इंटरनेट ऑफ थिंग्स सुरक्षा और गोपनीयता : आकलन और मूल्यांकन

Internet of Things Security & Privacy
: Assessment and Evaluation

ICS 35.030

BUREAU OF INDIAN STANDARDS
MANAKBHAVAN, 9 BAHADURSHAHZAFAR MARG
NEW DELHI 110002

September 2024 **Price Group**

Information System Security and Privacy Sectional Committee, LITD 17
(Formal Clauses to be added later on)

FOREWORD

This Indian Standard may be adopted by the Bureau of Indian Standards, after the draft finalized by Information System Security and Privacy Sectional Committee may be approved by the Electronics and Information Technology Divisional Council.

This document is applicable and useful to various stakeholders, including:

- IoT device manufacturers, seeking to enhance the security and privacy features of their products.
- System integrators and solution architects, tasked with creating secure IoT ecosystems.
- IT and security professionals responsible for safeguarding IoT deployments.
- Regulators and compliance officers overseeing adherence to IoT security and privacy standards.

Contents

Introduction

IoT has rapidly evolved, embedding itself in our daily lives and various industries, presenting a pressing need to safeguard the confidentiality, integrity, and privacy of data collected and transmitted by these devices. The proliferation of Internet of Things (IoT) devices has ushered in a new era of convenience and efficiency, yet this progress is accompanied by a growing concern for security and privacy. As more devices connect to the internet, they become potential targets for cyberattacks, data breaches, and privacy violations.

This document aims to address these challenges by offering guidance on securing IoT devices and preserving user privacy, thereby ensuring the continued growth and trustworthiness of the IoT landscape.

The assessment of Internet of Things is a way to identify the mistakes in application logic, configurations, implementation and deployment that jeopardize the security of IoT devices, networks, servers, web interfaces, mobile apps or data of IoT Ecosystem.

The intent of this document is to provide the approach and methodology for assessment and evaluation of IoT Device and to list out a detailed compliance checklist.

This document provides comprehensive guidance on establishing robust security and privacy measures for IoT (Internet of Things) devices.

This guidance specifically addresses the critical aspects of IoT device security and privacy. It aims to equip IoT device manufacturers, system integrators, and other stakeholders with the knowledge and tools required to:

- Design and produce IoT devices with robust security features that mitigate vulnerabilities and resist unauthorized access.
- Implement privacy-preserving mechanisms that ensure the responsible handling of sensitive user data.
- Adhere to established IoT security and privacy standards and regulations.
- Foster a culture of continuous improvement to adapt to emerging threats and evolving technologies.

1. Scope

This document provides the approach, and methodology for the assessment and evaluation to verify the implementation of controls for Internet of Things (IoT) Devices. This document refers to the controls as specified in IS/ISO/IEC 27400 and IS/ISO/IEC 27402 for IoT Devices and provides some additional controls for IoT Devices.

2. References

The standards given below contains provisions, which through reference in this text constitute provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreement based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed as follows:

IS/ISO/IEC 27400:2022 - Cybersecurity — IoT security and privacy — Guidelines

IS/ISO/IEC 27402:2023 Cybersecurity — IoT security and privacy — Device baseline requirements

Open Web Application Security Project (OWASP) Application Security Verification Standard (ASVS) Version 4.0.3

3. Acronyms

This clause provides a comprehensive list of acronyms used throughout the document.

ASLR	Address Space Layout Randomization
ASVS	Application Security Verification Standard
CPU	Central Processing Unit
DEP	Data Execution Prevention
IoT	Internet of Things
JTAG	Joint Test Action Group
OWASP	Open Web Application Security Project
PII	Personally Identifiable Information
PCBA	Printed circuit board assembly
SoC	System on Chip
SE	Secure Element
SWD	Serial Wire Debug
TPM	Trusted Platform Module
TEE	Trusted Execution Environment
UART	Universal Asynchronous Receiver-Transmitter
USB	Universal Serial Bus

4. Terms and Definitions

For the purpose of this document, the terms and definitions given in IS/ISO/IEC 27000 and IS/ISO/IEC 27400 apply.

5. Risk Assessment and Threat Modelling

5.1 General

In the context of IoT device security and privacy, it is necessary that IoT devices undergo a comprehensive risk assessment process at the device level, which is an integral part of a broader system-level risk assessment. This assessment should encompass several key considerations, like:

1. Intended Outcomes: The risk assessment process shall take into account the intended outcomes specific to the intended use case of the IoT device.

2. Stakeholder Needs and Expectations: The risk assessment process should also consider the needs and expectations of all relevant stakeholders, including those who are part of networks to which the IoT device connects. This assessment should address both physical and logical undesired effects.

3. Device Constraints: Recognizing that IoT devices often operate under constraints such as limited battery life, minimal memory, or constrained processing capabilities, these limitations should inform the risk treatment process.

The following guidelines and processes should be adhered to while conducting the risk assessment:

a) **Product Differentiation:** Determine if separate risk assessment and treatment processes are warranted for different IoT devices.
b) **Risk Treatment Options:** Select appropriate risk treatment options based on the outcomes of the risk assessment.
c) **Control Implementation:** Identify all necessary controls required to implement the chosen risk treatment options.
d) **Security and Privacy Features Identification:** Identify all security and privacy features associated with the IoT device that stem from the identified control.
e) **Feature Verification:** Compare the identified features to ensure that none are omitted inadvertently.
f) **Statement of Applicability:** Create a Statement of Applicability that includes the essential features and provides justifications for their inclusion or exclusion.
g) **Adherence to Other Standards:** If other standards related to device requirements are applicable, ensure compliance with the requirements of those standards.
h) **Risk Treatment Plan:** Develop a comprehensive risk treatment plan that outlines the steps and actions to mitigate identified risks.
j) **Risk Owner Communication:** Communicate the risk treatment plan to the designated risk owner, along with any residual risks. Obtain the risk owner's approval of the plan and their acknowledgment of any remaining risks, where applicable.

Furthermore, IoT devices shall implement the identified necessary features and controls outlined in the Statement of Applicability. This implementation should extend to all requisite features and controls.

Documentation for the entire risk assessment process, security and privacy features, omitted requirements, vulnerability disclosure processes, and security support policy shall remain available and accessible throughout the supported lifetime of IoT devices.

5.2 Risks

The security and privacy of IoT devices are susceptible to a variety of threats and vulnerabilities. A comprehensive understanding of these risks is essential for effective risk management. Table 1 outlines some of the risks associated with IoT device security and privacy:

Table 1: Risks

Sl. No.	Risk
R1	Failure to define, approve, and communicate an IoT security policy may result in inadequate measures to mitigate security threats, leaving devices vulnerable to exploitation.
R2	Undefined roles and responsibilities for IoT security may lead to ambiguity in accountability, potentially resulting in overlooked security measures and increased susceptibility to breaches.
R3	Incomplete identification of assets during IoT device development may overlook critical components, leading to inadequate protection of sensitive data and assets.
R4	Absence of mechanisms to apply insights from past security incidents may perpetuate vulnerabilities, increasing the likelihood and impact of future breaches.
R5	Unprotected application layer debugging interfaces pose a risk of unauthorized access and exploitation, compromising the integrity and confidentiality of the device.
R6	Failure to enable memory protection controls exposes the IoT device to memory-based attacks, jeopardizing the confidentiality and integrity of stored data.
R7	Active on-chip debugging interfaces pose a threat of unauthorized access and manipulation, potentially leading to exploitation and compromise of device functionality.
R8	Lack of implementation of trusted execution may allow unauthorized access to critical functions and data, compromising the confidentiality and integrity of the device.
R9	Insecure storage of sensitive data and cryptographic assets increases the risk of unauthorized access and compromise, potentially leading to data breaches and exploitation.
R10	Inadequate random number generation may lead to predictable cryptographic keys and compromise the confidentiality and integrity of communication channels.
R11	Exposure of sensitive traces on the printed circuit board increases the risk of physical tampering and unauthorized access, potentially compromising device security.
R12	Unencrypted inter-chip communication exposes sensitive data to interception and manipulation, increasing the risk of data breaches and unauthorized access.
R13	Lack of code signing and validation exposes the device to the risk of executing malicious or tampered firmware, compromising device integrity and functionality.
R14	Failure to overwrite sensitive data in memory increases the risk of data leakage and unauthorized access, potentially leading to exposure of sensitive information.
R15	Inadequate isolation between firmware apps may facilitate unauthorized access and compromise of sensitive data and device functionality.

