FDA Cybersecurity in Medical Devices guidance 2023

WHITEPAPER

By Wessel Kooyman
Senior Consultant
Sunbird Medical Devices
wessel@sunbirdmedicaldevices.com

Sunbird
MD | MEDICAL DEVICES

Table of Contents

Introduction 5
Definitions 6
References 7
General principles 8
Cybersecurity is Part of Device Safety and the Quality System Regulations 8
Designing for Security 8
Transparency 8
Submission Documentation 8
Using an SPDF to Manage Cybersecurity Risks 10
Security Risk Management 11
Threat modeling 12
Third-Party Software Components 13
Cybersecurity Transparency 16
Labeling recommendations for devices with cybersecurity risks 17
Vulnerability Management Plans 19
Appendix 1: Security control categories and associated recommendation 20
Authentication 21
Authorization 23
Cryptography 24
Code, Data, and Execution Integrity 25
Confidentiality 27
Event Detection and Logging 28
Resiliency and Recovery 30
Firmware and Software Updates 31
Appendix 2: Submission Documentation for Security Architecture Flows 32
Call-flow diagrams 33
Information Details for an Architecture View 34
Appendix 3: Submission Documentation for Investigational Device Exemptions 38

Introduction

This whitepaper describes in detail the new cybersecurity requirements by the FDA. These leapfrog the previous version and will require significant work by medical device manufacturers.

Source: "Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions"
https://www.fda.gov/regulatory-information/search-fda-guidance- documents/cybersecurity-medical-devices-quality-system-considerations-and- content-premarket-submissions

Definitions

SiMD = Software in a medical device (software on hardware / firmware)

SaMD = Software as a medical device (pure software product)

SPDF = Secure Product Development Framework

TPLC = total product life cycle

References

Cybersecurity for Networked Medical Devices Containing Off-the-Shelf (OTS) Software Guidance 2005

https://www.fda.gov/regulatory-information/search-fda-guidance-documents/cybersecurity-networked-medical-devices-containing-shelf-ots-software

Off-The-Shelf Software Use in Medical Devices

https://www.fda.gov/regulatory-information/search-fda-guidance-documents/shelf- software-use-medical-devices

Postmarket Management of Cybersecurity in Medical Devices

https://www.fda.gov/regulatory-information/search-fda-guidance-documents/cybersecurity-networked-medical-devices-containing-shelf-ots-software

Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices

https://www.fda.gov/regulatory-information/search-fda-guidance- documents/guidance-content-premarket-submissions-software-contained-medical- devices
AAMI TIR57: 2016/(R)2019 Principles for medical device security—Risk management
https://www.aami.org/detail-pages/product/aami-tir572016-r-2019-pdf- a152e000006j60wq

Multiple Function Device Products: Policy and Considerations

https://www.fda.gov/regulatory-information/search-fda-guidance-documents/multiple- function-device-products-policy-and-considerations

Requests for Feedback and Meetings for Medical Device Submissions: The Q-submission Program

https://www.fda.gov/regulatory-information/search-fda-guidance- documents/requests-feedback-and-meetings-medical-device-submissions-q- submission-program

General Principles

Cybersecurity is Part of Device Safety and the Quality System Regulations

Make security part of your QMS. In SOP's, include security requirements, risk management, and validation. An SPDF may be a way to do this.

Designing for Security

Reach security objectives like integrity, authorization, availability, confidentiality and secure updatability must be considering during design.

These objectives vary based on intended use, environment, interfaces, vulnerabilities and risk of patient harm due to vulnerability exploitation.

Transparency

Users of the device must know and understand the security context and risks.

Examples are undisclosed vulnerabilities, lack of user manual instructions for updating and configuring the device, undisclosed third-party software or API usages.

Submission Documentation

Should be proportional to the cybersecurity risks. When the product evolves, the documentation might need to evolve as well. In the 510(k) context, FDA uses equivalency to determine safety (see 21 CFR 807.100(b)(2)(ii)(B))

Using an SPDF to Manage Cybersecurity Risks

- NIST Cybersecurity Framework
- Medical Device and Health IT Joint Security Plan

Security Risk Management

You have to look at the larger context of the device and where it's used (its environment).

