, June 2013, https://datatracker.ietf.org/doc/html/rfc6960.
21. IETF RFC 6749, *The OAuth 2.0 Authorization Framework*, October 2012, https://datatracker.ietf.org/doc/html/rfc6749.
22. IETF RFC 6347, *Datagram Transport Layer Security Version 1.2*, January 2012, https://datatracker.ietf.org/doc/html/rfc6347.
23. IETF RFC 4347, *Datagram Transport Layer Security*, April 2006, https://datatracker.ietf.org/doc/html/rfc4347.
24. MITRE, *CWE List Version 4.6, Common Weakness Enumeration*. https://cwe.mitre.org/data/downloads.html.
25. NIST, "Lightweight Cryptography." *Information Technology Laboratory – Computer Security Resource Center*. https://csrc.nist.gov/projects/lightweight-cryptography.
26. ISO/IEC 29192-2:2019, *Information security — Lightweight cryptography — Part 2: Block ciphers*, https://www.iso.org/standard/78477.html.

27. NIST SP 800-152, *A Profile for U.S. Federal Cryptographic Key Management Systems*, October 2015, http://dx.doi.org/10.6028/NIST.SP.800-152.
28. IETF RFC 8995, *Bootstrapping Remote Secure Key Infrastructure (BRSKI)*, May 2021, https://datatracker.ietf.org/doc/html/rfc8995.
29. IETF RFC 8366, *A Voucher Artifact for Bootstrapping Protocols*, May 2018, https://datatracker.ietf.org/doc/html/rfc8366.
30. ISO/IEC 24760-1:2019, *Information technology – Security techniques – A framework for identity management -- Part 1: Terminology and concepts*, May 2019, https://www.iso.org/standard/77582.html.
31. ISO/IEC 29115:2013, *Information technology — Security techniques — Entity authentication assurance framework*, https://www.iso.org/standard/45138.html.
32. IEC 62443 series, *Industrial communication networks – Network and system security*, https://www.iec.ch/homepage.

4 COMPLIANCE NOTATION

CTA defines the following compliance terms:

shall	This word indicates specific provisions that are to be followed strictly (no deviation is permitted).
shall not	This phrase indicates specific provisions that are absolutely prohibited.
should	This word indicates that a certain course of action is preferred but not required.
should not	This phrase means a certain possibility or course of action is undesirable but not prohibited.
may	This phrase indicates that a certain course of action is optional, and this document does not express a recommendation as to preference.
need not	This phrase indicates that a certain course of action is not required (i.e., optional), and this document does not express a recommendation as to preference.

5 SECURE DEVICE CAPABILITIES – BASELINE

This section includes requirements for Device capabilities that are properties of the hardware and software, as opposed to business or development processes or capabilities.

As part of the Device's design specification process, it is critical to define the Security objectives within the context of the Device's operational environment. Security objectives are best assessed through a risk management process. In particular, the assessment and objectives will be expected to consider the impact the Devices could potentially have on the broader infrastructure in which they operate – for example, how the Devices might be put to use in a botnet attack and how such malicious use would be mitigated or prevented. [16]

The requirements that follow can be identified by a *field code* and a *title*. The field code consists of a two- or three-character category and a number, separated by a hyphen, e.g., **CAT-001** (i.e., the first requirement in the group "Category"). The title is a short descriptive name in square brackets – e.g., **[Sample Title]**.

The field code numbering scheme generally starts at 001 and increments by one for each new requirement. Exceptions may occur as the document is revised. For example, in development of this revised standard a new requirement DAR-000 was inserted ahead of the existing DAR-001, and the existing DAR-008 was dropped but marked as "Removed" to preserve the remaining numbering of requirements in that section. Where possible, this revision retains the ANSI/CTA-2088 (original) numbering of requirements.

5.1 Device Identifiers

> *"A unique value associated with the endpoint (or values associated with the functional entities within the endpoint) that exists in a namespace to allow it to be referenced without ambiguity. This value is distinct and distinguishes a device from all other devices."* See *C2 Consensus* [16] 5.1.1, definition of *Device Identifiers.*

The Device Identifier capability is for the Device or its Endpoints to provide a unique and – where possible – attestable and protected Identity parameter to a query from an authorized requestor.

Credentials are used to verify the Identity of the Endpoint. There are several levels of trust that can apply to an Endpoint, depending on the threat model of the particular Internet of Things (IoT) system. Each level of trust determines the minimum Security capabilities of the Credentials, including Credential uniqueness, Credential storage and Credential usage (e.g., for Authentication, Authorization, etc.). Digital certificates, public keys, passwords and biometrics are all examples of Credentials, but vary in their level of trust.

Credentials generated at time of manufacture are referred to as manufacturer credentials and should be used as Bootstrapping Credentials. Operational credentials are those established within an operational environment, after the time of manufacture.

This capability does not imply a requirement for a Public Key Infrastructure (PKI). While this capability is important in all Devices, how it is achieved will vary with Device complexity. Network asset identification is important for all Devices. For purposes of bootstrapping, what is required is either a secret, a public/private key pair, or a mechanism by which an ownership claim can be made.

This capability is helpful in all Devices, but particularly important now in Devices that do not have input functions or displays. It will be essential in the future for managing the tens of billions of IoT Devices predicted to be deployed, especially within managed network environments.

Network asset Identity is a building block that enables a broad range of Security controls that depend on proper handling of Identity. For example, Identity is the basis for trust in asset management, Authentication, Authorization and remote maintenance.

An Endpoint may have a single Identity, or multiple Identities, used for different applications.

One common example of a Credential is a cryptographic certificate (e.g., X.509 digital certificate). When a certificate is used as a Bootstrapping Credential, a transfer of ownership can occur. For example, the device may establish a trust anchor and operational Credential in a local deployment, with the ownership of the Credential in a key management server. Two examples of such bootstrapping are voucher-based systems [29][28] and FIDO Device Onboarding (FDO). Another example is the out-of-band (OOB) keying information used in Bluetooth Low Energy (BLE), where the operational credential is the derived Long Term Key (LTK). In this case, no certificates are used, and a Diffie-Hellman exchange occurs with information shared out of band.

The level of trust attributed to a Credential depends on its uniqueness and strength. An IP address or MAC address may be unique. However, they are very weak Credentials, as they can be falsified to impersonate another Endpoint. A cryptographic certificate is both unique (with appropriate randomness) and strong (depending on key type and length). However, if the private key associated with the certificate is not stored and processed in protected storage and memory, the certificate can still be compromised. Several standards exist that provide guidance on choosing the right level of protection for Endpoint Identity, including ISO/IEC 29115 [31], IEC 62443 [32], and ISO/IEC 24760-1 [30].

Potentially, the network asset identification value can be globally unique. Note that immutable Device identifiers can be problematic from a privacy perspective and are expected to be evaluated on that basis.

Stable Device identifiers, such as IMEI, are privacy-sensitive. Care should be taken to minimize the privacy risk of exporting or sharing such identifiers, especially to third parties and for reasons beyond the core necessary operation of the overall system or service. Developers are expected to consider risk mitigation strategies such as automatic or customer-driven identifier reset (MAC randomization is an example of this) and limiting access only to privileged system software that needs such access.

Note that in low-complexity Devices, for design purposes, there might not be an outward facing (printed and visible) serial number.

DI-001 [*Operational Credentials*] If a public/private key pair is used, these Credentials shall have a strength at least equivalent to Elliptic Curve Cryptography (ECC) P-256 as per NIST SP 800-57 [9], and as designated as acceptable by NIST SP 800-131A Rev. 2 [1], and should be an X.509 certificate based on the Elliptical Curve Digital Signature Algorithm (ECDSA), with ECC and ECDSA as specified in FIPS 186-4 [17].

