The validity of the "Certificate of Approval" will be issued for three years from date of issue.

Essential Security Requirements

Sr. No.	Category	Testing Parameter	What to be tested	Documents Required	Implementation Details	Comment by Developer Yes/No
1.	Hardware Level Security Parameter (supported by software)	1.1 Verify that application layer debugging interfaces such USB, UART, and other serial variants are disabled or protected by a complex password.	1. 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 2. 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 3. Testing, in presence of OEM team, to verify the enabling/disabling of all the ports	The vendor shall provide the following: a. Datasheet of the SoC being used in the device. b. Documentation related to ports/interfaces enabled in the production devices and the related access control mechanism for protection of the same. c. Process flow of the Manufacturing/ Provisioning of the device		

			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. 4. Process audit 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 sub components/peripherals.]			
		1.2 Verify that cryptographic keys and	Identifying all the keys and certificates being used in the device eco-system and	Vendor shall submit the following: 1. List of all keys		

		certificates are unique to each individual device.	verification through: • Testing, in presence of OEM team • Code review • Process audit of the key-life cycle process	and certificates being used in the device ecosystem 2. Key management life cycle (purpose, generation, storage, destruction/zeroization, validity, key changeover/rotation)		
		1.3 Verify that on-chip debugging interfaces such as JTAG or SWD are disabled or that available protection mechanism is enabled and configured appropriately.	1. 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 2. 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 3. Testing, in presence of OEM team, to verify the	The vendor shall provide the following: a. Datasheet of the SoC being used in the device. b. Documentation related to ports/interfaces enabled in the production devices and the related access control mechanism for protection of the same. c. Process flow of the Manufacturing/Provisioning of the device		

			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. 4. Process audit 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 sub components/peripherals.]			
		1.4 Verify that trusted execution is	Identifying whether TEE/SE/TPM is	The vendor shall provide the following:		

		implemented and enabled, if available on the device SoC or CPU.	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: **CASE 1: TEE/SE/TPM is not available:** No further assessment **CASE 2: TEE/SE/TPM is available and enabled:** Verification through code-review that crypto functions are called through TEE/SE/TPM APIs. **CASE 3: TEE/SE/TPM is available but not enabled by the vendor:** Termed as non-conformance to the requirement.	1. Datasheet of the SoC being used in the device. 2. User manual/ Technical specifications of the device 3. Code snippets of the TEE API call, wherever applicable		

			OEM is required to enable and implement the TEE/SE/TPM.			
		1.5 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.	Identifying all the keys and certificates being used in the device eco-system, sensitive data and their storage mechanism(s); and verification through: • Testing, in presence of OEM team • Code review • Process audit of the key-life cycle process	Vendor shall submit the following: 1. List of all keys and certificates being used in the device ecosystem 2. List of all the sensitive data with their intended usage and secure storage mechanism(s) as implemented along with secure configurations to be enabled in the device. 3. Key management life cycle (purpose, generation, storage, destruction/zeroization, validity, key changeover/rotation) private keys and certificates.		
		1.6Verify the presence of tamper resistance and/or tamper detection	Testing, in presence of OEM team, to verify the measures implemented in the device to prevent software and hardware	Vendor shall submit the following: 1. Measures available in the device to prevent software		

		features.	tampering.	tampering. 2. Measures available in the device to prevent hardware tampering.		
		1.7 Verify that any available Intellectual Property protection technologies provided by the chip manufacturer are enabled.	Testing, in presence of OEM team, to verify the enabling of the Intellectual Property protection technologies provided by the chip manufacturer, if available.	Vendor shall submit the following: 1. Datasheet of the SoC 2. Documentation regarding the Intellectual Property protection technologies provided by the chip manufacturer which have been enabled. 3. In case, no Intellectual Property protection technologies are being provided by the chip manufacturer, then a declaration stating the same.		
		1.8 Verify the device validates the boot image signature before loading.	Testing, in presence of OEM team, to verify the following: 1. Device boots up successfully with the documented	Vendor shall submit the following: 1. Datasheet of the SoC 2. Technical specifications of		

			secure boot process when a valid boot image is provided. 2. Device does not boot up when a tampered boot image (like with missing signature, invalid signature) is provided.	the device regarding secure boot (should consist of keys involved and their management life cycle*, signature validation process and any other secure mechanisms if implemented.)		
		1.9 Verify usage of cryptographically secure pseudo-random number generator on embedded device (e.g., using chip-provided random number generators).	Verification of the documentation provided by the vendor regarding the random number generators being used in the device. Verification through code-review that random number generators or related libraries as applicable are being used in the device.	Vendor shall submit the documentation regarding the random generators (either hardware based or software based or both) being used in the device with their intended usage. In case, hardware based random number generators are being used, vendors shall submit the following: 1. Datasheet of the SoC 2. Technical specifications of the device regarding random generators		