R16	Failure to configure secure compiler flags exposes firmware to various exploitation techniques, compromising device security and integrity.
R17	Lack of code protection in microcontrollers increases the risk of unauthorized access and manipulation of firmware, compromising device functionality and security.
R18	Use of banned C functions poses a risk of vulnerabilities and exploitation, potentially compromising device security and integrity.
R19	Incomplete documentation of third-party components and vulnerabilities increases the risk of exploitation and compromise through known vulnerabilities.
R20	Failure to review code for hardcoded credentials exposes devices to unauthorized access and exploitation, compromising device security.
R21	Inactive Intellectual Property protection technologies may lead to unauthorized reproduction and exploitation of device functionality, compromising intellectual property rights.
R22	Lack of support for disabling debugging interfaces in microcontrollers increases the risk of unauthorized access and manipulation, compromising device security.
R23	Inadequate protection from physical attacks increases the risk of reverse engineering and exploitation, compromising device security and confidentiality.
R24	Insufficient integration of security measures may result in vulnerabilities that could lead to malfunction or compromise of the device, posing safety risks.
R25	Failure to protect data-in-transit exposes sensitive information to interception and manipulation, compromising data confidentiality and integrity.
R26	Lack of validation of server connections exposes the device to the risk of connecting to malicious servers, compromising data confidentiality and integrity.
R27	Failure to mutually authenticate wireless communications increases the risk of unauthorized access and interception, compromising data confidentiality and integrity.
R28	Unencrypted wireless communications expose sensitive information to interception and manipulation, compromising data confidentiality and integrity.
R29	Failure to pin digital signatures to trusted servers exposes devices to the risk of connecting to malicious servers, compromising data confidentiality and integrity.
R30	Inadequate monitoring and logging of device states, events, and network traffic hinder detection and response to security incidents, increasing the risk of exploitation and compromise.
R31	Insecure storage of logs increases the risk of unauthorized access and manipulation, potentially compromising the integrity and confidentiality of logged information.
R32	Absence of tamper resistance and detection features increases the risk of physical tampering and unauthorized access, compromising device security.
R33	Delivery of IoT devices with insecure settings and configurations increases the risk of exploitation and compromise, jeopardizing device security.
R34	Unauthorized modification of IoT device configurations poses a risk of exploitation and compromise, compromising device security and functionality.
R35	Use of common values for critical security parameters increases the risk of exploitation and compromise, compromising device security and confidentiality.
R36	Absence of security controls against firmware reverse engineering increases the risk of unauthorized access and manipulation, compromising device security and integrity.

R37	Failure to implement authentication mechanisms increases the risk of unauthorized access to IoT systems and services, compromising data confidentiality and integrity.
R38	Inadequate protection of stored and transmitted data increases the risk of unauthorized access and manipulation, compromising data confidentiality and integrity.
R39	Vulnerability to OS Command Injection poses a risk of unauthorized access and manipulation, compromising device security and integrity.
R40	The absence of defined update procedures heightens the risk of unauthorized updates and exploitation.
R41	Unauthorized initiation of software updates for IoT devices can lead to exploitation of vulnerabilities or implantation of malicious code.
R42	Vulnerability to time-of-check vs time-of-use attacks during updates increases the risk of installing malicious or tampered firmware, compromising device integrity.
R43	Failure to validate firmware upgrade files before installation poses a security risk by potentially allowing the installation of malicious or tampered firmware, while neglecting verification of the cryptographic chain of trust during updates exacerbates this risk, jeopardizing device integrity and potentially compromising user privacy.
R44	Ability to downgrade to old firmware versions increases the risk of exploiting known vulnerabilities, compromising device security and functionality.
R45	Inadequate monitoring and reporting of vulnerabilities increases the risk of exploitation and compromise, jeopardizing IoT device as well as user security.
R46	Failure to wipe firmware and sensitive data upon tampering or receipt of invalid messages increases the risk of unauthorized access and manipulation, compromising device security.
R47	Lack of guidance on proper IoT device usage increases the risk of misuse and exploitation, compromising device security and functionality.
R48	Inadequate evaluation of supplier security measures increases the risk of acquiring insecure IoT device components, jeopardizing overall IoT device security.
R49	Insufficient or inaccurate design details can lead to undetected counterfeit components or hidden malware, compromising device integrity.
R50	Failure to implement comprehensive threat mitigation can result in the integration of counterfeit or tainted components, exposing the device to security vulnerabilities.
R51	Inadequate or outdated malware detection tools increase the risk of undetected malicious code being integrated into the final product.
R52	Ignoring supply chain risks can lead to the introduction of compromised components, which can undermine the security and functionality of the IoT device.
R53	Unauthorized disclosure of IoT device security information increases the risk of exploitation and compromise, jeopardizing device security and confidentiality.
R54	Inadequate removal of data and licensed software prior to disposal or re-use increases the risk of unauthorized access and exposure of sensitive information, compromising data confidentiality and integrity.

R55	Absence of a secure function to delete user data increases the risk of unauthorized access and exposure of sensitive information, compromising data confidentiality and integrity.
R56	Failure to incorporate privacy-enhancing features increases the risk of privacy violations and unauthorized access to personal data, compromising user privacy.
R57	Failure to ensure the strictest privacy settings by default increases the risk of privacy violations and unauthorized access to personal data, compromising user privacy.
R58	Lack of privacy notice detailing the data collection purpose increases the risk of unauthorized data collection and misuse, compromising user privacy.
R59	Failure to obtain consent before data collection increases the risk of unauthorized data collection and misuse, compromising user privacy.
R60	Failure to address end users' privacy concerns in device design increases the risk of privacy violations and unauthorized access to personal data, compromising user privacy.
R61	Lack of regular review of privacy controls increases the risk of privacy violations and unauthorized access to personal data, compromising user privacy.
R62	Failure to assign unique cryptographic keys and certificates increases the risk of unauthorized access and impersonation, compromising device privacy and security.
R63	Inadequate mapping of device identifiers to specific individuals increases the risk of privacy violations and unauthorized access to personal data, compromising user privacy.
R64	Failure to enforce authorized access increases the risk of unauthorized access and manipulation
R65	Unauthorized data collection risks compromising user privacy and autonomy.
R66	Insufficient authentication may lead to unauthorized privacy preference manipulation.
R67	Lack of secondary verification could result in irreversible harm to IoT users.
R68	Absence of an accountability framework increases the likelihood of data mishandling and privacy breaches, diminishing transparency and accountability in data processing practices.
R69	Insecure storage of PII of IoT device owner can result in data theft, identity fraud, and legal consequences.
R70	Poorly managed PII protection increases the risk of unauthorized access and disclosure.
R71	Failure to identify, document, and regularly update all relevant legal, statutory, regulatory, and contractual requirements related to IoT device security may result in non-compliance, legal penalties, and compromised device security.

5.3 Prioritizing Security and Privacy Risks

After identifying potential risks, it's essential to prioritize them based on their impact and likelihood. This prioritization informs resource allocation and risk mitigation efforts.

Factors to consider in prioritizing risks:

1. Impact: Assess the potential consequences of a security or privacy breach. Consider the financial, operational, reputational, and legal ramifications.

2. Likelihood: Estimate the likelihood of each risk occurring. Consider historical data, industry trends, and specific contextual factors.

3. Risk Tolerance: Define the organization's risk tolerance level. Some risks may be accepted if they fall within acceptable limits, while others require immediate mitigation.

4. Dependencies: Recognize interdependencies among risks. Addressing one risk may mitigate or exacerbate others.

5. Regulatory Compliance: Prioritize risks that have implications for regulatory compliance, as non-compliance can result in legal penalties.

By conducting a thorough risk assessment and prioritizing security and privacy risks, organizations can develop a targeted strategy for implementing security controls and privacy safeguards. This approach ensures that resources are allocated effectively to protect IoT devices against the most significant risks.

6. IoT Device Security & Privacy Verification Checkpoints

IoT device security is a critical component of ensuring the overall security and privacy of an IoT system. Devices are the frontline defense against potential threats and vulnerabilities. This clause provides the IoT device security and privacy verification checkpoints mapped to the risks given in table of this document and to the controls as specified in IS/ISO/IEC 27400, IS/ISO/IEC 27402 for IoT Devices, additional controls specified in this document and the risks identified in clause 5 of this document. These checkpoints are derived from IS/ISO/IEC 27400, IS/ISO/IEC 27402 and OWASP ASVS 4.0.3 Appendix C.
Security Checkpoints for IoT service developer and IoT service provider are given in Table 2.
Security Checkpoints for IoT user are given in Table 3.
Privacy checkpoints for IoT service developer and IoT service provider are given in Table 4.
Privacy checkpoints for IoT user are given in Table 5.

Table 2: Security Checkpoints for IoT service developer and IoT service provider

Sl. No.	Control		Verification Checkpoint	Associated Risk
	Title	**Description**		
1	Policy for IoT security	Control-01: A policy for IoT security should be defined, approved by management, published, communicated to relevant personnel and relevant external parties and reviewed at planned intervals or if significant changes occur.	V1.1 Ensure that a policy for IoT security is defined, approved by management, published, communicated to relevant personnel and relevant external parties and reviewed at planned intervals or if significant changes occur.	R1
2	Organization of IoT security	Control-02: Roles and responsibilities for security of IoT should be defined and allocated.	V2.1 Confirm that roles and responsibilities for IoT security are defined and allocated, with accountability clearly established.	R2
3	Asset management	Control-03: Information, IoT devices and systems and their	V3.1 Confirm that the IoT device developer has identified all assets	R3

		functions and operations to be protected should be identified.	(Information, IoT devices and systems) to be protected across the entire development process of the IoT device.	
4	Equipment and assets located outside physical secured areas	Control-04: Specific security measures should be applied to IoT equipment and assets which are located or operated outside physical secured areas.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-
5	Secure disposal or re-use of equipment	Control-05: All items of equipment containing storage media should be verified to ensure that any sensitive data and licensed software has been removed or securely overwritten prior to disposal or re-use.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-
6	Learning from security incidents	Control-6: Knowledge gained from analysing and resolving IoT security incidents should be used to reduce the likelihood or impact of future incidents.	V6.1 Ensure that mechanisms are in place to apply knowledge gained from analyzing and resolving IoT device security incidents to reduce the likelihood or impact of future incidents.	R4
7	Secure IoT system engineering principles	Control-7: Principles for engineering secure IoT systems that address designing and implementation of security functions, defence in depth and hardening of systems and software should be applied to the development of IoT systems.	V7.1 Verify that application layer debugging interfaces such USB, UART, and other serial variants are disabled or protected by a complex password.	R5
			V7.2 Verify that memory protection controls such as ASLR and DEP are enabled by the embedded/IoT operating system, if applicable.	R6
			V7.3 Verify that on-chip debugging interfaces such as JTAG or SWD are disabled or that available protection mechanism is enabled and configured appropriately.	R7
			V7.4 Verify that trusted execution is implemented and enabled, if available on the device SoC or CPU.	R8
			V7.5 Verify that sensitive data, private keys and certificates are stored securely in a Secure Element, TPM, TEE (Trusted Execution Environment), or protected using strong cryptography.	R9
			V7.6 Verify usage of cryptographically secure pseudo-random number generator on embedded device (e.g., using chip-	R10