DI-002 [*Roaming Privacy*] Devices shall employ approaches that do not require disclosure of Credentials to prevent operational Credentials from being exposed to unauthorized networks.

DI-003 [*Device Credentials*] Any Device without a display and input mechanism shall have a Device Credential. This Credential shall take the form of a public key, a certificate, or a private shared key.

The purpose of this requirement is to establish a unique Identity for the Device so that it can be connected to the correct environment automatically without the need for input from the owner of the Device. Section 5.2 specifies the manner in which these credentials are to be stored.

DI-004 [*Bootstrapping Mechanism*] Any Device without a display and input mechanism should employ at least one industry consensus standard bootstrapping mechanism, using the Device Credential, in order to establish an operational Credential.

Note that Device onboarding is also the subject of Section 8.2. The requirements in this section do not entirely replace that future consideration.

The purpose of DI-004 is to make it possible to securely onboard a Device into a local network environment so that it can be locally identified as being Authorized in a given environment. Standard bootstrapping mechanisms include the WiFi Alliance's Device Provisioning Protocol (EasyConnect), the Bluetooth SIG's Out of Band Pairing (OOBP) public key mechanism, or an IEEE 802.1ar certificate combined with RFCs 8366 or 8995 (BRSKI). Each of these standards has either open-source code or chipset-supported code available. Some of these mechanisms make use of the infrastructure and are alternatives to other bootstrapping approaches, such as interactive pairing that occurs in BLE. Some of these mechanisms (such as Device Provisioning Protocol) do not require additional infrastructure other than an app on a smart phone or tablet.

The infrastructure requirements for different onboarding mechanisms vary. The following table is meant to illustrate this point:

Mechanism	**Manufacture Time**	**Manufacturer Operations**	**Local operations (outside the IoT device hardware)**
Device Provisioning Protocol (DPP)	Install public/private key pair, make public key accessible (packaging, eBOM, etc.).	None.	A smartphone app may be used to install credentials. This app may be linked further to local infrastructure.
BRSKI [28]	Install certificate/private key (see IEEE 802.1ar below).	Operate Validation authority (MASA).	A BRSKI-compatible Join Proxy is required inside the domain; this may be a BRSKI-compatible router or other device.[4] Real-time internet connectivity to the MASA server is also required during the onboarding process.

[4] See [28] section 1.5, Requirements for Autonomic Networking Infrastructure (ANI) Devices.

Mechanism	Manufacture Time	Manufacturer Operations	Local operations (outside the IoT device hardware)
FIDO Device Onboarding	Install device certificate/private key, install FDO manufacturer credentials and create initial FDO Ownership Voucher outside of the device.	Choose appropriate certificate authority for device certificate; manage ownership vouchers and route them to supply chain entities.	Rendezvous server. This may be supplied by the manufacturer as an app for a closed network; it may be operated on the public internet by the manufacturer to service its devices; or it may be operated by a business such as an industry coalition. Currently available for download and compiling from lf-edge/secure-device-onboard.
IEEE 802.1ar	Incorporate a non-fungible globally-unique Initial Device Identifier (IDevID).	Operate a provisioning service.	An IoT hub, router or other device allows access based on the IDevID or on a locally created Locally Significant Device Identifier (LDevID) assigned to device owner.

5.2 Secured Access

> *"Protection of device operational and management capabilities (including the associated software and configuration) by requiring user[5] authentication to read or modify the configuration, including means to ensure device-unique credentials for administrative access, and by protecting access to interfaces."* See *C2 Consensus* [16] 5.1.2, definition of *Secured Access.*

This capability includes Authenticating and Authorizing users and other Devices or services for remote or local access to the Configuration and stored data. Authentication takes different forms and will depend on the application but may include requiring a Secure certificate from a trusted source, User Credentials, biometrics and multi-factor Authentication. Authentication should follow good cryptographic practices, including requiring complex passwords and period changing of some passwords, certificates or keys.

This capability also includes physical interfaces (e.g., debug ports or JTAG) as needed to ensure protection of the Configuration. This capability does not include preventing or detecting physical access to the Device.

This capability is intended to protect the Device from unauthorized access to the internals of the Device either remotely or when the malicious actor has physical access.

SA-001 [*Device Configuration*] The Device should allow changes to its Configuration.

The Device Configuration capability does not define which configuration settings should exist, simply that a mechanism to manage configuration settings exists.

[5] "User" refers to the consumer using a Device, a technician responsible for installation or maintenance, an authorized employee in a managed environment, etc.

SA-002 [*Authentication for Admin Access*] Successful User Authentication shall be completed before any Configuration change or administrative function is permitted.

5.2.1 Credentials and Logins

Credentials are necessary to securing access to and from any Device. Credentials come in a variety of forms, including passwords, passphrases (such as for a Wi-Fi network), username and password combination, X.509 certificates (such as those used by website domains) and other certificate formats. For use with TLS [18] and DTLS [22], the Internet Assigned Numbers Authority (IANA) [19] maintains a TLS Certificate Types registry.

Some Credentials are presented and used by the Device or its applications to allow it to be authenticated by external sites, Devices or people. These are the Device's own Credentials. Other Credentials can be used by the Device to authenticate external sites, Devices, software or people. What activities are authorized after successful Authentication is a separate matter of policy.

Any Credentials that can be used to compromise the Security of the Device or system that includes the Device (whether these are the Device's own Credentials or Credentials used to authenticate others) are to be secured as described in Section 5.4. Requirements for generating and managing Credentials are below.

For cases where stored Credentials are used to authenticate others (e.g., login for remote access to the Device or determining whether a web or cloud server can be trusted), it is important to use appropriate measures to defend against Credential-based attacks. The mechanism by which the Device verifies those other Credentials needs to be resistant to tampering in order to ensure that Credential verification is completed correctly. In many cases the authenticity of a Credential is verified using a public key present on the Device. Those keys are subject to the key management requirements in this document.

Credential-based attacks include brute force attacks (repeated guessing of passwords), certificate Identity spoofing, and using compromised certificates and passwords.

For any Credentials or passwords that could compromise the Security of the Device or system including the Device if they were successfully used by an attacker, requirements SA-003 through SA-016 apply:

SA-003 [*Rate-Limit Login Attempts*] Unsuccessful login attempts on Ports other than diagnostic Ports shall be rate-limited to prevent brute-force attacks and similar and should include exponentially-increasing password or Credential entry delays after a reasonable number of sequential incorrect entry attempts.

See Section 5.7 for additional logging requirements.

SA-004 [*Diagnostic Ports*] Diagnostic Ports should be secured by disabling or by limiting features and access to the minimum necessary to accomplish – e.g., field service

functions. Such ports should, to the extent possible, require passwords or Credentials for access. Such Ports should, to the extent possible, rate-limit Credential failures to limit brute-force attacks. See SA-003.

SA-005 [*Change Non-Unique Default Credentials on First Use*] When a non-unique default Credential exists that could compromise the Security of the Device or system, this Credential shall be required to be changed upon first use.

This includes Credentials that allow users to access the Device through graphical user interfaces (e.g., supplied from an embedded HTTP server) or command-line interfaces (e.g., supplied over SSH).

SA-006 [*Using Unique Default Credentials*] Devices should have unique (per Device) default Credentials for Credentials that could compromise the Security of the Device or system.

As computing capabilities become more powerful and less expensive, recommended best practices might change.