Not the same as ISO 14971 risk management, because that's focused on safety. There are different harms, for example, such as business and reputational risk.

In addition, the risk probabilities are not as probabilistic as in safety risks that use historical data or modelling.

Security risk assessment focuses on exploitability. In post-market products (released software) the FDA's Postmarket Cybersecurity Guidance can be used. But some of those risks do not apply to pre-market software. In these instances, a premarket exploitability assessment could either assume a worst-case assessment and implement appropriate controls, or provide a justification for a reasonable exploitability assessment of the risk throughout the total product lifecycle and how the risk is controlled.

AAMI TIR57:2016 details how the security and safety risk management processes should interface to ensure all risks are adequately assessed.

- Known vulnerabilities should be mitigated in the design of the device
- If comprehensive design mitigations are not possible, compensating controls should be considered
- When any known vulnerabilities are only partially mitigated or unmitigated by the device design, they should be assessed as reasonably foreseeable risks in the risk assessment and be assessed for additional control measures or risk transfer to the user/operator, or, if necessary, the patient
- Risk transfer, if appropriate, should only occur when all relevant risk information is known, assessed, and appropriately communicated to users and includes risks inherited from the supply chain as well as how risk transfer will be handled when the device/system reaches end of support and end of life and whether or how the user is able to take on that role (e.g., if the user may be a patient)

Threat modeling

It is recommended that pre-market submissions include a threat model section. There are multiple methodologies for doing this and the rationale for using this one(s) should be included.

Threat modelling can be done during design reviews, for instance.

- identify system risks and mitigations as well as inform the pre- and post-mitigation risks considered as part of the security risk assessment

- state any assumptions about the system or environment of use (e.g. hospital networks are inherently hostile, therefore manufacturers are recommended to assume that an adversary controls the network with the ability to alter, drop, and replay packets)

- capture cybersecurity risks introduced through the supply chain, manufacturing, deployment, interoperation with other devices, maintenance/ update activities, and decommission activities that might otherwise be overlooked in a traditional safety risk assessment processes.

Third-Party Software Components

See FDA documents “Off-The-Shelf (OTS) Software Use in Medical Devices” and “Cybersecurity for Networked Medical Devices Containing Off-the-Shelf (OTS) Software”.

Software components, both commercial as well as open source are common in medical device software. The risk of each component must be assessed and addressed. You must document all software components and mitigate the risks associated with them. Software suppliers must also comply with this. The information must be recorded in the Design History File and the Design Master Record.

As part of configuration management, device manufacturers should have custodial control of source code through source code escrow and source code backups. While source code is not provided in premarket submissions, if this control is not available based on the terms in supplier agreements, the manufacturer should include in premarket submissions a plan of how the third-party software component could be updated or replaced should support for the software end. The device manufacturer is also expected to provide to users whatever information is necessary to allow users to manage risks associated with the device.

Software bill of materials

SBOM includes both the device manufacturer-developed components and third- party components (including purchased/licensed software and open-source software), and the upstream software dependencies that are required/depended upon by proprietary, purchased/licensed, and open-source software.

Documentation supporting SBOM

Each SBOM item should have:

- The asset(s) where the software component resides
- The software component name
- The software component version
- The software component manufacturer
- The software level of support provided through monitoring and maintenance from the software component manufacturer
- The software component's end-of-support date
- Any known vulnerabilities

As part of the premarket submission, manufacturers should also describe how the known vulnerabilities (item above) were discovered to demonstrate whether the assessment methods were sufficiently robust. For third-party components with known vulnerabilities, device manufacturers should provide in premarket submissions:

- A safety and security risk assessment of each known vulnerability

- Details of applicable safety and security risk controls to address the vulnerability. If risk controls include compensating controls, those should be described in an appropriate level of detail

Security Assessment of Unresolved Anomalies

FDA recommends submitting a list of unresolved bugs, including an assessment of the impact on safety and effectiveness. Additional documentation may be recommended by the Premarket Software Guidance.

Bugs with security implications should be considered vulnerabilities. A risk assessment according to 21 CFR Part 820.30(g) should also include Common Weakness Enumeration (CWE) categories.
See https://cwe.mitre.org for more info.