				In case, software based random number generators are being used, vendors shall provide the libraries being used for the same.		
2.	Software/Firmware	2.1 Verify that memory protection controls such as ASLR and DEP are enabled by the embedded/ IoT operating system, if applicable.	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.	Vendor shall submit the declaration of the memory protection controls available and enabled in the device.		
		2.2 Verify that the firmware apps protect data-in-transit using transport layer security.	1. Verifying that strong encryption algorithms and secure TLS version is supported by the device to establish secure communication. 2. Verifying that device properly validates the server's TLS certificate to ensure that it is trusted and has not been tampered with. 3. Testing for	The vendor shall submit the specifications and documentation related to the configurations available in the applications and firmware related to transport layer security.		

			vulnerabilities which can affect the security of TLS connection such as padding oracle attacks, or weak cipher suites. 4. Using tools such as Nmap to identify open ports through which device can be accessed leading to unintended data retrieval. 5. Verifying that theTLS session(s) are resistant to attemptsof interception and decryption of network traffic using man-in-the-middle attacks using tools like Burpsuite.			
		2.3 Verify that the firmware apps validate the digital signature of server connections .	1. Identifying the scenarios when the device establishes the server connections with the external world and verifying the following: • Security features,	Vendor mentioning the use-cases when the device establishes server connections with the external world, with detailed information about the security measures in place while validating the digital		

			related to secure server connections and digital signature validation as implemented like strong cipher suites, secure TLS version, SSL pinning etc. supported by code walkthrough. • Proper certificate validation, certificate chain validation and certificate revocation checks are implemented in the device. 2. Testing for vulnerabilities which can affect	signatures of the server connections.		

			the security of TLS connection such as padding oracle attacks, or weak cipher suites. 3. Using tools such as Nmap to identify open ports through which device can be accessed leading to unintended data retrieval. 4. Verifying that TLS session(s) are resistant to attemptsof interception and decryption of network traffic using man-in-the-middle attacks using tools like Burpsuite.			
		2.4 Verify that any use of banned C functions are replaced with the appropriate safe equivalent functions.	Secure code review [both automated and manual], in presence of OEM team, using a licensed static analysis tool through any of the following approaches: 1. Visit to the evaluation agency by the vendor	Vendor shall provide : 1. Firmware binaries for code review. 2. Internal code review reports		

			with the firmware code and installing the licensed static analysis tool available with the evaluation agency in their systems. [Recommended] 2. 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. 3. Giving a remote access of the systems at vendor site to the evaluation agency for installing their licensed static analysis tool available with them. 4. 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.			
		2.5 Verify that each firmware maintains a software bill of materials cataloging third party components, versioning, and published vulnerabilities.	Verification of the submitted list of third party components by running automated tools like FACT on the firmware. Identifying vulnerabilities in the third party component(s) through publically available vulnerability databases 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.	Vendor shall submit the following: 1. Documentation for information on software bill of materials, including third-party components and versions. 2. Organization process and policies for the following: • Addressing and patching any identified vulnerabilities in third-party components. • Informing the customers about the security issues or vulnerabilities and providing security		

				updates and patches for the same. 3. Configuration management system and related policies for maintaining firmware and third party binaries, libraries and frameworks along with the patches/fixes issued to the devices.		
		2.6 Verify all code including third-party binaries, libraries, frameworks are reviewed for hardcoded credentials (backdoors).	Independent secure code review [both automated and manual] using a licensed static analysis tool through any of the following approaches: 1. 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] 2. Visit to the	Vendor shall provide : 1. Firmware binaries for code review. 2. Internal code review reports		

			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. 3. Giving a remote access of the systems at vendor site to the evaluation agency for installing their licensed static analysis tool available with them. 4. 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.			
		2.7 Verify that the firmware apps pin the digital	1. Identifying the scenarios when the device establishes the server	Vendor shall submit a document mentioning the use-cases when the device establishes		

		signature to a trusted server(s).	connections with the external world and verifying the following: • Security features, related to secure server connections and digital signature validation as implemented like strong cipher suites, secure TLS version, SSL pinning etc. supported by code walkthrough. • Proper certificate validation, certificate chain validation and certificate revocation checks are	server connections with the external world, with detailed information about the security measures in place while validating the digital signatures of the server connections.		