SA-007 [*Ensuring Device Credentials Can Be Updated*] Device Credentials that could compromise the Security of the Device or system including the Device if they were successfully used by an attacker shall be stored in a manner that allows them to be updated.

Requirements related to cryptographically securing Credentials are in Section 5.4.

SA-008 [*Including Unique Device Credentials in Firmware*] If included in firmware, Device Credentials that could compromise the Security of the Device or system including the Device if they were successfully used by an attacker shall be unique to the Device.

If Credentials are included in firmware, a firmware upgrade will be needed to change the Credentials. If the same Credentials are used for many Devices, ensuring the firmware of all Devices is updated if the Credentials become compromised is difficult.

SA-009 [*Unique Credentials and Keys*] Credentials and keys should be unique to a Device or Authorized User, and absent a reason to do so, should not be shared among Devices or Authorized Users.

Credentials can contain one or more parts, each of which can have different Security requirements. For example, a Credential can be composed of public Credential information and secret private Credential information, such as two halves of an asymmetric key or simply a username and password. The requirements for storage, transmission and use of these

Credentials are critical to avoiding Device or delegate impersonation to other systems and avoiding impersonation of other systems to the Device.

A Device can store Credentials that identify the Device itself (Device Credential) or that identifies a principal that the Device is empowered to represent (User Credential) to third parties. Availability can be compromised if these Credentials are modified: access to other systems would be interrupted. Confidentiality and authenticity can be compromised if these Credentials are duplicated; other systems can impersonate the principal and/or intercept or modify their communications.

SA-010 [*Storing/Transmitting Credential Information*] When the Device stores or transmits a Credential, that Credential shall be cryptographically protected from improper exposure or modification.

SA-011 [*Storing Private Credential Information*] If the Device stores a secret such as a private key or password, the secret shall be protected from being read or written by any processes that do not require such access.

SA-012 [*Transmitting Private Credential Information*] If the Device transmits a secret such as a private key or password as part of a Credential, it shall be transmitted in a manner that protects its confidentially from all parties that do not require access to it.

SA-013 [*Storing Public Credential Information*] If the Device stores a public key or username as part of a Credential, it shall be stored protected from write access by all processes that do not require write access to it.

SA-014 [*Transmitting Public Credential Information*] If the Device transmits a public key or username as part of a Credential, it shall be transmitted cryptographically protected from modification by all other parties.

A Device can store information used to verify the Identity of Claimants. Availability can be compromised if this information is modified: connections to or from other systems would be inappropriately rejected if the Device refuses a valid Credential. Confidentiality and authenticity can be compromised if the information is modified to represent Credentials under the control of an adversary or if re-usable Credentials such as passwords are acquired by an adversary.

SA-015 [*Safeguarding Private Credential Information*] Any stored Credentials, such as passwords or HMAC keys, shall be protected using cryptographic methods compliant with this standard. Where hash functions are used for this purpose, methods shall be implemented to prevent the brute-forcing of the preimage.

SA-016 [*Safeguarding Trust*] If the Device will authenticate a Credential presented by a Claimant, the information necessary to bind the Identity to the Credential (e.g., a local copy of a public key, a password hash) shall be protected from being modified by any processes that do not require such access.

5.2.2 Validate Certificates

It is not reasonable to check revocation lists every time a certificate is presented. But it is important to have a policy that periodically checks whether some or all of the stored certificates are in a revocation list. Certificate periods of validity can vary.

SA-017 [*Checking Signatures*] When X.509 certificates are used to establish Secure connections (e.g., using TLS or DTLS), the Device shall check whether the certificate is in its store of trusted certificates or was signed by a certificate in its store. This requirement does not preclude using Trust on First Use (TOFU) policies in accordance with SA-018.

SA-018 [*Limited Use of Untrusted Certificates*] A Device shall not authorize presenters of untrusted certificates to perform functions integral to the Security of the system.

SA-019 [*Certificates are Updateable]* If X.509 certificates are used in a system, then any trusted certificates shall be able to be updated. See DAR-004. This is particularly true of trusted root certificates, which may have expiration dates within the intended lifetime of the Device.

SA-020 [*Secure Online Updates*] Trusted root certificate updates should be performed in a Secure online mechanism and should be performed such that trusted root certificates (which need to be continued to be trusted) are updated or replaced before the validity periods end.

SA-021 [*Acquiring Network Time*] When X.509 certificates are used to establish Secure connections (e.g., using TLS or DTLS), the Device shall attempt to acquire and maintain the current date and time from the network. This may be done using the Network Time Protocol (NTP) [2], Precision Time Protocol (PTP) [3], or similar protocol. See SA-024 for additional recommendations regarding secure acquisition of time.

SA-022 [*Checking Certificate Validity Periods*] When X.509 certificates are used, and the Device has acquired the current date and time, the Device shall not accept certificates where the current date and time are outside the certificate's validity period.
Note this requirement does not speak to what a Device should do when X.509 certificates are used but the Device has not (yet) received the current date and time.

SA-023 [*Secure Time Required*] When checking validity periods of X.509 certificates or other time-related functions, the Device should protect itself against attacks on network time via Secure means such as RFC 8915 [4].

SA-024 [*Secure Time*] Devices should acquire time via Secure means such as RFC 8915 [4].

SA-025 [*Time Anomalies*] The Device should implement detection and response measures for anomalous time events (e.g., UTC time should never move backward outside of certain special cases); in such cases the Device should seek another source of time, discard invalid time or take other appropriate measures.

In the absence of a valid time sync, operations that do not require Secure time may continue to operate as normal, but as stated via the requirements above, operations that rely on Secure time should not continue to operate.

SA-026 [*Secure Time not Required*] If validity periods of X.509 certificates or other time related functions are not integral to the Security of the system, and the Device skips time-related X.509 validation and other time-related functions due to the absence of a valid time sync, the Device shall still otherwise validate certificates.

In the absence of a valid time sync, the Device may otherwise operate as if a valid time sync had occurred.

SA-027 [*Confirming Certificate Status Using CertificateStatus Structure*] When X.509 certificates are used to establish TLS or DTLS sessions, the Device shall validate any CertificateStatus structures ("stapled OCSP [20] responses") provided in the handshake with the certificate, as defined in [5]. SA-027 does not require clients to request CertificateStatus; in this case, CertificateStatus structures will not be included in the response.

SA-028 [*Confirming Certificate Status Using OCSP*] When X.509 certificates are used, the Device should implement Online Certificate Status Protocol (OCSP) [20] and should verify certificate validity using OCSP for each intermediate or end-entity certificate not provided with a stapled OCSP response.

Online (e.g., not stapled) OCSP transactions adds delay and is not generally used for real-time validation of certificates. However, periodically using online OCSP to check the status of certificates in a certificate store (including any stored CA certificates and certificates used to trust signatures of OCSP information – including OCSP information in CertificateStatus structures) might be appropriate.

The Device may support Certificate Revocation Lists (CRLs) to determine certificate validity.

CRLs can grow unreasonably large, and it might not be reasonable (and might not even be possible) to check revocation lists every time a certificate is presented.

SA-029 [*Refusing Sessions with Invalid Certificates*] The Device shall not use any certificate it has determined to be invalid (see Section 5.2.2).

SA-030 [*Terminating Sessions of Invalid Credentials*] If the Device determines that the presented Credentials are invalid (or no longer valid), the Device shall not continue to allow access to resources permitted by such Credentials.

SA-031 [*Validating Certificate Identity*] If X.509 certificates are used to establish TLS or DTLS sessions, the Device shall validate the Identity provided with the certificate against any other information the Device has regarding that Entity's Identity.