The criteria and rationales for addressing the resulting anomalies with security impacts should be provided as part of the security risk assessment documentation in the premarket submission.

Security Risk Management Documentation

Premarket submissions should include outputs of the security risk management processes, including a security risk management plan and security risk management report, as described in AAMI TIR57, including threat modeling, SBOM, and unresolved anomalies.

The security risk management report should:

- summarize the risk evaluation methods and processes, detail the security risk assessment, and detail the risk mitigation activities undertaken as part of a manufacturer's risk management processes

- provide traceability between the security risks, controls and the testing reports that ensure the device is reasonably secure

TPLC Security Risk Management

Cybersecurity risks should be constantly re-evaluated in the entire product cycle, including when new threat information becomes available during development and after release. A CAPA can be raised when new security threats emerge.

Fielded devices (on the market but no longer sold) need to be included, as their software can get outdated which will change their risk profiles.

The FDA recommends tracking and reporting these metrics:

- Percentage of identified vulnerabilities that are updated or patched (defect density)

- Time from vulnerability identification to when it is updated or patched

- ime from when an update or patch is available to complete implementation in devices deployed in the field

Averages of the above measures should be provided if multiple vulnerabilities are identified and addressed. These averages may be provided over multiple time frames based on volume or in response to process or procedure changes to increase efficiencies of these measures over time.

Cybersecurity Transparency

Transparency about cybersecurity is essential for users of the medical device. This can be achieved by labeling and a vulnerability management.

Labeling recommendations for devices with cybersecurity risks

For devices with cybersecurity risks, informing users of relevant security information may be an effective way to comply with labeling requirements relating to such risks. FDA also believes that informing users of security information through labeling may be an important part of QSR design controls to help mitigate cybersecurity risks and help ensure the continued safety and effectiveness of the device. Therefore, when drafting labeling for inclusion in a premarket submission, a manufacturer should consider all applicable labeling requirements and how informing users through labeling may be an effective way to manage cybersecurity risks and/or to ensure the safe and effective use of the device. Any risks transferred to the user should be detailed and considered for inclusion as tasks during usability testing (e.g., human factors testing) to ensure that the type of user has the capability to take appropriate actions to manage those risks.

The FDA recommends labeling to include:

- Device instructions and product specifications related to recommended cybersecurity controls appropriate for the intended use environment (e.g., anti-malware software, use of a firewall, password requirements)

- A list of network ports and other interfaces that are expected to receive and/or send data. This list should include a description of port functionality and indicate whether the ports are incoming, outgoing, or both, along with approved destination end-points

- Specific guidance to users regarding supporting infrastructure requirements so that the device can operate as intended (e.g., minimum networking requirements, supported encryption interfaces)

- An SBOM, in a machine readable format, preferably online and kept up-to-date constantly

- A description of where and how updates to the official software are made available

- A description of how the design enables the device to respond when anomalous conditions are detected (i.e., security events) in order to maintain safety and effectiveness. This should include notification to the user and logging of relevant information. Security event types could be configuration changes, network anomalies, login attempts, or anomalous traffic (e.g., send requests to unknown entities)

- A high-level description of the device features that protect critical functionality (e.g., backup mode, disabling ports/communications, etc.)

- A description of backup and restore features and procedures to restore authenticated configurations

- A description of the methods for retention and recovery of device configuration by an authenticated authorized user

- A description of the secure configuration of shipped devices, a discussion of the risk tradeoffs that might have been made about hardening options implemented by the device manufacturer, and instructions for user-configurable changes. Secure configurations may include end point protections such as anti-malware, firewall/firewall rules, allow lists, deny lists, security event parameters, logging parameters, and physical security detection, among others

- Where appropriate for the intended use environment, a description of how forensic evidence is captured, including but not limited to any log files kept for a security event. Log file descriptions should include how and where the log file is located, stored, recycled, archived, and how it could be consumed by automated analysis software (e.g., Intrusion Detection System, IDS)

- Where appropriate, technical instructions to permit secure network deployment and servicing, and instructions for users on how to respond upon detection of a cybersecurity vulnerability or incident

- Information, if known or anticipated, concerning device cybersecurity end of support and end of life. At the end of support, a manufacturer may no longer be able to reasonably provide security patches or software updates. If the device remains in service following the end of support, the manufacturer should have a pre-established and pre-communicated process for transferring the risks highlighting that the cybersecurity risks for end-users can be expected to increase over time

- Information on securely decommissioning devices by sanitizing the product of sensitive, confidential, and proprietary data and software

- A revision-controlled, Manufacturer Disclosure Statement for Medical Device Security (MDS2) and Customer Security Documentation as outlined in the HSCC Joint Security Plan (JSP) may address a number of the above recommendations.