			provided random number generators).	
			V7.7 Verify that sensitive traces are not exposed to outer layers of the printed circuit board.	R11
			V7.8 Verify that inter-chip communication is encrypted (e.g. Main board to daughter board communication).	R12
			V7.9 Verify the device uses code signing and validates code before execution.	R13
			V7.10 Verify that sensitive information maintained in memory is overwritten with zeros as soon as it is no longer required.	R14
			V7.11 Verify that the firmware apps utilize kernel containers for isolation between apps.	R15
			V7.12 Verify that secure compiler flags such as -fPIE, -fstack-protector-all, -Wl, -z, noexecstack, -Wl, -z, noexecheap are configured for firmware builds.	R16
			V7.13 Verify that micro controllers are configured with code protection.	R17
8	Secure development environment and procedures	Control-08: Secure development environment and procedures should be applied to the development of IoT systems.	V8.1 Verify that any use of banned C functions are replaced with the appropriate safe equivalent functions.	R18
			V8.2 Verify that each firmware maintains a software bill of materials cataloguing third-party components, versioning, and published vulnerabilities.	R19
			V8.3 Verify all code including third-party binaries, libraries, frameworks are reviewed for hardcoded credentials (backdoors).	R20
			V8.4 Verify that any available Intellectual Property protection technologies provided by the chip manufacturer are enabled.	R21
			V8.5 Verify that only micro controllers that support disabling debugging interfaces (e.g. JTAG, SWD) are used.	R22
			V8.6 Verify that only micro controllers that provide substantial	R23

			protection from de-capping and side channel attacks are used.	
9	Security of IoT systems in support of safety	Control-09: Security principles in support of safety should be applied to the development of IoT systems.	V9.1 Ensure the integration of security measures into IoT device development to maintain safety, including mechanisms to detect and halt erroneous or corrupted control data to prevent malfunctions.	R24
10	Security in connecting varied IoT devices	Control-10: An IoT system should be designed and implemented to ensure and maintain security in connecting varied IoT devices.	V10.1 Verify that the firmware apps protect data-in-transit using transport layer security.	R25
			V10.2 Verify that the firmware apps validate the digital signature of server connections.	R26
			V10.3 Verify that wireless communications are mutually authenticated.	R27
			V10.4 Verify that wireless communications are sent over an encrypted channel.	R28
			V10.5 Verify that the firmware apps pin the digital signature to a trusted server(s).	R29
11	Verification of IoT devices and systems design	Control-11: Design and implementation of IoT devices and IoT systems should be verified.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-
12	Monitoring and logging	Control-12: States, events and network traffic of IoT devices and systems should be monitored and logged.	V12.1 Ensure that states, events, and network traffic of IoT devices are monitored and logged.	R30
13	Protection of logs	Control-13: Logs for IoT devices and systems should be protected from leakage, destruction and unintended alteration.	V13.1 Validate that logs for IoT devices are protected from leakage, destruction, and unintended alteration.	R31
			V13.2 Verify the presence of tamper resistance and/or tamper detection features.	R32
14	Use of suitable networks for the IoT systems	Control-14: Applied network and communication technologies for IoT and systems should meet the needs of communication function, capacity and security, and of function and performance of IoT devices.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-
15	Secure settings and	Control-15: IoT devices and services should be delivered	V15.1 Verify that IoT devices are delivered with secure settings and configurations.	R33

	configurations in delivery of IoT devices and services	with secure settings and configurations.	V15.2 Ensure that only authorized entities can modify the configuration settings of the IoT device if they are modifiable.	R34
			V15.3 Verify that IoT devices ensure that common values for critical security parameters, such as global private keys or standard passwords, are replaced by values that are unique per device or explicitly defined by an appropriate external entity before they are put into operation.	R35
			V15.4 Verify security controls are in place to hinder firmware reverse engineering (e.g., removal of verbose debugging symbols).	R36
16	User and device authentication	Control-16: Authentication function of users and IoT devices for accessing IoT systems and services should be implemented and applied.	V16.1 Confirm the implementation and application of authentication mechanisms for IoT devices accessing IoT systems and services.	R37
			V16.2 Verify that IoT devices protect stored and transmitted data, including configuration settings, identifying data, user data, event logs, and sensitive security parameters against unauthorized access, modification, and disclosure, while also safeguarding software from unauthorized access and modification, utilizing cryptography for data confidentiality and integrity.	R38
			V16.3 Verify that the application and firmware components are not susceptible to OS Command Injection by invoking shell command wrappers, scripts, or that security controls prevent OS Command Injection.	R39
17	Provision of software and firmware updates	Control-17: Mechanism for updating software and firmware of IoT devices and systems should be designed, implemented and operated.	V17.1 Ensure that the update procedure is defined and includes validation of updates, configuration choices for automatic/manual updates, scheduling options, and notification settings.	R40
			V17.2 Ensure that software updates for IoT devices are securely initiated by authorized entities and that interruptions during updates minimize potential harm.	R41

			V17.3 Verify that the firmware update process is not vulnerable to time-of-check vs time-of-use attacks.	R42
			V17.4 Verify the device uses code signing and validates firmware upgrade files before installing. The update should verify the cryptographic chain of trust with the root of trust.	R43
			V17.5 Verify that the device cannot be downgraded to old versions (anti-rollback) of valid firmware.	R44
18	Sharing vulnerability information	Control-18: Vulnerabilities of IoT devices, systems and services should be monitored and informed to the IoT users and relevant parties along with associated risks.	V18.1 Ensure that vulnerabilities of IoT devices are actively monitored and reported to IoT users and relevant parties along with associated risks.	R45
19	Security measures adapted to the life cycle of IoT system and services	Control-19: Security measures of the IoT system and service should be adapted to and kept during the stages of the life cycle, including their development, operation, maintenance and destruction.	V19.1 Verify that the device wipes firmware and sensitive data upon detection of tampering or receipt of invalid message.	R46
20	Guidance for IoT users on the proper use of IoT devices and services	Control-20: The IoT users should be provided with guidance on the proper use of IoT devices with risks and undesirable effects of IoT system and service that can be derived from improper use of IoT devices.	V20.1 Verify that IoT users are provided with guidance on the proper use of IoT devices, including risks and potential undesirable effects.	R47
21	Determination of security roles for stakeholders	Control-21: Roles of IoT service developer, IoT service provider and other stakeholders in security of IoT system and service should be determined and agreed among relevant parties.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-
22	Management of vulnerable devices	Control-22: Vulnerable IoT devices should be detected, recorded, and alerts provided to IoT users and administrators of these devices.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-

23	Management of supplier relationships in IoT security	Control-23: Specifications and supporting obligations of suppliers for information security of IoT device and IoT service should be managed by the acquiring organization based on the contracts with suppliers.	V23.1 Ensure that the acquiring organization has a system in place to evaluate supplier security measures according to local laws and regulations.	R48
			V23.2 Design and architecture details till the PCBA and SoC level to be provided to aid in counterfeit mitigation and malware detection.	R49
			V23.3 Threat mitigation strategies for tainted and counterfeit products shall be implemented as part of product development.	R50
			V23.4 One or more up-to-date malware detection tools shall be deployed as part of the code acceptance and development processes. Malware detection techniques shall be used before final packaging and delivery (e.g., scanning finished products and components for malware using one or more up-to-date malware detection tools).	R51
			V23.5 Supply chain risk identification, assessment, prioritization, and mitigation shall be conducted.	R52
24	Secure disclosure of Information regarding security of IoT devices	Control-24: Information on the IoT device relevant to security of IoT services should be documented and disclosed only to the parties that require them.	V24.1 Ensure that documentation detailing IoT device security information is present and restrict disclosure solely to pertinent parties.	R53

Table 3: Security Checkpoints for IoT user

Sl. No.	Control		Verification Checkpoint	Associated Risk
	Title	**Description**		
1	Contacts and support service	Control-25:IoT users should only choose IoT devices and IoT services that provide contact information for support service.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-
2	Initial settings of IoT device and service	Control-26: Initial settings of IoT device and service should be applied correctly.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-

3	Deactivation of unused devices	Control-27: IoT devices should be deactivated and credentials revoked when they are no longer in use.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-
4	Secure disposal or re-use of IoT device	Control-28: Data and licensed software stored in IoT device should be removed or securely overwritten prior to disposal or re-use.	V28.1 Ensure that data and licensed software stored in IoT device are removed or securely overwritten prior to disposal or re-use.	R54
			V28.2 Verify the IoT device has a secure function allowing only authorized entities to delete relevant user data stored on the device in any memory type.	R55