For most web services, SA-031 includes checking the domain name the Device is attempting to communicate with against the domain name included in the certificate. The Common Name field and `subjectAltName` fields are commonly used to provide Identity information.

For certificates used by Remote Management protocols that allow Device Configuration which can compromise the Security of the Device or system including the Device if successfully used by an attacker, it is important that Identity be validated according to requirements of those protocols.

SA-032 [*Use Certificates for Indicated Purposes*] If X.509 certificates are used, the Device shall ensure each certificate is used only for its indicated logical purpose and each certificate's technical purpose is appropriately reflected within the X.509 Key Usage or X.509 Extended Key Usage X.509 extensions.

For example, the Extended Key Usage field must include the `id-kp-serverAuth` key purpose to authenticate a TLS server. Devices can also have policies that attach authorized uses to specific identities.

For certificates used by Remote Management protocols that allow Device Configuration which can compromise the Security of the Device or system including the Device if successfully used by an attacker, it is important that the Device confirms the certificate can be used for the protocol, according to requirements of the protocol.

5.2.3 User Interfaces, Console Ports and Remote Management Protocols

Some Devices support Remote Management protocols, Console Ports or a user interface that allows Device Configuration. If not sufficiently secured, an attacker can use such protocols and Interfaces to compromise the Security of the Device or the system that includes the Device.

SA-033 [*Supporting Remote Management and User Interface Protocols*] Devices that implement Remote Management and user interface protocols that can compromise the Security of the Device or system shall implement Security functionality for those protocols.

SA-034 [*Not Including Protocols with Known Security Flaws*] If a Remote Management or user interface protocol will be used to allow Device Configuration that can compromise the Security of the Device or system, Devices shall not implement Insecure or deprecated Remote Management or user interface protocols (or protocol versions).

SA-035 [*Implement the Most Recent Specifications and Guidelines*] Remote Management and user interface protocols that are deprecated by the Standards Developing Organization (SDO), industry consortia or alliance with responsibility for the protocol shall not be used.

The requirement of SA-036 includes refraining from use of versions or elements (parameters, options, etc.) of the protocol that have been deprecated by the maintenance organization. Common management and remote access protocols are listed in Annex A.

SA-036 [*Console Port Existence*] Unless a Device requires a Console Port, it should not have one.

SA-037 [*Console Port Accessibility*] If a Console Port is present, and users are not expected to use it, the Console Port should be difficult to access without Specialized or Proprietary Tools.

SA-038 [*Physical Console Port Authentication*] If a Console Port is present, it shall not allow command and control until after Authentication of an Authorized User.

While this standard makes no requirements upon development and debugging phases of product development, it is important that any debugging and/or testing interfaces are disabled, removed or secured on production units.

5.2.4 Web Services

Applications inside Devices are often expected to communicate with websites and cloud services on the Internet. This communication is usually initiated from the Device.

Section 5.3 provides requirements for securing data in transit (between the Device and an external site). Section 5.2.2 provides for validating website X.509 certificates.

SA-039 [*No Device-Stored Cloud Credentials*] Username and password combinations (for the Device to use when logging in to the website/cloud server) should not be stored on the Device. A protocol such as OAuth2 [21] should be used instead of storing username and password combinations. See DAR-001.

SA-040 [*No Hardcoded Credentials*] Device Credentials presented to websites and cloud servers that allow access by the Device to Sensitive Data shall not be hardcoded into firmware.

SA-041 [*X.509 Certificate Device Credentials*] X.509 certificate Device Credentials shall be able to be Securely updated.

Devices are expected to support current best practices related to keys, single sign-on, passwords and other Credentials.

5.2.5 Trust on First Use

SA-042 [*TOFU Policies*] When Trust on First Use (TOFU) is used, end-users should be given the ability to accept or reject the initial trust choice and to review and reconsider their choice in the future.

5.3 Data In Transit is Protected

> *"Protection of the confidentiality and integrity of selected categories of transmitted data via sound cryptographic means, e.g., HMACs, TLS / DTLS, IPsec, or SSH."* See *C2 Consensus* [16] 5.1.3, definition of *Data In Transit Is Protected.*

5.3.1 Physical Networking Technologies Supporting Ethernet MAC

Physical layer networking protocols define mechanisms for transmitting bits of information over a physical medium (including air). Physical networking layer technologies that support a MAC layer consistent with IEEE 802.3 include Wi-Fi (802.11), G.hn, 100BaseT Ethernet, HomePlug, MoCA and other similar mechanisms. These protocols are capable of running IPv4 and IPv6 protocol over their MAC layer.

Implementations of network technologies that go over a medium that is always fully contained within a single physical network rarely include support for encryption at the physical layer. This includes GigE and 100BaseT Ethernet running over dedicated CAT6 or other usable grade of CAT wires.

Physical layer technologies that run over powerline, coax or wireless media define how to encrypt the physical layer technology.

DIT-001 [*Encrypting Physical Layer Technologies*] Devices that communicate using protocols over an Ethernet MAC should support the data link layer encryption mechanisms currently required by the certification authority for that physical layer technology.

DIT-002 [*No Deprecated Physical Layer Encryption*] Deprecated physical layer encryption mechanisms should not be supported.

There are times when commercial requirements require the use of older, deprecated, possibly Insecure protocols for backwards compatibility (or other) purposes; except in those cases, use of deprecated mechanisms is discouraged.

DIT-003 [*Implementing Physical Layer Trust Mechanisms*] If one or more pairing or "trust" mechanisms are defined and recommended for the physical layer technology, at least one such mechanism shall be supported.

For encryption to be useful, Devices need a means to disseminate encryption keys to trusted Devices wanting to join a network or to pair with each other. This requires a method for users to indicate "trust" to be implemented.

Physical layers that support encryption generally define one or more of the following pairing or "trust" mechanisms:

- Entering a passphrase into a user interface: technologies that support passphrases will have minimum requirements for acceptable passphrases.
- Pushbutton pairing: pushbuttons can be physical or implemented in a graphical user interface (GUI).

- QR Codes with a 3rd Device that can scan codes and communicate with Devices.
- Certificates.

5.3.2 Physical Networking Technologies Without Ethernet MAC

Physical layer technologies that define a MAC layer substantively different than Ethernet MAC include 802.15 (Thread, Zigbee), Z-Wave, X10 and other protocols designed for use by constrained Devices.

DIT-004 [*Physical Networking Technologies without Ethernet MAC*] Devices implementing networking technologies that do not support 802.3 MAC frames should support the encryption mechanisms currently recommended for that technology by a recognized certification authority for that technology.

5.3.3 Link-Layer Application Protocols

Some application protocols are designed to only be used on a local network. This is usually done by defining the protocol to be transported directly over the link-layer protocol (e.g., Ethernet MAC) or to use IP link-local addressing. These application protocols are often used without encryption or signed packets.

Examples of such protocols include Multicast DNS (mDNS) (commonly used for discovering Devices and services on the local network), discovery of CoRE resources, SSDP (used for UPnP discovery and advertisement on a local network), LLDP (used to communicate Ethernet topology information), and IEEE 1905.1 (used to communicate physical layer topology and to manage multi access point networks).

No restrictions are placed on use of protocols that only operate over the link layer or use link-layer addressing.

5.3.4 Encrypting IP Transport Protocols

DIT-005 [*Implementing TLS for TCP/IP*] If the Device implements TCP/IP, it should implement TLS.

