Cybersecurity Labelling Scheme

FOR MEDICAL DEVICES

BY CYBER SECURITY AGENCY OF SINGAPORE

Cybersecurity Labelling Scheme for Medical Devices [CLS(MD)] Publication No. 5

Minimum Test Specifications

October 2023
Version 0.5

FOREWORD

The Cybersecurity Labelling Scheme for Medical Devices [CLS(MD)] is part of efforts from the Ministry of Health (MOH), Cyber Security Agency of Singapore (CSA), Health Sciences Authority (HSA), and Synapxe to better secure Singapore's cyberspace and to raise cyber hygiene levels in medical devices.

Under the CLS(MD), the cybersecurity label for medical devices would provide an indication of the level of security in medical devices. It aims to improve security awareness by making such provisions more transparent to healthcare users and empowers them to make informed purchasing decisions for medical devices with better security using the information on the cybersecurity label.

The CLS(MD) seeks to incentivise manufacturers to develop and provide medical devices with enhanced cybersecurity provisions. The labels also serve to differentiate medical devices with better cybersecurity safeguards in the market, from their competitors.

At the same time, CSA intends to engage other like-minded partners for mutual recognition of the CLS(MD) with the objective of eliminating duplicated assessments across national boundaries.

The CLS(MD) is managed by the Cybersecurity Certification Centre (CCC) under the ambit of the Cyber Security Agency of Singapore (CSA). The CLS(MD) is jointly owned by MOH and CSA.

AMENDMENT RECORD

Version	Date	Author	Changes
0.5	October 2023	Cyber Security Agency of Singapore	Draft

CONTENTST

1 INTRODUCTION

1.0.1 This document provides the test specifications and methodology for the Cybersecurity Labelling Scheme for Medical Devices [CLS(MD)]. It outlines the set of minimum test cases to be performed by the testing laboratory (TL) under Levels 3 and 4.

1.0.2 The intended audience for this document is the manufacturers who are interested in getting their medical device labelled under the CLS(MD) and testing laboratories who are responsible for testing the devices under the CLS(MD) scheme.

1.0.3 The following roles are commonly referred in this document:

1. **Manufacturer** of the **Device Under Test (DUT)**
2. **Testing Laboratory (TL)** that performs the testing.
3. **Cybersecurity Certification Centre (CCC)** that oversees the CLS(MD) projects.

2 MINIMUM TEST SPECIFICATION

2.1 METHOD OF USE

2.1.1 The minimum test specification specifies test objectives that shall be met through the tests devised by the Testing Laboratory (TL) to assert that the Device Under Test (DUT) is reasonably resistant to basic attacks as part of testing under Levels 3 and 4. Details of the testing required under Levels 3 and 4 can be found in CLS(MD) Publication #2 – Scheme Specifications [1].

2.1.2 The minimum test specification is applicable to medical devices permissible for labelling under the CLS(MD), as indicated in CLS(MD) Publication #1 – Overview [2].

2.1.3 The minimum test specification neither explicitly specify nor restrict the methods, tools, or tests that the testing laboratory may use to meet the test objective. It is up to the testing laboratory's tools and expertise to achieve the test objective and validate that the DUT is indeed conformant to the requirements. However, some tools are suggested within this document for reference.

2.1.4 The minimum test specification spans over the following attack vectors:

1. Ports and Services
2. Firmware
3. Firmware Updates
4. Communications
5. Configuration Portal
6. Mobile Application
7. Authentication
8. Other Attacks

2.1.5 Alongside testing, the TL is also required to verify that the manufacturer's declaration of conformity (conformity checklist) is indeed being implemented. If the TL identifies discrepancies between the manufacturer's declaration of conformity and device implementation, the TL shall attempt to seek clarifications with the manufacturer and provide such information to the CCC.

2.2 PORTS AND SERVICES

2.2.1 Ports are essential to deliver network services. However, vulnerable implementations may be subjected to exploitation, and the opening of more than the required basic ports further amplifies this risk. With the use of a network port scanner or equivalent tools, the testing laboratory shall identify the list of open ports and services available on the DUT.