Vulnerability Management Plans

Recognizing that cybersecurity risks evolve as technology evolves throughout a device's TPLC, FDA recommends that manufacturers establish a plan for how they will identify and communicate vulnerabilities that are identified after releasing the device with users. This plan can also support risk management processes in accordance with 21 CFR 820.30(g) and corrective and preventive action processes in accordance with 21 CFR 820.100.
FDA recommends that manufacturers submit their vulnerability communication plans as part of their premarket submissions so that FDA can assess whether the manufacturer has sufficiently addressed how to maintain the safety and effectiveness of the device after marketing authorization is achieved.

Vulnerability communication plans should include the following elements:

- Personnel responsible
- Sources, methods, and frequency for monitoring for and identifying
- Vulnerabilities (e.g., researchers, NIST NVD, third-party software manufacturers, etc.)
- Periodic security testing to test identified vulnerability impact Timeline to develop and release patches
- Update processes
- Patching capability (i.e., rate at which update can be delivered to devices) Description of their coordinated vulnerability disclosure process
- Description of how manufacturer intends to communicate forthcoming remediations, patches, and updates to customers

Appendix 1: Security control categories and associated recommendation

Authentication

There are generally two types of authentication controls - information and entities and a properly-secured system is able to prove the existence of both.

Authentication of **information** exists where the device and the system in which it operates is able to prove that information originated at a known and trusted source, and that the information has not been altered in transit between the original source and the point at which authenticity is verified. It is important to note that while authenticity implies that data is accurate and has been safeguarded from unauthorized user modification (i.e., integrity), integrity alone does not provide assurance that the data is real and came from a trusted source. Therefore, for the purposes of this guidance, authentication is discussed as a larger security objective over integrity.

Authentication of **entities** exists where a device and the system in which it operates is able to prove the identity of an endpoint (whether hardware and/or software) from which it is sending and/or receiving information, or authorized user/operator at that endpoint.

As part of normal operations within a secure system, devices are expected to verify the authenticity of information from external entities, as well as prove the authenticity of information that they generate. A system that appropriately accounts for authenticity will evaluate and ensure authenticity for: (1) information at rest (stored); (2) information in transit (transmitted); (3) entity authentication of communication endpoints, whether those endpoints consist of software or hardware; (4) software binaries; (5) integrity of the execution state of currently running software; and (6) any other appropriate parts of the system where a manufacturer's threat model and/or risk analyses reveal the need for it.

Recommendations regarding authentication

- Use cryptographically strong authentication, where the authentication functionality resides on the device, to authenticate personnel, messages, commands updates, and as applicable, all other communication pathways. Hardware-based security solutions should be considered and employed when possible

- Authenticate external connections at a frequency commensurate with the associated risks. For example, if a device connects to an offsite server, then the device and the server should mutually authenticate each session and limit the duration of the session, even if the connection is initiated over one or more existing trusted channels

- Use appropriate user authentication (e.g., multi-factor authentication to permit privileged device access to system administrators, service technicians, or maintenance personnel, among others, as needed)

- Require authentication, and permission in certain instances, before permitting software or firmware updates, including those updates affecting the operating system, applications, and anti-malware functionality

- Strengthen password protections. Do not use passwords that are hardcoded, default, easily-guessed, or easily compromised (e.g., passwords that are the same for each device; unchangeable; can persist as default; difficult to change; and/or vulnerable to public disclosure)

- Implement anti-replay measures in critical communications such as potentially harmful commands. This can be accomplished with the use of cryptographic nonces (an arbitrary number used only once in a cryptographic communication)