DIT-006 [*Encrypting TCP/IP with TLS*] If the Device implements TLS, it shall comply with requirements in the section “Minimum Requirements for TLS Clients” in the current revision of NIST SP 800-52 Rev. 2 [6].

DIT-007 [*Encrypting UDP/IP with DTLS*] Devices with support for UDP over IP protocol should implement DTLS 1.2 [22] and future revisions of DTLS.

DIT-008 [*Earlier DTLS Versions*] DTLS 1.0 [23] should not be implemented.

DIT-009 [*Protecting Credentials in Transit*] Device Credentials that could compromise the Security of the system, including the Device, if such Credentials were successfully used by an attacker, shall be cryptographically secured when in transit.

5.3.5 Integrity and Authenticity

DIT-010 [*Integrity and Authenticity in Transit*] The Integrity and Authenticity of data in transit shall be verified using sound cryptographic means if the Security of the system depends on the Integrity and Authenticity of data in transit.

There are applications where such protections are not required for baseline levels of security, such as the minimum necessary to prevent botnet exploitation or pivoting into a network. Manufacturers should carefully consider the threat scenarios (risk assessment) for their application and whether such protections are indeed necessary.

5.4 Data at Rest is Protected

"Protection of the confidentiality and integrity of selected categories of stored data via sound cryptographic means." See *C2 Consensus* [16] 5.1.4, definition of *Data At Rest Is Protected.*

DAR-000 [*Protection of Data at Rest – Confidentiality*] The confidentiality of data at rest shall be ensured using sound cryptographic means.

DAR-001 [*Protection of Data at Rest – Integrity and Authenticity*] The Integrity and Authenticity of data at rest shall be verified using sound cryptographic means.

DAR-001 may be satisfied by using authenticated encryption that also provides confidentiality, or by an additional authenticity control applied to the data. The controls may be applied at the message, file or media level (e.g., Full Disk Encryption) as appropriate. Systems that otherwise swap or page program memory to non-volatile storage need not meet this requirement for data stored in pinned memory.

If usernames and passwords for the Device to log in to a remote server are stored on the Device, DAR-000 and DAR-001 require that they be cryptographically secured. See also SA-039.

DAR-002 [*Data Protection when not Powered*] Data that is stored on the Device that, if compromised, would enable attacks at scale such as botnet attacks (e.g., Credentials) shall be protected against an attacker using Specialized Tools, even for power-off conditions and subcomponents of the Device that could be physically removed and Sensitive Data extracted.

Encryption at rest can meet this requirement; simply soldering a Device to a circuit board does not constitute protection.

DAR-003 [*Trusted Root Storage*] A receiver, at a minimum, shall have sufficient non-volatile storage to store the set of trusted root certificates and the set of trusted OCSP [20] responder certificates.

DAR-004 [*Ability to Update Trusted Roots*] A Device shall have a mechanism for updating the set of trusted root certificates and trusted OCSP [20] responder certificates due to expiration, revocation, update or other reasons.

DAR-005 [*Updating Roots Via Image*] In the event a root certificate or OCSP [20] responder certificate needs to be replaced or updated, updates shall be performed by an update of the software code image.

DAR-006 [*Initial Trusted Roots*] An initial set of trusted root certificates and OCSP [20] responder certificates shall be installed as part of the software code image installed during the manufacturing process, and the certificate store shall be periodically verified by the Device to be valid and unchanged from the original image. The store shall be verified at least at every boot-up and should be verified at least monthly. The store should be verified by comparing a computed Secure hash against a stored Secure hash, by a code signing mechanism, or by an equivalent cryptographic mechanism at boot-up.

DAR-007 [*Runtime Storage of Certificates*] In the event a Device copies trusted certificates into runtime memory (e.g., when expanding a compressed software image), the certificate storage memory area shall be configured as "read only" such that any attempt to write in this area is prevented (e.g., by hardware memory management).

DAR-008 (Removed)

DAR-009 [*Frequency of Integrity and Authenticity Checks*] Verification of the integrity and authenticity of data at rest should occur at a frequency appropriate to the risk of adversary modification of stored data.

DAR-010 [*Secure Boot*] Each firmware and software executable at boot time should have an associated signature meeting the requirements of CRY-003.

The Secure boot may have multiple data stores for storing such certificates and hashes.

DAR-011 [*Untrusted Executable*] If Secure Boot is used and an executable file cannot be validated as trusted, it shall be treated as untrusted and shall not be executed.

5.5 Industry Accepted Protocols are Used for Communications

> *"Use of secure, widely used protocols, excluding deprecated and replaced versions and protocols, for communications to and from the device."* See *C2 Consensus* [16] 5.1.5, definition of *Industry Accepted Protocols are Used for Communications*.

This section contains general requirements intended to apply to all protocols used by a Device. Specific requirements for protocols used to establish encrypted or otherwise secured connections are in Section 5.3. Specific requirements for protocols used to remotely manage Devices are in Section 5.2.3.

IAP-001 [*Use Standardized Protocols*] Communications protocols used should be compliant with:

1) A technical standard from an international standards body including ISO, IEC, ITU, IETF, ETSI; or

2) A technical standard from a recognized regional standards body including CTA, SCTE, ATSC, or regionally-accredited standards bodies; or
3) A technical standard from a recognized industry alliance or consortia such as Wi-Fi Alliance, Bluetooth SIG, or the ZigBee Alliance.

IAP-002 [*No Deprecated* Protocols] Communications protocols that are deprecated by any of the following authorities: IETF, NIST, CERT; or by the SDO, industry consortia or alliance with responsibility for the protocol, including Wi-Fi Alliance, Bluetooth SIG, etc., should not be used.

Note that SA-035 includes more stringent requirements for remote access or management protocols.

IAP-003 [*Use Recommended Profiles/Constraints*] Profiles and constraints for communications protocols that are described by any of the following authorities: IETF, NIST, CERT; or by the SDO, industry consortia, or alliance with responsibility for the protocol, including Wi-Fi Alliance, Bluetooth SIG, etc., should be followed.

The above two constraints recommend against using deprecated, abandoned or Insecure technologies, features or options of communications protocols.

5.6 Data Validation

"Parsing and limiting input data to prevent it from being used directly as code, commands, or other execution flow inputs; and encoding output data in a form appropriate to and limited to its intended usage." See *C2 Consensus* [16] 5.1.6, definition of *Data Validation*.

The scope of input data is all data received and processed by the Device, including data strings entered through the Device UI or web interface (including passwords), files uploaded to the Device, container meta-data, messages (e.g., SOAP and XML) and other data transferred via APIs and services.

Devices can be attacked by malicious data strings (e.g., executable code disguised as input data) or files for the purpose of subverting the Device, causing it to fail or behave incorrectly. The input data can originate from the user or from other Devices or services. Attackers can use automated tools to find and exploit failures to validate input data. To protect the Device, the user and the network from attacks, all data coming into the Device must be thoroughly validated. It is also important that data leaving the Device has been validated or is otherwise well formed, safe to pass on and as intended.

DV-001 [*Canonicalization of Input*] The Device shall convert all input to a canonical form prior to further use.

DV-002 [*Validation of input*] The Device shall validate input data for length, character type and acceptable values or ranges.

DV-003 [*Filtering of Input*] The Device should use allow-listing rather than deny-listing when filtering input data.