Table 4: Privacy checkpoints for IoT service developer and IoT service provider

S.no	Controls		Verification Checkpoint	Associated Risk
	Title	**Description**		
1	Prevention of privacy invasive events	Control-29: Privacy enhancing capabilities should be built in the IoT devices and IoT services.	V29.1 Audit the IoT device to confirm the incorporation of privacy-enhancing features.	R56
2	IoT privacy by default	Control-30: Stakeholders in an IoT system should ensure that without any IoT user interaction or intervention, the strictest privacy settings apply by default.	V30.1 Ensure that stakeholders of IoT device ensure the strictest privacy settings by default without requiring IoT user interaction or intervention.	R57
3	Provision of privacy notice	Control-31-1: The IoT user should be provided with a privacy notice which states personal data collected by the IoT device and IoT service and purpose of its use.	V31.1.1 Confirm that IoT users are provided with a privacy notice detailing the collection of personal data by IoT devices and the purpose of its use.	R58
		Control-31-2: Consent of the IoT user to the privacy notice should be obtained before collecting the personal data or changing the purpose of use.	V31.2.1 Verify that the consent to privacy notice is obtained from IoT users before data collection by IoT device or changes in use.	R59
4	Verification of IoT functionality	Control-32: Independent verification of IoT device, data components and IoT service components should be supplied to provide visibility and assurance to all stakeholders that the IoT	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-

		device or service is operating as per stated objectives.		
5	Consideration of IoT users	Control-33: End users' privacy requirements and concerns should be addressed in designing the IoT device and service.	V33.1 Validate that end users' privacy requirements and concerns are addressed in the design of IoT devices.	R60
6	Management of IoT privacy controls	Control-34: The effectiveness of privacy controls in the IoT device and service should be reviewed, and new privacy risks be identified on a continuous basis considering the evolving privacy needs of end users and regulatory requirements.	V34.1 Obtain a declaration from the IoT device developer confirming regular review of privacy controls' effectiveness and continuous identification of new privacy risks.	R61
7	Unique device identity	Control-35-1: IoT system developers (especially device developers) should use a method that uniquely identifies each IoT device to improve privacy for identifying IoT device suspected to be relevant to a cyber incident.	V35.1.1 Ensure that unique cryptographic keys and certificates are assigned to each individual IoT device to enhance privacy and aid in identifying devices relevant to cyber incidents.	R62
		Control-35-2: IoT service providers should use, if required, a method to allow a unique mapping between a given IoT device and an IoT user to improve privacy for identifying the mapping between IoT device and IoT user(s).	V35.2.1 Ensure a documented process exists to map device identifiers to specific individuals or user profiles for IoT devices. This mapping should be securely maintained and accessible solely by authorized IoT users.	R63
8	Fail-safe authentication	Control-36: The system should ensure that implemented authentication cannot be bypassed, tampered, or falsified in any reasonable method.	V36.1 Verify IoT devices enforce authorized access to interfaces with proper authentication and resist any attempts to bypass, tamper with, or falsify implemented authentication measures.	R64
9	Minimization of indirect data collection	Control-37: Collection of data from indirect sources should be minimized or not collected at all.	V37.1 Verify that IoT devices minimize the collection of indirect data (data collected without user participation) to only what is necessary for operation, unless explicit user consent is obtained.	R65
10	Communication of privacy preferences	Control-38: User preferences of privacy controls should be only added, modified, or deleted when the authorized	V38.1 Validate that user preferences for privacy controls can only be added, modified, or deleted when the	R66

		user is authenticated to the system.	authorized user is authenticated to the IoT device.	
11	Verification of automated decision	Control-39: Automated decision provided by IoT services should be verified.	V39.1 Ensure that there is a secondary, independent verification for automated decisions made by IoT devices that could cause irreversible harm to users.	R67
12	Accountability for stakeholders	Control-40: Accountability for various stakeholders should be established.	V40.1 Review documentation to confirm the presence of an accountability framework that outlines data privacy responsibilities for the IoT device.	R68
13	Unlinkability of PII	Control-41: The IoT system should ensure that the PII of the user owning a device cannot be identified.	V41.1 Ensure that PII of the device owner is saved securely with proper access control in place.	R69
14	Sharing information on PII protection measures of IoT devices	Control-42: PII protection measures related to privacy risk in IoT devices should be appropriately managed and only disclosed to the parties that require them.	V42.1 Ensure that PII protection measures related to privacy risk in IoT devices are appropriately managed and only disclosed to the parties that require them.	R70

Table 5: Privacy checkpoints for IoT user

S.no	Controls		Verification Checkpoint	Associated Risk
	Title	Description		
1	User consent	Control-43: Consent for use of personal data for the IoT device and service should be provided only after considering the necessity and its probable impact if there is a data breach. Consent should be withdrawn if the IoT output is no longer needed or if there is a concern with the IoT device or service.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-
2	Purposeful use for connecting with other devices and services	Control-44: Connection of IoT device and service with other devices or services should be allowed only if there is a valid need.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-
3	Certification/validation of PII protection	Control-45: Certification or validation of privacy protection features with respect to the IoT device and service should be sought.	Not Applicable for IoT Device Assessment (Applicable for IoT Ecosystem)	-
4	Legal, statutory, regulatory and	Control-46: Legal, statutory, regulatory and contractual requirements relevant to IoT	V46.1 Verify that all legal, statutory, regulatory, and contractual requirements relevant to IoT device	R71

	contractual requirements	Device security and the organization's approach to meet these requirements shall be identified, documented and kept up to date.	security, along with the organization's approach to meet these requirements, are identified, documented, and regularly updated.	

Implementation of controls shall be evaluated through verification of check points listed in above tables. Evaluation methodology for verification of check points is given in Annex-A. Description of assurance levels for compliance process and to categorize levels of security and privacy of IoT Devices is given in Annex-B.

Annex A

Evaluation Methodology

This annexure provides comprehensive evaluation methodologies for assessing security and privacy checkpoints in IoT devices. These methodologies are designed to ensure a thorough examination and mitigation of potential risks associated with IoT devices. Evaluation methodology for each check point is given in Table 6.

Table 6: Evaluation Methodology

Sl. No.	Security & Privacy Checkpoint	Evaluation Methodology
1.	V1.1 Ensure that a policy for IoT security is defined, approved by management, published, communicated to relevant personnel and relevant external parties and reviewed at planned intervals or if significant changes occur.	a) Examine the IoT security policy document, management approval records, and communication logs. b) Conduct interviews with key personnel to confirm their awareness and understanding of the policy.
2.	V2.1 Confirm that roles and responsibilities for IoT security are defined and allocated, with accountability clearly established.	a) Review documents outlining the defined roles and responsibilities. b) Interview personnel to confirm their understanding of their roles and responsibilities. c) Check for evidence of accountability mechanisms, such as audit reports or performance reviews.
3.	V3.1 Confirm that the IoT device developer has identified all assets across the entire development process of the IoT device.	a) Verify that IoT device developer asset inventory includes all hardware, software, firmware, data, network components, and third-party dependencies. b) Ensure that asset identification is documented for all phases: requirements gathering, design, development, testing, deployment, and maintenance. c) Evaluate the tools and techniques employed for asset identification (e.g., automated discovery tools, manual audits, threat modelling).
4.	V6.1 Ensure that mechanisms are in place to apply knowledge gained from analysing and resolving IoT device security incidents to reduce the likelihood or impact of future incidents.	a) Review the Incident Response Plan for procedures on documenting, analyzing, and resolving security incidents. b) Review a sample of incident response cases to check that corrective actions were implemented. c) Check for the existence of a knowledge base or repository where lessons learned are stored. d) Verify that recent incidents and their lessons learned are included in training sessions.
5.	V7.1 Verify that application layer debugging interfaces such USB, UART, and other serial variants are	a) Identification of the availability of debugging interfaces such as USB, UART, and other serial variants through the Datasheet of the SoC being used in the device under test

	disabled or protected by a complex password.	b) Verification and validation of the ports/interfaces enabled in the production devices and the related access control mechanism for protection of the same as declared in the vendor documentation c) Testing, in presence of OEM team, to verify the enabling/disabling of all the ports and debugging interfaces such as USB, UART, and other serial variants using their relevant hardware-based debuggers and access control mechanisms in case the interface is enabled. d) Process verification of the manufacturing facility to validate the vendor's claim regarding the debugging interfaces which are closed/disabled during provisioning. [For instance, through Block connection diagram depicting pin connections between the host microcontroller and its interactions with various subcomponents/peripherals.]
6.	V7.2 Verify that memory protection controls such as ASLR and DEP are enabled by the embedded/IoT operating system, if applicable.	a) Testing, in presence of OEM team, to verify the declared memory protection controls available and enabled in the device using command line-based tools/commands or any other open-source tool like DEP, EMET tool etc.
7.	V7.3 Verify that on-chip debugging interfaces such as JTAG or SWD are disabled or that available protection mechanism is enabled and configured appropriately.	a) Identification of the availability of debugging interfaces such as USB, UART, and other serial variants through the Datasheet of the SoC being used in the device under test b) Verification and validation of the ports/interfaces enabled in the production devices and the related access control mechanism for protection of the same as declared in the vendor documentation c) Testing, in presence of OEM team, to verify the enabling/disabling of all the ports and debugging interfaces such as USB, UART, and other serial variants using their relevant hardware-based debuggers and access control mechanisms in case the interface is enabled. d) Process verification of the manufacturing facility to validate the vendor's claim regarding the debugging interfaces which are closed/disabled during provisioning. [For instance, through Block connection diagram depicting pin connections between the host microcontroller and its interactions with various subcomponents/peripherals.]
8.	V7.4 Verify that trusted execution is implemented and enabled, if available on the device SoC or CPU.	a) Identifying whether TEE/SE/TPM is available or not in the device through the SoC datasheet and technical documentation submitted by the vendor. Further assessment is done on the basis of scenarios as applicable to device as defined below:

		i. CASE 1: TEE/SE/TPM is not available: No further assessment ii. CASE 2: TEE/SE/TPM is available and enabled: Verification through code-review that crypto functions are called through TEE/SE/TPM APIs. iii. CASE 3: TEE/SE/TPM is available but not enabled by the vendor: Termed as non-conformance to the requirement. OEM is required to enable and implement the TEE/SE/TPM.
9.	V7.5 Verify that sensitive data, private keys and certificates are stored securely in a Secure Element, TPM, TEE (Trusted Execution Environment), or protected using strong cryptography.	a) Identifying all the keys and certificates being used in the device eco-system, sensitive data and their storage mechanism(s); and verification through: i. Testing, in presence of OEM team ii. Code review iii. Process audit of the key -life cycle process
10.	V7.6 Verify usage of cryptographically secure pseudo-random number generator on embedded device (e.g., using chip-provided random number generators).	a) Verification of the documentation provided by the vendor regarding the random number generators being used in the device. b) Verification through code-review that random number generators or related libraries as applicable are being used in the device.
11.	V7.7 Verify that sensitive traces are not exposed to outer layers of the printed circuit board.	a) Conduct a thorough review of the PCB design schematics and layout. b) Verify that sensitive traces carrying critical data or signals (such as cryptographic keys, sensitive communications lines, or high-frequency signals) are routed on inner layers of the PCB.
12.	V7.8 Verify that inter-chip communication is encrypted (e.g. Main board to daughter board communication).	a) Analyze the device's firmware for implemented encryption mechanisms, focusing on inter-chip communication routines. b) Verify the methods of encryption key generation, distribution, and storage. c) Monitor encryption in inter-chip communication by connecting the IoT device to the appropriate test equipment (e.g. logic analyser).
13.	V7.9 Verify the device uses code signing and validates code before execution.	a) Testing, in presence of OEM team, to verify the following: i. Device boots up successfully with the documented secure boot process when a valid boot image is provided. ii. Device does not boot up when a tampered boot image (like with missing signature, invalid signature) is provided.
14. 14	V7.10 Verify that sensitive information maintained in memory is	a) Determine the types of sensitive information handled by the IoT device (e.g., passwords, encryption keys, personal data).

	overwritten with zeros as soon as it is no longer required.	b) Document specific memory locations or buffers where sensitive information is stored during processing. c) Perform a static code analysis to verify that sensitive information is overwritten with zeros. Look for explicit memory clearing functions (e.g., memset(), SecureZeroMemory()) used in the code. d) Use debugging tools to monitor memory regions before and after sensitive data is used. e) Confirm that memory regions previously containing sensitive information are overwritten with zeros once the data is no longer required. f) Perform memory dumps and analyze the dumps for any traces of sensitive data.
15.	V7.11 Verify that the firmware apps utilize kernel containers for isolation between apps.	a) Examine the device's technical documentation to understand its architecture and app isolation mechanisms. b) Look for references to kernel containers, containerization frameworks (e.g., Docker, LXC), or other isolation techniques. c) Access the device's operating system (OS) through secure shell (SSH) or serial connection. d) Execute apps and attempt to access resources or data from other apps, verifying the isolation boundaries. e) List and inspect running containers using container management tools. f) Verify that each app runs within its container and check the isolation parameters (e.g., namespaces, cgroups).
16.	V7.12 Verify that secure compiler flags such as -fPIE, -fstack-protector-all, -Wl,-z,noexecstack, -Wl,-z, noexecheap are configured for firmware builds.	a) Examine the build scripts (e.g., Makefile, CMakeLists.txt) to identify the compiler and linker flags being used. b) Verify the presence of the following or similar compiler flags in build system file configuration: c) -fPIE (Position Independent Executable) d) -fstack-protector-all (Enables stack protection for all functions) e) -Wl,-z,noexecstack (Prevents execution of code on the stack) f) -Wl,-z,noexecheap (Prevents execution of code on the heap)
17.	V7.13 Verify that micro controllers are configured with code protection.	a) Identify the specific code protection features supported by the microcontroller (e.g., Flash lock bits, code readout protection, secure boot). b) Connect the microcontroller to a debugger or programming tool to access and review its protection settings.
18.	V8.1 Verify that any use of banned C functions are replaced with the appropriate safe equivalent functions.	a) Secure code review [both automated and manual], in presence of OEM team, using a licensed static analysis tool through any of the following approaches: i. Visit to the evaluation agency by the vendor with the firmware code and installing the licensed static analysis

		tool available with the evaluation agency in their systems. [Recommended] ii. Visit to the evaluation agency by the vendor with the firmware code and any licensed static analysis tool available with them and demonstrating the code review activity in the presence of representatives of evaluation agency. iii. Giving a remote access of the systems at vendor site to the evaluation agency for installing their licensed static analysis tool available with them. iv. Giving a remote access of the systems at vendor site to the evaluation agency containing the firmware code along with the licensed static analysis tool available with the vendors.
19.	V8.2 Verify that each firmware maintains a software bill of materials cataloguing third-party components, versioning, and published vulnerabilities.	a) Verification of the submitted list of third-party components by running automated tools like FACT on the firmware. b) Identifying vulnerabilities in the third-party component(s) through publicly available vulnerability databases. c) Verification and validation of the process defined by the vendor for providing regular security updates and patches for the firmware to address any known vulnerabilities in third -party components.
20.	V8.3 Verify all code including third-party binaries, libraries, frameworks are reviewed for hardcoded credentials (backdoors).	a) Independent secure code review [both automated and manual] using a licensed static analysis tool through any of the following approaches: i. Visit to the evaluation agency by the vendor with the firmware code and installing the licensed static analysis tool available with the evaluation agency in their systems. [Recommended] ii. Visit to the evaluation agency by the vendor with the firmware code and any licensed static analysis tool available with them and demonstrating the code review activity in the presence of representatives of evaluation agency. iii. Giving a remote access of the systems at vendor site to the evaluation agency for installing their licensed static analysis tool available with them. iv. Giving a remote access of the systems at vendor site to the evaluation agency containing the firmware code along with the licensed static analysis tool available with the vendors.
21.	V8.4 Verify that any available Intellectual Property protection technologies provided by the chip manufacturer are enabled.	a) Testing, in presence of OEM team, to verify the enabling of the Intellectual Property protection technologies provided by the chip manufacturer, if available.

22.	V8.5 Verify that only micro controllers that support disabling debugging interfaces (e.g. JTAG, SWD) are used.	a) Evaluate the availability of debugging interfaces such as USB, UART, and other serial variants through the datasheet of the System on Chip (SoC) utilized in the device under test. b) Confirm and validate the enabled ports/interfaces in the production devices, alongside the access control mechanisms implemented for their protection, as stipulated in the vendor documentation. c) Conduct testing, with the Original Equipment Manufacturer (OEM) team present, to verify the enabling or disabling of all ports and debugging interfaces such as USB, UART, and other serial variants. d) Utilize relevant hardware-based debuggers and access control mechanisms to ensure the interfaces are properly managed when enabled. e) Verify the processes at the manufacturing facility to substantiate the vendor's claim that debugging interfaces are closed or disabled during provisioning. This can be achieved by reviewing block connection diagrams that illustrate pin connections between the host microcontroller and its interactions with various subcomponents and peripherals.
23.	V8.6 Verify that only micro controllers that provide substantial protection from de-capping and side channel attacks are used.	a) Review datasheets, technical specifications, and security documentation provided by microcontroller manufacturers. b) Obtain sample microcontrollers or access to development boards/kits for evaluation purposes. c) Perform physical penetration tests to assess resistance against de-capping and attempts to extract sensitive information from the microcontroller's internals.
24.	V9.1 Ensure the integration of security measures into IoT device development to maintain safety, including mechanisms to detect and halt erroneous or corrupted control data to prevent malfunctions.	a) Examine design documents to verify the inclusion of security features such as data validation, encryption, authentication, and fail-safe mechanisms. b) Perform code reviews to identify potential security and safety flaws and ensure adherence to secure coding practices. c) Develop and execute test cases that simulate erroneous or corrupted control data scenarios, focusing on both security and safety impacts.
25.	V10.1 Verify that the firmware apps protect data-in-transit using transport layer security.	a) Use network monitoring tools like Wireshark to capture traffic between the IoT device and its communication partners. b) Perform operations that involve data transmission (e.g., sending commands or data). c) Ensure that the data is encrypted using TLS. Look for indications of encryption, such as the presence of TLS handshake messages and encrypted data packets.
26.	V10.2 Verify that the firmware apps validate	a) Prepare test environments with both legitimate and malicious server certificates.

	the digital signature of server connections.	b) Use a test server with a valid digital signature and a test server with an invalid or compromised signature. c) Connect the IoT device to the test server with a valid digital signature. d) Verify that the firmware successfully validates the signature and establishes a secure connection. e) Connect the IoT device to the test server with an invalid or compromised digital signature. f) Verify that the firmware rejects the connection attempt and handles the error appropriately.
27.	V10.3 Verify that wireless communications are mutually authenticated.	a) Testing, in presence of OEM team, to verify the process of mutual authentication as laid down in the documentation by the vendor.
28.	V10.4 Verify that wireless communications are sent over an encrypted channel.	a) Identifying all the security mechanisms being used in the communication process verification through: i. Testing, in presence of OEM team ii. Code review iii. Process audit of the key-life cycle process
29.	V10.5 Verify that the firmware apps pin the digital signature to a trusted server(s).	a) Obtain a list of trusted servers that the firmware is expected to interact with for digital signature verification. b) Examine the firmware's source code or binary to identify how it handles digital signatures and server communication. c) Look for mechanisms where the firmware checks for signatures against a predefined list of trusted servers. Check for hardcoded values related to server addresses and digital signatures.
30.	V12.1 Ensure that states, events, and network traffic of IoT devices and systems are monitored and logged.	a) Ensure that the logging settings are correctly configured for all devices and systems. b) Simulate various states and events on the IoT devices and observe if they are captured correctly by the monitoring tools. c) Generate and capture network traffic and verify that it is logged appropriately.
31.	V13.1 Validate that logs for IoT devices protected from leakage, destruction, and unintended alteration.	a) Ensure that access to logs is restricted to authorized personnel only through role-based access controls (RBAC). b) Confirm that log data is encrypted both in transit and at rest using industry-standard encryption protocols (e.g., TLS, AES). c) Verify that logs cannot be modified without proper authorization and that any changes are traceable.
32.	V13.2 Verify the presence of tamper resistance and/or tamper detection features.	a) Examine the physical construction of the device. Look for secure enclosures, tamper-evident seals, and screws.