2.2.2 The testing laboratory shall make use of the manufacturer's conformity checklist as reference information for the following tests in this section.

2.2.3 All open ports and services are to be further investigated. Unnecessary ports and services shall be reported to CCC. Unnecessary ports and services are defined as those not necessary for the basic functioning of the device.

2.2.4 It is expected that the device may require certain ports and services to be open for operational purposes. This can be allowed if there is reasonable justification for doing so. However, if the testing laboratory is able to exploit the DUT through the open ports and services, then the device is deemed to have failed.

2.2.5 If there are any disparities between the testing laboratory's findings and the manufacturer's conformity checklist, the testing laboratory is to seek clarifications with the manufacturer on all detected open ports or services to confirm their functions and purposes.

No.	Test Objective	Remarks
1	To ensure that the device does not have unnecessary or potentially vulnerable open ports and services over its network interfaces.	This test involves multiple discovery scans conducted over each of the available network interfaces (e.g., WLAN, LAN, WAN, etc.). A device can provide different ports and services over different network interfaces. In particular, the following ports and services must not be available: 1. Telnet over port 23 2. Secure Shell (SSH) over port 22 3. Remote management Tools: NMAP, Zenmap, Nessus, Hydra, Nexpose NMAP Command: nmap -sU -sT -p0-65535 <IP address>

		If the testing laboratory has access to the device's command line, the netstat network utility can also be used for this test's purpose. Should the availability of the Telnet and SSH services be configurable via the device's configuration portal, further attempts to investigate these services shall be made. The lab is to enable these services and to connect to these services. If these services request for user credentials, the lab shall attempt to perform a brute-force attack on the login credentials using Hydra or other similar tools (e.g., John the Ripper). It may be possible to discover the required credentials using tools such as the 'creds' module in RouterSploit, related public discussion boards and forums for networking devices, or through related information attained through Open-Source Intelligence (OSINT).
2	To ensure that the device does not suffer from known exploits that can be conducted using typical vulnerability scanning and exploitation tools.	This test involves using various popular vulnerability-scanning tools such as (e.g., Metasploit, Immunity Canvas, etc.) to scan the device for vulnerabilities.
3	To ensure that the device does not enable Remote Administration by default.	This test involves examining the device for open ports/services for the purpose of remote administration. The test laboratory shall also examine the configuration of the device to ensure that Remote Administration is disabled.

2.3 FIRMWARE

2.3.1 The testing laboratory shall attempt to retrieve and analyse the firmware of the DUT. The testing laboratory may retrieve the firmware via available hardware debugging ports on the DUT, or by downloading the firmware from the manufacturer's webpage, or through other means such as dumping the firmware root access via Telnet/SSH, or removing the flash memory and downloading the firmware using flash reading tools.

2.3.2 In addition, the manufacturer shall provide the firmware to the testing laboratory and this provision should be documented. The TL shall also verify that the provided firmware is of the same version as what is stated in the application, by means of verifying the hash (SHA-256) or checksum value.

No.	Test Objectives	Remarks
1	To ensure that the device (including the manufacturer's website) shall not allow an attacker to retrieve sensitive credentials and contents from its firmware (i.e., secure storage).	This test involves the examination of the contents of the firmware for sensitive files, configuration files, password files. The firmware can be retrieved either via physical attacks (hardware debug ports, extracting firmware from the NAND/NOR), or via logical attacks (gaining root access via Telnet/SSH/command injection attacks on the device's configuration portal). Examples of sensitive materials not limited to the following: - Universal manufacturer wide default accounts and password materials - Cryptographic key materials - Login credentials - Login credentials to back-end servers. For root credentials obtained, the testing laboratory shall determine whether this credential is valid only for a specific unit or valid across all units of the same device model. Tools: Binwalk, Binary Ninja
2	To ensure that the device does	Hidden or undocumented

	not have hidden accounts that are undocumented to the end-user.	account could include user, device management, and service accounts. For such accounts, the permissions or privileges shall be reported in the test report and provided to CCC. Password cracking may be performed as appropriate.
3	To ensure that software services should run with the least privileges necessary.	This can be done by checking the permissions of running processes after initialization.