- Provide mechanisms for verifying the authenticity of information originating from the device, such as telemetry. This is especially important for data that, if spoofed or otherwise modified, could result in patient harm, such as the link between a continuous glucose monitor (CGM) system and an automated insulin pump

- Do not rely on cyclic redundancy checks (CRCs) as security controls. CRCs do not provide integrity or authentication protections in a security environment. While CRCs are an error detecting code and provide integrity protection against environmental factors (e.g., noise or EMC), they do not provide protections against an intentional or malicious actor

- Consider how the device and/or system should respond in event of authentication failure(s)

Authorization

Within an adequately designed authorization scheme, the principle of least privileges should be applied to users, system functions, and others, to only allow those entities the levels of system access necessary to perform a specific function.

While authentication schemes based on cryptographically-proven designs are generally considered more robust and are therefore preferred, meaningful authorization checks can be performed based on other compelling evidence (e.g., benefit/risk assessment in accordance with Section 6.5 of AAMI TIR57 and associated supporting justification and as evidenced through security testing).

For example, a medical device programmer that is capable of Near-Field Communications (NFC) could have elevated privileges that are granted based on a signal of intent over NFC that cannot physically be produced by another unauthorized device over Radio-Frequency (RF) (e.g., a home monitor).

Recommendations

- Limit authorized access to devices through the authentication of users (e.g., user ID and password, smartcard, biometric, certificates, or other appropriate authentication method)

- Use automatic timed methods to terminate sessions within the system where appropriate for the use environment

- Employ an authorization model that incorporates the principle of least privileges by differentiating privileges based on the user role (e.g., caregiver, patient, health care provider, system administrator) or device functions

- Design devices to “deny by default” (i.e., that which is not expressly permitted by a device is denied by default). For example, the device should generally reject all unauthorized connections (e.g., incoming TCP, USB, Bluetooth, serial connections). Ignoring requests is one form of denying authorization

Cryptography

Cryptographic algorithms and protocols are recommended to be implemented to achieve the secure by design objectives. While high-quality, standardized cryptographic algorithms and protocols are readily available, several commercial products that include cryptographic protections have been shown to have exploitable vulnerabilities due to improper configurations and/or implementations.

Recommendations

- Select industry-standard cryptographic algorithms and protocols, and select appropriate key generation, distribution, management and protection, as well as robust nonce mechanisms

- Use current NIST recommended standards for cryptography (e.g., FIPS 140-256, NIST Suite B57), or equivalent-strength cryptographic protection that are expected to be considered cryptographically strong throughout the service life of the device

- Design a system architecture and implement security controls to prevent a situation where the full compromise of any single device can result in the ability to reveal keys for other devices

    - Avoid using master-keys stored on device, or key derivation algorithms based solely on device identifiers or other readily discoverable information

    - Avoid using device serial numbers as keys or as part of keys. Device serial numbers may be disclosed by patients seeking additional information on their device or might be disclosed during a device recall to identify affected products and should be avoided as part of the key generation process. Public-key cryptography can be employed to help meet this objective

- Implement cryptographic protocols that permit negotiated parameters/ versions such that the most recent, secure configurations are used, unless otherwise necessary

- Do not allow downgrades, or version rollbacks, unless absolutely necessary for safety reasons. Downgrades can allow attackers to exploit prior, less protected versions and should be avoided

Code, Data, and Execution Integrity

Many cybersecurity incidents are caused, at their root, by the violation of some form of device integrity. This includes the violation of stored code, stored and operational data, or execution state.