		b) Simulate tampering attempts such as opening the device or disconnecting components. Observe if the device detects and logs these events. c) Verify the presence and effectiveness of alert mechanisms (e.g., alarms, notifications) triggered by tampering attempts.
33.	V15.1 Verify that IoT devices are delivered with secure settings and configurations.	a) Examine the documentation to verify that it includes secure default settings and recommendations for secure configurations. b) Assess the initial setup process for security best practices. c) Use automated tools to scan the device for common vulnerabilities related to default settings and configurations.
34.	V15.2 Ensure that only authorized entities can modify the configuration settings of the IoT device if they are modifiable.	a) Examine the device's access control lists or similar configurations to ensure that only authorized entities have modification rights.
35.	V15.3 Verify that IoT devices ensure that common values for critical security parameters, such as global private keys or standard passwords, are replaced by values that are unique per device or explicitly defined by an appropriate external entity before they are put into operation.	a) Obtain a sample of the device and perform a factory reset to revert it to its initial state. b) Check the initial values of critical security parameters (e.g., private keys and passwords) after the reset. c) Attempt to configure the device with new values and observe if the device enforces uniqueness or external definition for these parameters.
36.	V15.4 Verify security controls are in place to hinder firmware reverse engineering (e.g., removal of verbose debugging symbols).	a) Testing, in presence of OEM team, to verify the security controls as provided by the vendor to hinder firmware reverse engineering.
37.	V16.1 Confirm the implementation and application of authentication mechanisms for users and IoT devices accessing IoT systems and services.	a) Verify the presence of user authentication mechanisms (e.g., passwords, biometrics, multi-factor authentication) for accessing IoT systems and services. b) Confirm the use of device authentication mechanisms (e.g., certificates, pre-shared keys, unique identifiers) for IoT devices accessing the network.
38.	V16.2 Verify that IoT devices protect stored and transmitted data, including configuration	a) Verify documentation for details on data storage protection mechanisms. b) Assess the effectiveness of access control measures by attempting unauthorized access to stored data.

	settings, identifying data, user data, event logs, and sensitive security parameters, against unauthorized access, modification, and disclosure, while also safeguarding software from unauthorized access and modification, utilizing cryptography for data confidentiality and integrity.	c) Check the use of cryptographic checksums or hashes to verify software integrity. d) Evaluate the security of the software update process, including digital signature verification.
39.	V16.3 Verify that the application and firmware components are not susceptible to OS Command Injection by invoking shell command wrappers, scripts, or that security controls prevent OS Command Injection.	a) List all interfaces (e.g., web interfaces, APIs, command-line interfaces) that accept user inputs and interact with the operating system. b) Ensure that the application and firmware do not invoke shell command wrappers or scripts that could be exploited for OS Command Injection. c) Perform a thorough review of the source code, focusing on input validation and sanitization. Look for functions that execute OS commands, such as system(), exec(), popen(), and similar. d) Conduct penetration tests to simulate OS command injection attacks. Use tools like OWASP ZAP, Burp Suite, and Metasploit to identify vulnerabilities.
40.	V17.1 Ensure that the update procedure is defined and includes validation of updates, configuration choices for automatic/manual updates, scheduling options, and notification settings. The update should maintain the cryptographic chain of trust with the root of trust.	a) Verification shall be done as per the applicable scenario: i. Case 1: Automatic OTA updates are available: A standard operating procedure for issuing automatic updates/upgrades to the in-field devices is required to be submitted by the vendor which can then be evaluated by the evaluation agency. ii. Case 2: Automatic OTA updates are not available and vendor provides manual updates: A standard operating procedure for issuing manual updates/upgrades to the in-field devices is required to be submitted by the vendor which can then be evaluated by the evaluation agency. b) Confirm that the update process maintains the cryptographic chain of trust from the root of trust. c) Ensure that certificates used in the update process are validated against the root of trust.
41.	V17.2 Ensure that software updates for IoT devices are securely initiated by authorized entities and that interruptions during	a) Test the update initiation process by attempting to initiate updates with both authorized and unauthorized credentials. b) Simulate various interruption scenarios (e.g., power loss, network disconnection) during the update process.

	updates minimize potential harm.	c) Evaluate the device's ability to roll back to the previous stable state or resume the update process safely.
42.	V17.3 Verify that the firmware update process is not vulnerable to time-of-check vs time-of-use attacks.	a) Testing, in presence of OEM team, to verify the measures implemented in the device to make it resistant to time-of-check vs time-of-use attacks.
43.	V17.4 Verify the device uses code signing and validates firmware upgrade files before installing.	a) Testing, in presence of OEM team, to verify the following: i. Device gets successfully updated with the documented secure upgrade process when a valid update package is provided. ii. Device does not boot up when a tampered update package (like with missing signature, invalid signature) is provided.
44.	V17.5 Verify that the device cannot be downgraded to old versions (anti-rollback) of valid firmware.	a) Testing, in presence of OEM team, to verify that the device cannot be downgraded to old versions (anti-rollback) of valid firmware.
45.	V18.1 Ensure that vulnerabilities of IoT devices are actively monitored and reported to IoT users and relevant parties along with associated risks.	a) Obtain and review documentation describing the vulnerability monitoring system. b) Verify integration with external vulnerability databases and threat intelligence feeds. Simulate the identification of a vulnerability. c) Observe the reporting process and timing. d) Simulate risk assessment for an identified vulnerability. e) Evaluate the communication of risks to users.
46.	V19.1 Verify that the device wipes firmware and sensitive data upon detection of tampering or receipt of invalid message.	a) Review the device documentation and design specifications to identify the tamper detection methods implemented (e.g., physical tamper switches, sensors, or software-based detection). b) Perform physical and software tampering attempts to trigger the detection mechanisms. Observe and record the device's response.
47.	V20.1 Verify that IoT users are provided with guidance on the proper use of IoT devices, including risks and potential undesirable effects.	a) Verify the presence of user manuals, online help, and other resources that provide guidance on the proper use of IoT devices. b) Verify that the user guidance includes detailed information on the potential risks associated with the IoT device. c) Confirm that it addresses both security risks (e.g., unauthorized access, data breaches) and safety risks (e.g., physical harm, malfunction).
48.	V23.1 Ensure that the acquiring organization has a system in place to evaluate supplier security measures	a) Obtain and review the acquiring organization's policies, procedures, and governance framework related to supplier evaluation and security requirements. b) Design and architecture details till the PCBA and SoC level to be provided to aid in counterfeit mitigation and malware detection.

	according to local laws and regulations.	c) Verify if the organization has established policies that align with local laws and regulations governing supplier security measures, such as data protection laws. d) Assess if the organization has clearly defined security criteria that suppliers must meet, covering areas such as data protection, confidentiality, integrity, availability, and compliance with legal requirements.
49.	V23.2 Design and architecture details till the PCBA and SoC level to be provided to aid in counterfeit mitigation and malware detection.	a) Verify design documentation at the PCBA and SoC levels, ensuring availability and completeness. b) Cross-check documentation with actual components and assess traceability from manufacturer to device integration. c) Use X-ray imaging, visual inspection, and electrical testing to verify component authenticity, identify hardware vulnerabilities, and perform firmware analysis for anomalies.
50.	V23.3 Threat mitigation strategies for tainted and counterfeit products shall be implemented as part of product development.	a) Ensure that threat analysis is conducted to identify risks related to counterfeit and tainted products and assess the effectiveness of existing mitigation strategies. b) Evaluate the integration of threat mitigation strategies in the product development lifecycle, ensuring they include verification and validation processes for components and subsystems.
51.	V23.4 One or more up-to-date malware detection tools shall be deployed as part of the code acceptance and development processes. Malware detection techniques shall be used before final packaging and delivery (e.g., scanning finished products and components for malware using one or more up-to-date malware detection tools).	a) Ensure that malware detection tools are current and relevant, and integrated into code acceptance and development pipelines. b) Ensure that regular scanning of source code, binaries, and firmware are done and results are analyzed to confirm no malware presence. c) Ensure that final malware scans are performed before product packaging, review logs and reports to verify all components are found to be clean.
52.	V23.5 Supply chain risk identification, assessment, prioritization, and mitigation shall be conducted.	a) Ensure that a comprehensive supply chain risk analysis is conducted, including sourcing, transportation, and storage. b) Ensure that suppliers are engaged for transparency, their risk management practices are assessed, and risks are prioritized by severity and likelihood. c) Ensure that risk mitigation strategies are tailored, ongoing monitoring and auditing processes are implemented, and corrective actions are enforced as needed.