2.4 FIRMWARE UPDATES

2.4.1 The testing laboratory shall investigate if the firmware update process of the device is secure.

2.4.2 Firmware updates are typically provided either as a full binary firmware package or as a smaller binary package containing only updated portions of the code.

2.4.3 Firmware updates are typically performed over the following methods:

- Manual update
- Automated update

No.	Test Objectives	Remarks
1	**Secure Firmware Transmission**: To ensure that the device retrieves a firmware update securely.	This test involves checking that the device retrieves a firmware update via HTTPS, and that HTTPS is securely configured.
2	**Firmware Downgrade**: To ensure that the device does not allow an attacker to downgrade the firmware.	This test involves intentionally uploading a lower version of the firmware to check if the device rejects a lower version firmware. The manufacturer is to provide a lower version firmware to facilitate testing.
3	**Unsigned Firmware**: To ensure that the device does not install an unsigned firmware.	This test involves intentionally uploading an unsigned firmware to check if the device rejects an unsigned firmware. The manufacturer is to provide an unsigned firmware package for testing.
4	**Tampered and Illegitimate Firmware Update**: To ensure that the device does not accept a tampered firmware update package from an untrusted source.	This test is applicable if the device offers an avenue for the user to upload a firmware update package manually to the device. The firmware is typically downloaded from the manufacturer's portal. The test involves uploading an illegitimate and tampered firmware update package and checking if the device rejects the update. While the manufacturer may offer advanced/expert users the ability to load custom firmware, this function should not be the

		default configuration and users should be explicitly notified that the loading of customised firmware is not recommended and users who wish to proceed would need to accept the associated security risks. Example scenarios: 1. Incorrect signatures (e.g., one bit of the signature is changed) 2. Valid signature tested with misconfigured device (e.g., device date is set to a value outside the validity of the public key) 3. The device only checks that the signature field is filled but does not verify the signature.
5	**Unencrypted Firmware:** To ensure that the firmware binary file is encrypted if it is available for download on the manufacturer's web portal.	The testing laboratory shall confirm if the firmware update file is indeed encrypted and not compressed. Should encryption be used, the testing laboratory shall examine whether the encryption key is retrievable.

2.5 COMMUNICATIONS

2.5.1 The testing laboratory shall investigate if the device is susceptible to the following attacks.

No.	Test Objectives	Remarks
1	Default communication settings should be secured. For example, wireless devices should employ minimally WPA2-PSK-AES-CGM on its wireless interface.	For example, the use of unsecure protocols such as WEP or WPA should not be allowed. Disallowing the use of WEP or WPA prevents simple brute force attack against the WEP or WPA password.
2	To ensure that the device communicates in a secure manner with associated cloud services over the internet, local network, peer devices, configuration portal, and the companion mobile application.	If the device does not secure communications over the internet or local network, peer devices, on its configuration portal, or the companion mobile application, it may be possible for an attacker to conduct a man-in-the-middle attack and sniff critical user credentials. The testing laboratory shall test that the device is protected against man-in-the-middle attacks, version downgrades and negotiations to use weak cipher schemes. Tools: Wireshark, tcpdump, testssl.sh, SSLstrip
3	To ensure that the communication protocol implementation is not vulnerable to common attacks. Examples for TLS: Heartbleed, POODLE, etc. Examples for Bluetooth: SweynTooth, etc	For Bluetooth protocol, the testing laboratory shall test for replay attacks and other attacks that could lead to revealing users' device information and potentially personal data. For Zigbee, the test laboratory shall test whether the ZigBee implementation is vulnerable such that it is possible for an attacker to

		join the local network by exploiting known vulnerabilities (e.g., CVE-2020-6007). Tools: testssl.sh, hci tool, Gatttool, Zbwireshark, KillerBee, Zbreplay, zbassocflood, etc
4	To ensure that the device does not collect and send network statistic or telemetry data back to the manufacturer by default.	For most devices, this function is configurable via the configuration portal. However, even if disabled, the testing laboratory shall attempt to monitor outgoing traffic coming from the device to ensure that the device is indeed conformant and not sending such data. If data is still being sent, the testing laboratory shall record the destination IP address(s) and where possible the type of data being sent. If the device supports sending of network statistic or telemetry data back to the manufacturer, the data shall be protected prior to sending. Tool: Wireshark
5	To ensure that the pairing process of the device is secure.	For devices which may requiring pairing to a hub or with peer devices, the TL shall determine if the pairing process employed is secure.