Recommendations

- Code Integrity
    - Authenticate firmware and software. Verify authentication tags (e.g., signatures, message authentication codes (MACs)) of software/firmware content, version numbers, and other metadata. The version numbers intended to be installed should themselves be signed or have MACs. Devices should be electronically and visibly identifiable (e.g., Unique device identifier (UDI), model number, serial number)
    - Allow installation of cryptographically authenticated firmware and software updates, and do not allow installation where such cryptographic authentication either is absent or fails. Use cryptographically signed updates to help prevent any unauthorized reductions in the level of protection (downgrade or rollback attacks) by ensuring that the new update represents an authorized version change
        - One possible approach for authorized downgrades would be to sign new metadata for downgrade requests which, by definition, only happen in exceptional circumstances
    - Ensure that the authenticity of software, firmware, and configuration are validated prior to execution, e.g., “allow-listing” based on digital signatures
    - Disable or otherwise restrict unauthorized access to all test and debug ports (e.g., JTAG, UART) prior to delivering products
    - Employ tamper evident seals on device enclosures and their sensitive communication ports to help verify physical integrity
- Data Integrity
    - Verify the integrity of all incoming data, ensuring that it is not modified in transit or at rest. Cryptographic authentication schemes verify integrity, but do not verify validity
    - Validate that all data originating from external sources is well-formed and compliant with the expected protocol or specification. Additionally, as appropriate, validate data ranges to ensure they fall within safe limits

- Protect the integrity of data necessary to ensure the safety and effectiveness of the device, e.g., critical configuration settings such as energy output

- Execution Integrity
  - Use industry-accepted best practices to maintain and verify integrity of code while it is being executed on the device. For example, Host-based intrusion Detection/Prevention Systems (HIDS/HIPS) can be used to accomplish this goal

  - Carefully design and review all code that handles the parsing of external data using automated (e.g., static and dynamic analyses) and manual (i.e., code review) methods

Confidentiality

Manufacturers should ensure support for the confidentiality of any/all data whose disclosure could lead to patient harm (e.g., through the unauthorized use of otherwise valid credentials, lack of encryption). Loss of confidentiality of credentials could be used by a threat-actor to effect multi-patient harm. Lack of encryption to protect sensitive information and or data at rest and in transit can expose this information to misuse that can lead to patient harm. For example, confidentiality is required in the handling and storage of cryptographic keys used for authentication because disclosure could lead to unauthorized use/abuse of device functionality.

Event Detection and Logging

Event detection and logging are critical capabilities that should be present in a device and the larger system in which it operates in order to ensure that suspected and successful attempts to compromise a medical device may be identified and tracked. These event detection capabilities and logs should include storage capabilities, if possible, so that forensic discovery may later be performed.

Recommendations

- Implement design features that allow for security compromises and suspected compromise attempts to be detected, recognized, logged, timed, and acted upon during normal use. Acting upon security events should consider the benefit/risk assessment in accordance with Section 6.5 of AAMI TIR57 in determining whether it is appropriate to affect standard device functionality during a security event

- Ensure the design enables forensic evidence capture. The design should include mechanisms to create and store log files off the device to track security events. Documentation should include how and where log files are located, stored, recycled, archived, and how they could be consumed by automated analysis software (e.g.,Intrusion Detection System (IDS)). Examples of security events include, but are not limited to, configuration changes, network anomalies, login attempts, and anomalous traffic (e.g., sending requests to unknown entities)

- Design devices such that the potential impact of vulnerabilities is limited by specifying a secure configuration. Secure configurations may include endpoint protections, such as anti-malware, firewall/firewall rules, allow-listing, defining security event parameters, logging parameters, and/or physical security detection

- Design devices such that they may integrate and/or leverage antivirus/anti-malware protection capabilities. These capabilities may vary depending on the type of device and the software and hardware components it contains
    - For devices that leverage Windows Operating System
        - Antivirus/anti-malware is recommended on the device. Manufacturers are recommended to qualify multiple options to support user preferences for different options, especially if the device is used in health care facility environments.

    - For devices that leverage other Commercial Operating Systems (i.e., Ubuntu, Unix, Linux, Apple, Android, etc.)
        - Antivirus/anti-malware may be recommended based on the environment and associated risks of the device. Different operating systems will likely follow a case-by-case determination based on network exposure and risk

- For devices that leverage Embedded Operating Systems (i.e., Real-Time Operating Systems, Windows embedded, etc.)
    - Antivirus/anti-malware is generally not needed unless a particular risk or threat is identified that would not be addressed by other expected security controls

- Design devices to enable software configuration management and permit tracking and control of software changes to be electronically obtainable (i.e., machine readable) by authorized users

- Design devices to facilitate the performance of variant analyses such that the same vulnerabilities can be identified across device models and product lines

- Design devices to notify users when malfunctions, including those potentially related to a cybersecurity breach, are detected

- Consider designing devices such that they are able to produce a SBOM in a machine readable format

Resiliency and Recovery

Devices should be designed to be resilient to possible cybersecurity incident scenarios (also known as “cyber-resiliency”). Cyber-resiliency capabilities are important for medical devices because they provide a safety margin against unknown future vulnerabilities.