DV-004 [*Common Web Application Attacks*] For Devices that present a web page as the user interface for an administrative console that allows Device Configuration changes, and Devices that support similar administrative functions via APIs, the Device shall implement measures to prevent the following common attack types defined in the 2017 OWASP Top 10 Application Security List [7]:

1) Injection
2) XML External Entities (XXE)
3) Cross-Site Scripting (XSS)
4) Insecure Deserialization

DV-005 [*Embedded Systems Attacks*] The Device should implement measures to prevent common attack types by following Secure coding practices for input data. The Device should use a type-safe language; if this is not possible, the Device should implement bounds checking and use safe typing and safe string handling functions. The Device should use some form of stack protection such as canaries or Address Space Layout Randomization (ASLR).

Examples of common attack types (see the CWE list [24] for details):

- Buffer Overflow (Stack/Heap) (CWE-121/122),
- Integer Overflow (CWE Category ID 872),
- Format String (CWE Category ID 133),
- Invalid pointer handling (e.g., double free/dangling pointer) (CWE Category ID 465, CWE-415, 416).

This item is extremely important. However, for many kinds of Device architectures, it is difficult or impossible to independently test it Post-Market. It is included here to emphasize this importance, although it uses the keyword ‘should’ because a ‘shall’ requirement would not always be testable in the general case.

DV-006 [*CWE Programming Language Views*] The CWE list [24] outlines common issues found in specific programming languages such as C, C++, Java and PHP. These lists should be reviewed and effective countermeasures incorporated into software development policies.

DV-007 (Removed)

Devices may allow for file upload or transfer to the Device. Example use cases include firmware update, updating device configuration and addition of user-specific customizations. If any mechanism exists for the device to accept file upload or transfer of files for storage on a file system under control of the Device, then the file name, file size and file contents should be verified as described below. Files failing the verifications may be rejected or rendered harmless through sanitization.

DV-008 [*Subfiles*] If a file containing subfiles will have any of its subfiles rendered into individual files, each of those files shall be validated as described in each of the other Data Validation requirements. Example formats that include subfiles are TAR and ZIP.

DV-009 [*File Size*] File sizes shall be limited to be smaller than a size that would harmfully exhaust Device resources. The size of files shall be controlled when the file is accepted as input, such as through a file upload function, and again prior to the completion of any on-device transformation that may increase the file's size.

It is important to control the expanded file size to limit the damage caused by uploaded decompression bombs.

DV-010 [*File Format*] If the Device accepts file upload or transfer of data of defined types, the file contents should be verified to conform to the stated type. Files should not be accepted if the file type, as determined by file name and/or MIME type, does not match the file contents, or if there is a discrepancy between file header parameters and the observed properties of the file. For example, a function that accepts an PNG image should reject a TAR file as input.

DV-011 [*Non-Execution of Uploads*] Files uploaded or transferred to the device, other than through an authorized software update process, shall be rendered non-executable if the Device file system supports doing so.

Controlling executability of uploaded files limits the damage caused by uploads of files that are valid both as the expected file type and as an executable. Examples include the GIFAR attack files, which are valid GIF images and simultaneously valid Java executable archives.

5.7 Event Logging

"A limited persistent record in the device of relevant events, secured and available to authorized users." See *C2 Consensus* [16] 5.1.7, definition of *Event Logging*.

ELG-001 [*Event Logging*] If the Device supports Secured access or management (see Section 5.2.3), the Device should support logging of Security events. Such events include (but are not limited to) attempts to log in to any remote access or administrative interfaces and periods of degraded Device performance.

ELG-002 [*Enabling/Disabling Event Logging*] The Device should log events by default unless there are performance, Device lifetime, or powering implications, in which case the Device may allow logging to be enabled or disabled.

ELG-003 [*Logging Firewall Events*] If the Device supports a firewall and supports Remote Management or a user interface (see Section 5.2.3), the Device shall meet Logging requirements of ICSA Labs Firewall Certification Criteria Baseline Module [8].

ELG-004 [*Log Memory Location*] The memory location for logging should be appropriate for the planned number of logs stored and available space. The location for log storage

should consider how logs are accessed to ensure appropriate security precautions are included by design. For non-volatile memory, wear-leveling from program and erase cycles should be considered when designing for expected performance and device lifetime.

The intention is to capture and provide users with access to information that could be indicative of an attack.

5.8 Cryptography

> *"Where cryptography is used, use open, published, proven, and peer-reviewed cryptographic methods with appropriate parameter, algorithm and option selections."* See *C2 Consensus* [16] 5.1.8, definition of *Cryptography*.

"The purpose of cryptography is to ensure confidentiality, integrity and availability. Example uses may include protecting data in transit (outside the device and in certain cases within the device), protecting data at rest, authentication, authorization, etc. Determining the data to be protected requires some judgement; see related sections. However, examples of such data may include sensitive data (credentials, etc.) and user defined data (PII, access credentials, etc.)." See *C2 Consensus* [16] 5.1.8, discussion of Cryptography.

Security systems must be designed and implemented properly and utilize underlying cryptographic subsystems correctly to be effective. The entire cryptographic system internal and external to the Device must be managed securely end-to-end. Even when Secure and proven algorithms are used, the system can still be vulnerable through other parts of the systems – e.g., Insecure key management, side channel attacks, etc. Even when using a proven off-the-shelf solution, any points of interface with the solution can be vulnerable and must be scrutinized.

IoT Devices might be resource constrained so lightweight, updateable and scalable cryptographic methods are recommended. NIST has a project on selecting methods suitable for lightweight cryptography. More information is available at the NIST Lightweight Cryptography project site [25].

CRY-001 [*Cryptographic Methods*] The Device shall only use strong, industry-standard, open and peer-reviewed cryptographic methods. The Device shall not use proprietary cryptographic methods.

Due to the complexity in designing and implementing cryptographic algorithms, product designers are expected to only use publicly-disclosed algorithms that have been subject to peer review and academic study. Use of proprietary or altered cryptographic algorithms provides little to no value to the Security of a system and is of significantly less value than using known-secure algorithms with an appropriately generated and managed key.

CRY-002 [*Deprecated Cryptographic Methods*] The Device shall not use deprecated, disallowed, weak or broken cryptographic methods as per NIST SP 800-131A Rev. 2 [1].

CRY-003 [*Security Strength*] The Device shall implement a cryptographic Security strength equivalent to at least 112 bits of Security. The Device should implement a cryptographic Security strength equivalent to at least 128 bits of Security. Consult NIST SP 800-57 Table 2, section 5.6.1 Comparable Algorithm Strengths [9].

Security strength is generally the workload required to identify the key used, expressed in bits. Asymmetric algorithms often require an increased number of key bits to provide a strength equivalent to symmetric algorithms. For example, 3072-bit RSA is equivalent in Security strength to 128-bit AES-128. However, the Security strength of cryptographic algorithms represents more than this simple equivalency of the number of bits of the cryptographic key and must consider attacks that reduce the difficulty of attacking that particular algorithm. For example, there are analytical methods that reduce the workload of an attack and thus the Security strength must include such factors.

CRY-004 [*Device Lifecycle*] The Device shall use cryptographic methods and parameters (e.g., key length) that are considered to be Secure over the lifetime of the Device in alignment with NIST SP 800-131A Rev. 2 [1]. Cryptographic methods should be updateable.

CRY-005 [*Lightweight Cryptography*] If using lightweight cryptography, the Device should use methods that comply with ISO/IEC 29192 [26].

CRY-006 [*Randomness*] Randomly generated values for cryptographic use shall be generated using either a pseudorandom generator as per NIST SP 800-90A Rev. 1 [10] or a true source of randomness as per FIPS-140-3 [11].

CRY-007 [*Key Insertion and Storage*] Key insertion shall be done using a Secure method. The Device shall store unencrypted keys in a tamper-resistant location such as a Secure Element (e.g., Hardware Security Module or Trusted Platform Module).