2.6 CONFIGURATION PORTAL

2.6.1 The testing laboratory shall investigate if the device's configuration portal is secure. Majority of the configuration portals are typically accessed by means of webpage or via a mobile application. For web configuration portals, standard tests such as directory traversal, cross-site scripting, cross site request forgery, etc. would apply. This is especially so, if the web configuration portal is made available remotely (i.e., over Internet). The testing laboratory should consider other suitable web application penetration testing during the freeform testing phase.

2.6.2 For mobile applications, Chapter 2.7 - Mobile Application relating to companion mobile apps would apply.

No.	Test Objectives	Remarks
1	To ensure that the device installation/maintenance follows security best practices on usability.	For example, the installation procedures to setup the device should present secure-by-default options already turned on.
2	To ensure that the device does not have hidden URLs (configuration pages, firmware update pages, URLs that can be used to enable telnet or other services, etc.).	The testing laboratory is to conduct a brute-force attack on the configuration portal to determine if unknown/hidden URLs exists. Tools: OWASP ZAP, Dirbuster, Dotdotpwn
3	To ensure that the device does not allow unauthenticated users to configure the device or to access the configuration portal.	The device shall allow only authenticated users to access the configuration portal and to make changes (settings, firmware update, etc.). The device should authenticate the administrator and such authentication should not be bypassable. E.g., brute force.
4	Where databases are involved, to ensure it is not vulnerable to SQL injection vulnerabilities.	The device shall validate or sanitise inputs or implement other mitigation measures such as prepared statements to prevent SQL injection vulnerabilities. Tool: SQLMap
5	To ensure that the device's configuration portal is not susceptible to command injection attacks.	The device shall perform input validation to prevent command injection attacks. Tool: Burpsuite, Commix

6	To ensure that the device is protected against session hijacking attacks.	The device shall implement minimum session time outs and cross-site-request-forgery (CSRF) tokens for its configuration portal.
7	To ensure that the device is protected against cross-site scripting attacks.	Tool: Burpsuite, Wfuzz, XSStrike

2.7 MOBILE APPLICATION

2.7.1 The testing laboratory shall investigate the following through:

No.	Test Objectives	Remarks
1	To ensure that the app does not communicate in an unsecure manner.	Using Wireshark or other similar tools, the network traffic from the companion application shall be inspected for the use of HTTPS. The testing laboratory shall examine the TLS version and permitted cipher suites used. Tool: Wireshark, testssl.sh
2	To check that the app employs SSL pinning.	There are two ways to bypass SSL. 1. Adding a custom Certificate Authority to the User Certificate Store (e.g., using BurpSuite proxy). 2. Instrumentation attack (Frida Hook). This typically required a rooted or jailbroken phone. By ensuring that the app utilises SSL pinning, the first method of using custom Certificate Authority certificates is prevented. This makes it slightly more difficult for an attacker to perform a man-in-the-middle attack on the SSL connection. An attacker can conduct an SSL-bypass attack on the app to conduct research on communications between the app-server-device. By monitoring the requests between the mobile app client and backend, an attacker can easily map available server-side APIs and gain insight into the communication protocol, and also replay and manipulate requests to test for server-side vulnerabilities.
3	To ensure that the app does not store sensitive credentials in an unsecure manner.	For Android applications, the application should preferably store credentials using the Android KeyStore system as the bare

		minimum. For iOS, the application should preferably make use of the Apple Keychain services API.
4	To ensure that the mobile app is only communicating to legitimate URLS.	The objective of the check is to ascertain that the device/app is not communicating with suspicious servers/services. This would require manufacturers to submit a list of servers/services that the device makes use of, and the test laboratory is to verify that indeed the device/app is not communicating with any other URLs outside of the list.
5	To ensure that the app does not contain hard-coded sensitive materials (private keys, passwords, etc.)	-
6	To check that the logs do not contain sensitive information.	-

2.8 AUTHENTICATION

2.8.1 The following examples, not limiting to the following, are applicable to all passwords available on the device:

- Wi-Fi passwords, configuration portal passwords, user and service account passwords, etc.