53.	V24.1 Ensure that documentation detailing IoT device security information is present and restrict disclosure solely to pertinent parties.	a) Ensure the existence and enforcement of policies governing the management of IoT device security documentation. b) Ensure that documentation access is restricted to authorized personnel only. c) Ensure that sensitive IoT device security information is encrypted both in transit and at rest to protect against unauthorized access.
54.	V28.1 Ensure that data and licensed software stored in IoT device are removed or securely overwritten prior to disposal or re-use.	a) Verify the existence of formal policies and procedures for data sanitization and device disposal. b) Confirm the use of approved data destruction techniques (e.g., cryptographic erasure, degaussing, physical destruction). c) Ensure that licensed software is removed or deactivated in compliance with software license agreements. d) Conduct tests to ensure the tools effectively remove data and software. e) Validate that no residual data or software remains on devices after sanitization.
55.	V28.2 Verify the IoT device has a secure function allowing only authorized entities to delete relevant user data stored on the device in any memory type.	a) Attempt to bypass authentication mechanisms using standard penetration testing techniques. b) Simulate role assignments and attempt to perform deletions from non-authorized roles. c) Perform controlled deletion operations and verify that only the targeted user data is deleted.
56.	V29.1 Audit the IoT device to confirm the incorporation of privacy-enhancing features.	a) Obtain a comprehensive list of privacy-enhancing features the device claims to support. b) Examine the source code for the implementation of privacy features. c) Verify data minimization practices and anonymization techniques..
57.	V30.1 Ensure that stakeholders of IoT device ensure strict privacy settings by default without requiring IoT user interaction or intervention.	a) Verify the user manual, technical documentation, and privacy policy of the IoT device. b) Assess the initial configuration process to ensure privacy settings are automatically applied.
58.	V31.1.1 Confirm that IoT users are provided with a privacy notice detailing the collection of personal data by IoT devices and the purpose of its use.	a) Verify if the privacy notice is easily accessible to users through device interfaces, websites, or mobile applications. b) Check if the privacy notice clearly explains the purposes for which data is collected, how it will be used, and whether it provides information on user rights regarding their data.
59.	V31.2.1 Verify that the consent to privacy notice is obtained from IoT users before data	a) Assess how consent is obtained when users first interact with the IoT device, including the methods used (e.g., checkboxes, explicit consent forms).

	collection by IoT device or changes in use.	b) Verify if the IoT Service Provider maintains records of consent obtained, updates made to consent preferences, and audit trails.
60.	V33.1 Validate that end users' privacy requirements and concerns are addressed in the design of IoT devices.	a) Ensure that all relevant privacy requirements and concerns of end users are identified and documented. b) Conduct functional testing to ensure privacy features (e.g., data encryption, access controls) work as intended.
61.	V34.1 Obtain a declaration from the IoT device developer confirming regular review of privacy controls' effectiveness and continuous identification of new privacy risks.	a) Ensure that the declaration includes: i) Frequency of privacy control reviews. ii) Processes for identifying new privacy risks. iii) Roles and responsibilities of personnel involved in these activities. iv) Any recent findings or updates made to privacy controls based on these reviews. b) Review historical records of privacy reviews and risk assessments conducted over a defined period (e.g., the past 2-3 years).
62.	V35.1.1 Ensure that unique cryptographic keys and certificates are assigned to each individual IoT device to enhance privacy and aid in identifying devices relevant to cyber incidents.	a) Identify all keys and certificates utilized within the device ecosystem and conduct verification through the following methods: i. Testing in the presence of the Original Equipment Manufacturer (OEM) team. ii. Code review. iii. Process audit of the key lifecycle management process.
63.	V35.2.1 Ensure a documented process exists to map device identifiers to specific individuals or user profiles for IoT devices. This mapping should be securely maintained and accessible solely by authorized IoT users.	a) Check if access to mapping data is based on documented policies that define who can access, modify, or delete mapping information. b) Verify if logging mechanisms are in place to track access to mapping data, detect anomalies, and generate audit trails. c) Assess the effectiveness of authentication methods (e.g., MFA) in ensuring that only authorized personnel can access mapping data.
64.	V36.1 Verify IoT devices enforce authorized access to interfaces with proper authentication and resist any attempts to bypass, tamper with, or falsify implemented authentication measures.	a) Check device specifications and documentation for implemented authentication methods b) Perform security testing to validate the strength of authentication controls.
65.	V37.1 Verify that IoT devices minimize the collection of indirect	a) Test scenarios to verify that IoT devices collect only necessary data for their intended operation.

	data (data collected without user participation) to only what is necessary for operation, unless explicit user consent is obtained.	b) Simulate user interactions to assess the effectiveness of consent prompts and user understanding. c) Evaluate how devices respond to user preferences and consent settings over time (e.g., honoring opt-out requests).
66.	V38.1 Validate that user preferences for privacy controls can only be added, modified, or deleted when the authorized user is authenticated to the IoT device.	a) Simulate various scenarios, such as correct authentication attempts, incorrect password entries, and session timeout handling. b) Verify that unauthorized users are unable to bypass authentication measures to gain access to sensitive privacy controls.
67.	V39.1 Ensure that there is a secondary, independent verification for automated decisions made by IoT devices that could cause irreversible harm to users.	a) Identify and list all automated decisions made by the IoT device that have the potential to cause irreversible harm to users. b) Simulate scenarios where automated decisions are critical and could potentially harm users or impact safety. c) Test the response of secondary verification systems to unexpected inputs, errors in primary decision-making systems, or deliberate attempts to bypass verification.
68.	V40.1 Review documentation to confirm the presence of an accountability framework that outlines data privacy responsibilities for the IoT device.	a) Verify the documentation covers the following key components of an accountability framework: i. Data Collection: Clear explanation of what data is collected by the IoT device. ii. Data Processing: Details on how the data is processed, including any transformations, aggregations, or analyses performed. iii. Data Storage: Information on where and how the data is stored, including the security measures in place. iv. Data Sharing: Policies regarding data sharing with third parties, including any conditions or restrictions. v. User Consent: Processes for obtaining user consent for data collection and processing. vi. User Rights: Description of user rights regarding their data, such as access, correction, deletion, and portability. vii. Accountability Measures: Outline of the responsibilities of different stakeholders (e.g., manufacturers, service providers, users) concerning data privacy and security. viii. Compliance: Information on how the device complies with relevant data protection regulations and standards.
69.	V41.1 Ensure that PII of the device owner is saved securely with	a) Verify that the PII is encrypted using industry-standard encryption algorithms (e.g., AES-256) both at rest and during transmission.

	proper access control in place.	b) Ensure proper key management practices are in place, including key generation, storage, rotation, and destruction. c) Review access control policies to ensure they limit access to PII based on the principle of least privilege. d) Evaluate the authentication mechanisms (e.g., passwords, multi-factor authentication) used to grant access to PII. e) Verify that access to PII is logged, including successful and unsuccessful access attempts. f) Verify that PII is securely deleted when no longer needed.
70.	V42.1 Ensure that PII protection measures related to privacy risk in IoT devices are appropriately managed and only disclosed to the parties that require them.	a) Audit the mechanism used to secure the details of PII protection measures within the IoT device and ensure the secure disclosure of these details to authorized parties.
71.	V46.1 Verify that all legal, statutory, regulatory, and contractual requirements relevant to IoT device security, along with the organization's approach to meet these requirements, are identified, documented, and regularly updated.	a) Request and review all relevant documentation, including but not limited to: i. Legal and regulatory requirement documents ii. Internal policies and procedures iii. Contracts and agreements b) Verify that there is a comprehensive list of all legal, statutory, regulatory, and contractual requirements relevant to IoT device security. c) Review the internal audit process to verify that it includes regular checks for compliance with documented requirements. d) Check if there are any discrepancies between documented policies and actual practices.

Annex B

Illustrative Mapping of IoT Device Security & Privacy Checkpoints to Assessment Levels

To ensure a comprehensive approach to security and privacy, organizations often categorize their measures into different levels, with each level representing a different degree of rigor and complexity. This document allows IoT users or service developers to conduct risk assessments and select the appropriate assurance level based on identified risks. The risks are identified in line with the intent of standards IS/ISO/IEC 27001, IS/ISO/IEC 27402 and OWASP ASVS 4.0.3. The levels of IoT Device Security & Privacy Assessment and Evaluation are structured across three assurance levels: Level 1, Level 2, and Level 3 in line with below mentioned descriptions:

Level 1: Basic Security and Privacy

At Level 1, the focus is on implementing fundamental security and privacy measures to provide a baseline level of protection for IoT devices and data. This level is suitable for simple IoT deployments and devices with limited capabilities.

Level 2: Enhanced Security and Privacy

Level 2 involves a more robust security and privacy approach, suitable for more complex IoT deployments and devices that handle sensitive data or operate in more challenging environments.

Level 3: Advanced Security and Privacy

Level 3 represents the highest level of security and privacy for IoT devices and systems. It is suitable for mission-critical applications, highly sensitive data, and deployments in high-risk environments.

The choice of security and privacy level depends on factors such as the IoT device's purpose, the data it handles, the potential impact of security breaches, and the regulatory environment. Organizations should conduct a thorough risk assessment to determine the appropriate level of security and privacy controls needed for their specific IoT deployments. In scenarios where risks differ significantly from those outlined in this document, the compliance assessments can be conducted at enhanced levels designated as L1+ or L2+ as mentioned below:

i. L1+: Additional security and privacy measures beyond Level 1.
ii. L2+: Enhanced requirements surpassing Level 2 standards.

Additionally, compliance with relevant industry standards and regulations, such as IT Act, Digital Data Protection Act, should also be considered when defining security and privacy requirements for IoT devices.

This annexure provides detailed verification points mapped to each assurance level (L1, L2, L3). These points serve as benchmarks for evaluating compliance with the specified security and privacy requirements. IoT stakeholders can ensure thorough evaluation and validation of device security and privacy measures according to the chosen assurance level.

The IoT device security and privacy checkpoints, extracted from IS/ISO/IEC 27400, IS/ISO/IEC 27402, and OWASP ASVS 4.0.3 Appendix C, are mapped to assessment levels as given in Table 7.