CRY-008 [*Key Provisioning*] A Secure process shall be used for the provisioning of keys in alignment with NIST SP 800-133 Rev. 2 [12].

CRY-009 [*No Key Reuse*] Cryptographic key material shall be protected from unauthorized disclosure through reuse – e.g., by using different keys for development and production deployment.

CRY-010 [*Key Length*] The Device shall use cryptographic keys of sufficient type and length for the use case and expected lifetime of the Device. Symmetric encryption keys should be of a type and length considered resistant to post-quantum attacks.

CRY-011 [*Time Source*] The Device shall use a Secure time source when cryptography relies on it.

CRY-012 [*Side Channel Leakage*] The Device should be resistant to known side channel attacks.

5.9 Patchability

"The ability to verifiably update a device's configuration firmware or software, post-market, with patches that are authenticated to ensure that they have been deployed by an authorized entity as well as to verify the integrity of the patch." See *C2 Consensus* [16] 5.1.9, definition of *Patchability*.

PAT-001 [*Patchability*] The Device shall have a cryptographically Secure mechanism for authorized entities to update a Device's firmware and software after development or installation.

PAT-002 [*Security of Patch*] The update mechanism shall utilize cryptographically Secure mechanisms that assure integrity and authenticity of the firmware or software update.

PAT-003 [*Update Authorization*] When updating, the Device shall verify that the firmware or software update was created by an authorized author.

PAT-004 [*Unattended Patching*] Devices should provide an option to allow users to choose to install security patches without further user intervention.

The update mechanism may utilize mechanisms which assure confidentiality of the firmware or software update.

Firmware or software updates may be delivered by any appropriate mechanism, including via a network and via portable storage Devices (e.g., a USB memory stick).

PAT-005 [*Secure Default State*] Devices shall define Secure Default Settings for configurable security options such as recognized authenticators and firewall settings.

Requirements PAT-006 through PAT-008 rely upon the Secure Default State as defined above in PAT-005.

PAT-006 [*Restoring to Secure Default State*] Devices shall provide a mechanism for users to recover from potentially insecure configuration choices by reverting configurable security options to the defined Secure Default State.

PAT-007 [*Patching Secure Default State Definition*] Updates to the Device's firmware and software that add new configurable security options shall update the Secure Default State to contain Secure defaults for those options.

PAT-008 [*Vulnerabilities in Secure Default State Definition*] The Manufacturer shall, upon discovery that the configuration defined in the Secure Default State is Insecure, provide updates to the Secure Default State such that future use of the mechanism

for restoring to Secure Default State will restore corrected Configuration rather than known-insecure Configuration settings. This requirement shall apply to devices during the devices' Security Support Phase and does not impose any requirements on the length of the Security Support Phase.

5.10 Reprovisioning

> *"The ability for authorized users to securely reconfigure and redeploy a device post-market, especially to return the product to factory defaults or an authorized restore point, and securely remove data collected by the device (that is not essential to its configuration), within a defined period established by the organization."* See *C2 Consensus* [16] 5.1.10, definition of *Reprovisioning (which includes definition of Deprovisioning).*

What is user-specific data will depend on the use case of the Device; however, generally location information and information that can be personally identified or linked to an individual are considered user-specific data.

REP-001 [*Deprovisioning*] At least one Field Deprovisioning procedure that includes Secure purge or destroy operations (as defined in NIST SP 800-88 Rev.1 [13]), cryptographic erasure, or physical removal of the media containing user-specific data shall be provided in a place and manner available to expected Authorized users.

REP-002 [*Deprovisioning Period*] The Field Deprovisioning procedure in REP-001 shall not be limited by the defined period for Reprovisioning.

Reprovisioning might be limited by a defined period by the Manufacturer; REP-002 requires that a deprovisioning method is always possible even after that period has expired.

REP-003 [*Deprovisioning by Users*] The Field Deprovisioning procedure in REP-001 shall be completable by Authorized users without the use of Specialized Tools.

REP-003 requires that the Manufacturer determine who is authorized to Deprovision a Device. The Deprovisioning procedures should be made available to anyone in physical possession, meaning that the Authorized user may be a new homeowner in the case of a smart home device, or a reseller or lessor of the Device. Remote Deprovisioning should be subject to greater scrutiny, since there exists the potential for abuse over the internet. Therefore, remote Deprovisioning may be reserved to the Manufacturer, or may not be provided as an option at all.

REP-004 (Removed)

REP-005 [*Reprovisioning or Disposal*] Either a Reprovisioning procedure to restore a Device to a usable state OR a notice that Reprovisioning is not possible and therefore Secure disposal is required shall be provided in Device documentation or on Device packaging.

REP-006 [*Effectiveness of Deprovisioning and Reprovisioning*] Following the application of any Manufacturer-specified Deprovisioning or Reprovisioning procedure, it shall be infeasible to extract from the Device any data pertaining to any previous owner using state-of-the-art laboratory techniques.

REP-007 [*Cryptographic Erasure*] Cryptographic Erasure, if used for Reprovisioning or Deprovisioning, shall be completed in accordance with the procedures and use cases in NIST SP 800-88 Rev.1, section 2.6 [13].

REP-008 [*Physical Media Removal*] Physical media removal for Deprovisioning and replacement for Reprovisioning, if used, shall be performable by the Device owner using only Commonly Available Tools, and without Specialized or Proprietary Tools.

6 PRODUCT LIFECYCLE MANAGEMENT CAPABILITIES – BASELINE

This section has requirements for important capabilities that are in scope for the organization, rather than the Device. Device capabilities are typically observable on a given Device. These product lifecycle management capabilities are activities of the manufacturing organization (or otherwise responsible development organization) that are important in the context of overall Security of the Device.

6.1 Vulnerability Submission and Handling

6.1.1 Definition

A defined and managed process for accepting Vulnerability notifications and acting on them.

6.1.2 Requirements

VUL-001 *[Vulnerability Identification]* The Manufacturer shall have a process for ongoing identification of potential vulnerabilities that defines activities such as the Manufacturer's participation in industry threat-sharing programs or a mechanism for accepting unsolicited notifications of potential vulnerabilities.

VUL-002 *[Vulnerability Contact]* The Manufacturer shall have a publicly-stated point of contact for outside reporting of potential vulnerabilities, such as a web page with a contact form or a documented Security email address.

VUL-003 *[Default Vulnerability Contact]* As unsolicited Vulnerability reports can be sent to security@example.com, where example.com is the Manufacturer's email domain, the Manufacturer should ensure that Vulnerability disclosure messages received at that address are directed to staff responsible for processing Vulnerability disclosures.

VUL-004 *[Vulnerability Handling]* The Manufacturer should, upon identification, evaluate a potential Vulnerability in terms of risk factors, scope of affected products, availability of mitigations, and other factors, and prioritize action accordingly. Organizations should allocate resources to address identified Vulnerabilities according to that prioritization.

6.2 EoL/EoS Updates and Disclosure

6.2.1 Definition

A defined Manufacturer policy covering the handling of any post end-of-life (EoL) or end-of-service (EoS) Device Vulnerabilities, if and how updates will be available, and what to do with the Device at EoL or EoS.

6.2.2 Requirements

EoL/EoS-001 *[Notifications]* The Manufacturer shall indicate in a place and manner available to customers and users, including those in secondary or resale markets, its policy on disclosure of, and providing Security updates for, discovered Device Vulnerabilities, and what to do with the Device after the end of the Security Support Phase.