2.8.2 The TL shall also make use of the manufacturer's checklist (particularly the manufacturer's supporting evidence for provision 5.1 – No universal default passwords) to ensure that the device has indeed implemented all security mechanisms and policies as claimed in the checklist.

No.	Test Objectives	Remarks
1	To ensure that the device is not susceptible to a brute-force attack on its login function.	The testing laboratory is to examine the device's login functions for the possibility of a brute-force attack. This test is applicable on all login functions of the device (configuration portal, companion mobile application, etc.). The TL shall verify that the device has indeed implemented all authentication rate limiting mechanisms as described in their manufacturer checklist, and that they are adequate and suitable to make brute-force attacks impracticable.
2	If the device comes with a pre-installed password, ensure that the device's pre-installed password is unique per device, and that the password does not appear in breach corpuses[1].	The pre-installed passwords of several units shall be compared to ensure that each of them is unique and sufficiently randomised. The passwords should not appear to be easily guessable.
3	Ensure that the device does not default to a common password upon factory reset.	The tester shall perform a factory reset on two units. If the passwords of the two units are of the same value, then the device shall be deemed as failed.

[1] Reference list of Common passwords are available at:

https://github.com/danielmiessler/SecLists/tree/master/Passwords,

2.9 OTHER ATTACKS

2.9.1 Listed below are other attack vectors that must considered by the TL.

No.	Test	Remarks
1	Physical attacks	Physical attack could be considered for applicable product categories, especially if it is non-damaging. Invasive/damaging physical attacks could be considered (e.g., accessing debug interfaces or even physical memory extraction) if it allows the retrieval of a universal secret.
2	Side channel analysis and fault injection	Simple side channel analysis and fault injection could be considered if allows the retrieval of a universal secret.
3	To ensure that the device does not have hardware ports such as JTAG or UART.	This test involves opening the device to examine for the presence of JTAG or UART ports on the PCB and to see if the JTAG/UART ports can be used to retrieve critical credentials or attain super user access. Both sides of the device board shall be examined. In the presence of JTAG or UART ports on the PCB, the lab shall minimally check if they are disabled by attempting to connect to them. If these ports are enabled, the lab shall determine if it is possible to achieve a connection. On Linux-based firmware, the lab shall see if it is possible to boot into Single User Mode over the interface. For example, an attacker may be able to retrieve critical credentials (e.g., root administrator or super user access) via means of JTAG/UART access, or by manipulating the bootloader to boot the device in Single User mode.

		This vulnerability is severely critical if the recovered credentials are applicable to other units of the same model or even other models/products of the same brand.
4	To ensure that the device does not have unnecessary exposed physical interfaces.	This test seeks to ensure that the device do not have unnecessary exposed physical interfaces that would present an additional attack interface. For example, if the USB port is only used for powering up the device, then the data pins of the USB port shall be disconnected.
5	To ensure that the device limits the number of allowed USB device classes to be connected on its USB interfaces.	This test seeks to ensure that the device limits the USB classes to the required minimum (for operation) to restrict the attack surface. Tools: USB emulator (e.g., FaceDancer)

3 REFERENCES

[1] Cyber Security Agency of Singapore, “CLS(MD) Publication #2 - Scheme Specifications,” Version 0.5, October 2023.

[2] Cyber Security Agency of Singapore, “CLS(MD) Publication #1 - Overview of CLS(MD),” Version 0.5, October 2023.

4 ACRONYMS

The following acronyms are used in CLS(MD) Publication 1, 2 and this document:

CCC	Cybersecurity Certification Centre
CSA	Cyber Security Agency of Singapore
DUT	Device Under Test
HPL	Historical Product List
LPL	Labelled Product List
TL	Testing Laboratory