Table 7: Assessment levels

Sl. No.	Security & Privacy Checkpoint	L1	L2	L3	Reference Check point
1.	Ensure that a policy for IoT security is defined, approved by management, published, communicated to relevant personnel and relevant external parties and reviewed at planned intervals or if significant changes occur.			✓	V1.1
2.	Confirm that roles and responsibilities for IoT security are defined and allocated, with accountability clearly established.			✓	V2.1
3.	Confirm that the IoT device developer has identified all assets across the entire development process of the IoT device.			✓	V3.1
4.	Ensure that mechanisms are in place to apply knowledge gained from analyzing and resolving IoT device security incidents to reduce the likelihood or impact of future incidents.			✓	V6.1
5.	Verify that application layer debugging interfaces such USB, UART, and other serial variants are disabled or protected by a complex password.	✓	✓	✓	V7.1
6.	Verify that memory protection controls such as ASLR and DEP are enabled by the embedded/IoT operating system, if applicable.	✓	✓	✓	V7.2
7.	Verify that on-chip debugging interfaces such as JTAG or SWD are disabled or that available protection mechanism is enabled and configured appropriately.	✓	✓	✓	V7.3
8.	Verify that trusted execution is implemented and enabled, if available on the device SoC or CPU.	✓	✓	✓	V7.4
9.	Verify that sensitive data, private keys and certificates are stored securely in a Secure Element, TPM, TEE (Trusted Execution Environment), or protected using strong cryptography.	✓	✓	✓	V7.5
10.	Verify usage of cryptographically secure pseudo-random number generator on embedded device (e.g., using chip-provided random number generators).		✓	✓	V7.6
11.	Verify that sensitive traces are not exposed to outer layers of the printed circuit board.			✓	V7.7
12.	Verify that inter-chip communication is encrypted (e.g. Main board to daughter board communication).			✓	V7.8
13.	Verify the device uses code signing and validates code before execution.			✓	V7.9
14.	Verify that sensitive information maintained in memory is overwritten with zeros as soon as it is no longer required.			✓	V7.10

15.	Verify that the firmware apps utilize kernel containers for isolation between apps.			✓	V7.11
16.	Verify that secure compiler flags such as -fPIE, -fstack-protector-all, -Wl,-z,noexecstack, -Wl,-z, noexecheap are configured for firmware builds.			✓	V7.12
17.	Verify that micro controllers are configured with code protection.			✓	V7.13
18.	Verify that any use of banned C functions are replaced with the appropriate safe equivalent functions.	✓	✓	✓	V8.1
19.	Verify that each firmware maintains a software bill of materials cataloguing third-party components, versioning, and published vulnerabilities.	✓	✓	✓	V8.2
20.	Verify all code including third-party binaries, libraries, frameworks are reviewed for hardcoded credentials (backdoors).	✓	✓	✓	V8.3
21.	Verify that any available Intellectual Property protection technologies provided by the chip manufacturer are enabled.		✓	✓	V8.4
22.	Verify that only micro controllers that support disabling debugging interfaces (e.g. JTAG, SWD) are used.			✓	V8.5
23.	Verify that only micro controllers that provide substantial protection from de-capping and side channel attacks are used.			✓	V8.6
24.	Ensure the integration of security measures into IoT device development to maintain safety, including mechanisms to detect and halt erroneous or corrupted control data to prevent malfunctions.			✓	V9.1
25.	Verify that the firmware apps protect data-in-transit using transport layer security.	✓	✓	✓	V10.1
26.	Verify that the firmware apps validate the digital signature of server connections.	✓	✓	✓	V10.2
27.	Verify that wireless communications are mutually authenticated.	✓	✓	✓	V10.3
28.	Verify that wireless communications are sent over an encrypted channel.	✓	✓	✓	V10.4
29.	Verify that the firmware apps pin the digital signature to a trusted server(s).		✓	✓	V10.5
30.	Ensure that states, events, and network traffic of IoT devices and systems are monitored and logged.			✓	V12.1
31.	Validate that logs for IoT devices protected from leakage, destruction, and unintended alteration.			✓	V13.1
32.	Verify the presence of tamper resistance and/or tamper detection features.		✓	✓	V13.2
33.	Verify that IoT devices are delivered with secure settings and configurations.			✓	V15.1

34.	Ensure that only authorized entities can modify the configuration settings of the IoT device if they are modifiable.	✓	✓	✓	V15.2
35.	Verify that IoT devices ensure that common values for critical security parameters, such as global private keys or standard passwords, are replaced by values that are unique per device or explicitly defined by an appropriate external entity before they are put into operation.	✓	✓	✓	V15.3
36.	Verify security controls are in place to hinder firmware reverse engineering (e.g., removal of verbose debugging symbols).		✓	✓	V15.4
37.	Confirm the implementation and application of authentication mechanisms for users and IoT devices accessing IoT systems and services.			✓	V16.1
38.	Verify that IoT devices protect stored and transmitted data, including configuration settings, identifying data, user data, event logs, and sensitive security parameters, against unauthorized access, modification, and disclosure, while also safeguarding software from unauthorized access and modification, utilizing cryptography for data confidentiality and integrity.	✓	✓	✓	V16.2
39.	Verify that the application and firmware components are not susceptible to OS Command Injection by invoking shell command wrappers, scripts, or that security controls prevent OS Command Injection.	✓	✓	✓	V16.3
40.	Ensure that the update procedure is defined and includes validation of updates, configuration choices for automatic/manual updates, scheduling options, and notification settings. The update should maintain the cryptographic chain of trust with the root of trust.	✓	✓	✓	V17.1
41.	Ensure that software updates for IoT devices are securely initiated by authorized entities and that interruptions during updates minimize potential harm.	✓	✓	✓	V17.2
42.	Verify that the firmware update process is not vulnerable to time-of-check vs time-of-use attacks.		✓	✓	V17.3
43.	Verify the device uses code signing and validates firmware upgrade files before installing.		✓	✓	V17.4
44.	Verify that the device cannot be downgraded to old versions (anti-rollback) of valid firmware.		✓	✓	V17.5
45.	Ensure that vulnerabilities of IoT devices are actively monitored and reported to IoT users and relevant parties along with associated risks.			✓	V18.1
46.	Verify that the device wipes firmware and sensitive data upon detection of tampering or receipt of invalid message.			✓	V19.1

47.	Verify that IoT users are provided with guidance on the proper use of IoT devices, including risks and potential undesirable effects.			✓	V20.1
48.	Ensure that the acquiring organization has a system in place to evaluate supplier security measures according to local laws and regulations.	✓	✓	✓	V23.1
49.	Design and architecture details till the PCBA and SoC level to be provided to aid in counterfeit mitigation and malware detection.	✓	✓	✓	V23.2
50.	Threat mitigation strategies for tainted and counterfeit products shall be implemented as part of product development.	✓	✓	✓	V23.3
51.	One or more up-to-date malware detection tools shall be deployed as part of the code acceptance and development processes. Malware detection techniques shall be used before final packaging and delivery (e.g., scanning finished products and components for malware using one or more up-to-date malware detection tools).	✓	✓	✓	V23.4
52.	Supply chain risk identification, assessment, prioritization, and mitigation shall be conducted.	✓	✓	✓	V23.5
53.	Ensure that documentation detailing IoT device security information is present and restrict disclosure solely to pertinent parties.			✓	V24.1
54.	Ensure that data and licensed software stored in IoT device are removed or securely overwritten prior to disposal or re-use.			✓	V28.1
55.	Verify the IoT device has a secure function allowing only authorized entities to delete relevant user data stored on the device in any memory type.	✓	✓	✓	V28.2
56.	Audit the IoT device to confirm the incorporation of privacy-enhancing features.			✓	V29.1
57.	Ensure that stakeholders of IoT device ensure strict privacy settings by default without requiring IoT user interaction or intervention.			✓	V30.1
58.	Confirm that IoT users are provided with a privacy notice detailing the collection of personal data by IoT devices and the purpose of its use.			✓	V31.1.1
59.	Verify that the consent to privacy notice is obtained from IoT users before data collection by IoT device or changes in use.			✓	V31.2.1
60.	Validate that end users' privacy requirements and concerns are addressed in the design of IoT devices.			✓	V33.1
61.	Obtain a declaration from the IoT device developer confirming regular review of privacy controls' effectiveness and continuous identification of new privacy risks.	✓	✓	✓	V34.1

62.	Ensure that unique cryptographic keys and certificates are assigned to each individual IoT device to enhance privacy and aid in identifying devices relevant to cyber incidents.	✓	✓	✓	V35.1.1
63.	Ensure a documented process exists to map device identifiers to specific individuals or user profiles for IoT devices. This mapping should be securely maintained and accessible solely by authorized IoT users.			✓	V35.2.1
64.	Verify IoT devices enforce authorized access to interfaces with proper authentication and resist any attempts to bypass, tamper with, or falsify implemented authentication measures.	✓	✓	✓	V36.1
65.	Verify that IoT devices minimize the collection of indirect data (data collected without user participation) to only what is necessary for operation, unless explicit user consent is obtained.			✓	V37.1
66.	Validate that user preferences for privacy controls can only be added, modified, or deleted when the authorized user is authenticated to the IoT device.			✓	V38.1
67.	Ensure that there is a secondary, independent verification for automated decisions made by IoT devices that could cause irreversible harm to users.			✓	V39.1
68.	Review documentation to confirm the presence of an accountability framework that outlines data privacy responsibilities for the IoT device.			✓	V40.1
69.	Ensure that PII of the device owner is saved securely with proper access control in place.			✓	V41.1
70.	Ensure that PII protection measures related to privacy risk in IoT devices are appropriately managed and only disclosed to the parties that require them.			✓	V42.1
71.	Verify that all legal, statutory, regulatory, and contractual requirements relevant to IoT device security, along with the organization's approach to meet these requirements, are identified, documented, and regularly updated.	✓	✓	✓	V46.1