This requirement can be met by publishing the EoL/EoS policy on a public-facing web page. This topic must be considered carefully by the Manufacturer; EoL and EoS policies are connected to Vulnerability handling, product lifecycle, terms of service and more.

EoL/EoS-002 *[Guidance for Secure Disposal]* Guidance for the Secure disposal of removed media should be provided in Device documentation.

6.3 Device Intent Documentation

6.3.1 Definition

An explanation of the Device's as-designed network usage that is made available by the Manufacturer publicly, in product documentation, or other means for Device users.

6.3.2 Requirements

DIN-001 [*Device Intent Documentation*] The Manufacturer shall maintain publicly-available documentation (where the provenance and integrity of the documentation is verifiable) of the as-designed intended network usage in normal operation mode, including websites and URLs, and ports.

DIN-002 [*Device Memory Documentation*] The Manufacturer shall maintain publicly-available documentation of the components of the Device that can contain data pertaining to the owner or user and the types of data that can be stored.

7 BIBLIOGRAPHY

As noted in the Scope [1], this standard specifies certain Device Security capabilities and related organizational Security capabilities and recommendations for Devices and Device systems. This establishes a 'baseline', also as noted in the Scope [1]. A baseline is not a final destination for

all possible products; it is a common capabilities basis upon which to build for Security appropriate to the risk assessment and use case of the Device and its requirements.

Use case analysis, risk assessment and requirements planning are properly done at the early stage of product development. This standard is directly based on *The C2 Consensus on IoT Device Security Baseline Capabilities* [16]. Additional resources for product planning include:

- NIST NISTIR 8228*, Considerations for Managing Internet of Things (IoT) Cybersecurity and Privacy Risks*, June 2019, https://csrc.nist.gov/publications/detail/nistir/8228/final.
- ETSI 'DTS/CYBER-0039' Work Item *CYBER; Cyber Security for Consumer Internet of Things*, https://portal.etsi.org/webapp/WorkProgram/Report_WorkItem.asp?WKI_ID=54761.
- ETSI EN 303 645 V2.1.1*, Cyber Security for Consumer Internet of Things: Baseline Requirements*, https://www.etsi.org/deliver/etsi_en/303600_303699/303645/02.01.01_60/en_303645v020101p.pdf.
- GSMA IoT Security Guidelines and Assessment, https://www.gsma.com/iot/iot-security/iot-security-guidelines/.
- NIST NISTIR 8259, *Foundational Cybersecurity Activities for IoT Device Manufacturers*, May 2020, https://csrc.nist.gov/publications/detail/nistir/8259/final.
- UK Department for Digital, Culture, Media & Sport*, Code of Practice for Consumer IoT Security*, https://www.gov.uk/government/publications/code-of-practice-for-consumer-iot-security.
- Global Platform, *Security Evaluation Standard for IoT Platforms (SESIP) v1.1*, https://globalplatform.org/specs-library/.
- Galen Hunt, George Letey, and Edmund B. Nightingale, Microsoft. *The Seven Properties of Highly Secure Devices.* https://www.microsoft.com/en-us/research/wp-content/uploads/2017/03/SevenPropertiesofHighlySecureDevices.pdf (accessed 28 April 2022).

8 REGARDING FUTURE SECURE CAPABILITIES – PHASE IN OVER TIME

"The following items are considered significant enough that they should be baseline capabilities. However, for various reasons they cannot be considered baseline at this time. The expectation is that they will become baseline and developers should carefully consider the capabilities in their planning." See *C2 Consensus* [16] Annex A, *Regarding Future Secure Capabilities – Phase in Over Time.*

8.1 Device Intent Signaling

"*Means for the device to provide information to routers or firewalls upstream what kind of traffic the device was intended to produce."* See *C2 Consensus* [16] A.1, definition of *Device Intent Signaling.*

DIS-001 [*Security Assessment*] Device manufactures should assess their Devices using a risk management process such as the NIST Cybersecurity Framework [15]. This Assessment informs the Security objectives, which in turn inform what specific Security controls will be employed on the Device.

DIS-002 [*Security Objectives*] Device manufacturers should maintain a written description of the Security objectives for each Device or device class.

DIS-003 [*Minimum Objectives*] Device Security objectives should describe, at minimum, the impact the Device could potentially have on the broader infrastructure in which it operates, including how the Devices might be prevented from being used in a botnet attack. [16]

DIS-004 [*Minimum Controls*] The Security controls should address at least the minimum Security objectives, including how the Device will restrict the volume, destination and content of its emitted communications in order to avoid harming the Security of the systems that will relay, process or receive those messages.

8.2 Device Network Onboarding

"Network Onboarding" for a Device is the means to enable a network operator or Device manager to cryptographically ensure that a Device, when first attached to a network, is identified, authenticated and authorized. It is the process of Authenticating the Device, Authorizing that Device with Credentials, and Configuring it to be able to communicate within the desired Security domain. Correct identification of the Device and explicit, non-automated, approval from the network manager are both critical to the exchange.

DOB-001 *[Bootstrapping Identity]* A Device that is not pre-provisioned to a specific deployment should* have a bootstrapping Identity that is intended for use solely for bootstrapping purposes. At a bare minimum, this Identity should* be a public/private key pair and should be an X.509 certificate.

DOB-002 *[Onboarding Mechanism]* Devices that require network bootstrapping should* employ a network bootstrapping mechanism that establishes proof to the Device that it belongs on a particular network.

*It is anticipated that this "should" will become a "shall" when these items become part of the Baseline in the future; this information is provided now as a recommendation and for future planning.

Annex A. (Informative) Common Management and Remote Access Protocols

The following management and remote access protocols are commonly implemented by Devices to allow user, operator and machine-to-machine (M2M) Configuration and firmware upgrades over the Internet Protocol (IP). Outdated and Insecure implementations of these protocols are often responsible for allowing a Device to be compromised.

Specification	Protocol Name	Responsible Organization	Link
RFC 3411	Simple Network Management Protocol (SNMP)	IETF	https://tools.ietf.org/html/rfc3411
RFC 4253	The Secure Shell (SSH) Transport Layer Protocol	IETF	https://tools.ietf.org/html/rfc4253
RFC 6120	Extensible Messaging and Presence Protocol (XMPP)	IETF	https://tools.ietf.org/html/rfc6120
RFC 6421	Network Configuration Protocol (NETCONF)	IETF	https://tools.ietf.org/html/rfc6241
RFC 7252	The Constrained Application Protocol (CoAP)	IETF	https://tools.ietf.org/html/rfc7252
RFC 7540	Hypertext Transfer Protocol Version 2 (HTTP/2)	IETF	https://tools.ietf.org/html/rfc7540
RFC 8040	RESTCONF Protocol	IETF	https://tools.ietf.org/html/rfc8040
TR-069	CPE WAN Management Protocol (CWMP)	Broadband Forum (BBF)	https://www.broadband-forum.org/technical/download/TR-069.pdf
TR-369	User Services Platform (USP)	Broadband Forum (BBF)	https://usp.technology/specification/
LwM2M	Lightweight M2M (LwM2M)	Open Mobile Alliance (OMA)	http://openmobilealliance.org/release/LightweightM2M
MQTT	MQTT Version 5	OASIS	https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html

Consumer Technology Association Document Improvement Proposal

If in the review or use of this document a potential change is made evident for safety, health or technical reasons, please email your reason/rationale for the recommended change to standards@CTA.tech.

Consumer Technology Association
Technology & Standards Department
1919 S Eads Street, Arlington, VA 22202
FAX: (703) 907-7693 standards@CTA.tech

Consumer Technology Association™