			implemented in the device.			
		2.8 Verify security controls are in place to hinder firmware reverse engineering (e.g.removal of verbose debugging symbols).	Testing, in presence of OEM team, to verify the security controls as provided by the vendor to hinder firmware reverse engineering.	Vendor shall submit the documentation regarding the security controls in place to hinder firmware reverse engineering.		
		2.9 Verify that the firmware update process is not vulnerable to time-of-check vs time-of-use attacks.	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.	Vendor shall submit the measures implemented in the device to make it resistant to time-of-check vs. time-of-use attacks.		
		2.10 Verify the device uses code signing and validates firmware upgrade files before installing.	Testing, in presence of OEM team, to verify the following: 1. Device gets successfully updated with the documented secure upgrade process when a valid update package is provided. 2. Device does not boot up when a	Vendor shall submit the process of achieving secure firmware upgrade which should consist of keys involved and their management life cycle*, signature validation process and any other secure mechanisms if implemented.		

			tampered update package (like with missing signature, invalid signature) is provided.			
		2.11 Verify that the device cannot be downgraded to old versions (anti-rollback) of valid firmware.	Testing, in presence of OEM team, to verify that the device cannot be downgraded to old versions (anti-rollback) of valid firmware.	Vendor shall submit the process of achieving secure firmware upgrade which should consist of keys involved and their management life cycle*, signature validation process and any other secure mechanisms if implemented.		
		2.12 Verify that firmware can perform automatic firmware updates upon a predefined schedule.	Verification shall be done as per the applicable scenario: **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 as per C20, C21 and C22	Vendor shall provide the following: 1. Modes of updates available i.e. automatic, manual or both. 2. Organizational process and policies regarding the issuing of updates to the devices.		

			security requirement. **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 as per C20, C21 and C22 security requirement.			
3.	Secure Process Conformance	3.1 Verify that wireless communications are mutually authenticated.	Testing, in presence of OEM team, to verify the process of mutual authentication as laid down in the documentation by the vendor.	Vendors shall provide the documentation regarding the process of mutual authentication as implemented in the device when wireless communications are initiated. In case, the device does not support wireless communications, the vendor shall provide a declaration for the		

				same.		
		3.2 Verify that wireless communications are sent over an encrypted channel.	Identifying all the security mechanisms being used in the communication process verification through: • Testing, in presence of OEM team • Code review • Process audit of the key-life cycle process	Vendors shall provide the documentation regarding the security measures implemented in the device to prevent tampering of the data being sent through wireless mode of communication. In case, the device does not support wireless communications, the vendor shall provide a declaration for the same.		
		3.3 Verify that whether trusted sources are being used for sourcing the components of the device i.e. trusted supply chain through a managed Bill of materials for critical hardware component		Vendor shall submit Bill of materials for critical hardware components (related to security functions like SoC).		

		s (related to security functions like SoC) is in use.				
		3.4 Supply chain risk identification, assessment, prioritization, and mitigation shall be conducted. Supply chain risk/business continuity planning policy documents, playbooks reflecting how to handle supply chain disruption, post-incident summary documents need to be submitted and demonstrate the same.		Vendor shall submit the following: Supply chain risk identification, assessment, prioritization, and mitigation documents. Supply chain risk/business continuity planning policy documents, playbooks reflecting how to handle supply chain disruption, post-incident summary documents.		
		3.5 Verify the no proprietary network protocols		Document for Network protocols used in the device.		

		are being used in the device. If yes, then complete implementation details and the source code for the same shall be provided.				
4.	Security Conformance at product development stage	4.1 Design and architecture details till the PCBA and SoC level to be provided to aid in counterfeit mitigation and malware detection.		Design and architecture documents till the PCBA and SoC level.		
		4.2 Threat mitigation strategies for tainted and counterfeit products shall be implemented as part of product development.	Process and method artifacts need to be submitted and demonstrate the same.			
		4.3 One or more up-to-date malware	List of components that have been identified as			

		detection tools shall be deployed as part of the code acceptance and developme nt processes. Malware detection techniques shall be used before final packaging and delivery (e.g., scanning finished products and component s for malware using one or more up-to-date malware detection tools).	requiring tracking targets of tainting/counterf eiting, CM tool. Quality assurance process need to be submitted and demonstrate the same.			
		4.4 Supply chain risk identificatio n, assessment, prioritizatio n, and mitigation shall be		Supply chain risk/business continuity planning policy documents, playbooks reflecting how to handle supply chain disruption, post-incident		

		conducted.		summary documents need to be submitted and demonstrate the same.		