Recommendations

- Implement features that protect critical functionality and data, even when the device has been partially compromised. For example, process isolation, virtualization techniques, and hardware-backed trusted execution environments all provide mechanisms to potentially contain the impact of a successful exploitation of a device

- Design devices to provide methods for retention and recovery of trusted default device configuration by an authenticated, authorized user

- Design devices to specify the level of resilience, or independent ability to function, that any component of the system possesses when its communication capabilities with the rest of the system are disrupted, including disruption of significant duration

- Design devices to be resilient to possible cybersecurity incident scenarios such as network outages, Denial of Service, excessive bandwidth usage by other products, disrupted quality of service (QoS), and/or excessive jitter (i.e., a variation in the delay of received packets)

Firmware and Software Updates

Devices should be capable of being updated in a secure and timely manner to maintain safety and effectiveness throughout the product's lifecycle. Despite best efforts, undiscovered, exploitable vulnerabilities may exist in devices after they are marketed. This is especially true over the device's service life, as threats evolve over time and exploit methods change, and become more sophisticated.

FDA recommends that manufacturers should not only build in the ability for devices to be updated, but that manufacturers also plan for the rapid testing, evaluation, and patching of devices deployed in the field.

Recommendations

- Design devices to anticipate the need for software and firmware patches and updates to address future cybersecurity vulnerabilities. This will likely necessitate the need for additional storage space and processing resources

- Consider update process reliability and how update process works in event of communication interruption or failure. This should include both considerations for hardware impacts (timing specifics of interruptions) and which phase of the update process the interruption or failure occurs

- Consider cybersecurity patches and updates that are independent of regular feature update cycles

- Implement processes, technologies, security architectures, and exercises to facilitate the rapid verification, validation, and distribution of patches and updates

- Preserve and maintain full build environments and virtual machines, regression test suites, engineering development kits, emulators, debuggers, and other related tools that were used to develop and test the original product to ensure updates and patches may be applied safely and in a timely manner

- Maintain necessary third-party licenses throughout the supported lifespan of the device. Develop contingency plans for the possibility that a third-party company goes out of business or stops supporting a licensed product. Modular designs should be considered such that third-party solutions could be readily replaced

Appendix 2: Submission Documentation for Security Architecture Flows

In premarket submissions, FDA recommends that manufacturers provide detailed information for the views identified. Methods for providing the views and the expectations for the level of detail to provide are discussed in the sections below. In addition to diagrams and explanatory text, call-flow views can be provided to convey some of the information details expected to be addressed in the architecture views.

Call-flow diagrams

A call-flow view is a diagram with explanatory text that describes the sequence of process or protocol steps in explicit detail. For each of the views, manufacturers may provide call-flow information to detail the communications included in the associated use case.

Call-flow views should provide specific protocol details of the communication pathways between parts of the system, to include authentication or authorization procedures and session management techniques. These views should be sufficiently detailed such that engineers and reviewers should be able to logically and easily follow data, code, and commands from any asset (e.g., a manufacturer server) to any other associated asset (e.g., a medical device), while possibly crossing intermediate assets (e.g., application). The call-flow views may also include items from the information details identified below for the views identified if the information is better represented or conveyed through a call-flow view.

Information Details for an Architecture View

For each view described, manufacturers should provide a system-level description and analysis inclusive of end-to-end security analyses of all the communications in the system regardless of intended use. This should include detailed diagrams and traces for all communication paths as described below. Security-relevant analysis requires the ability to construct and follow a detailed trace for important communication paths, which describes how data, code, and commands are protected between any two assets in the device's system. This analysis can also help identify the software that should be included in the SBOM for each device.

The FDA recommends that security architecture views should include at least the following:

- Detailed diagrams and supporting explanatory text that identify all manufacturer and network assets of the system in which the device will operate, including but not limited to:
  - Device hardware itself (including assessments for any commercial platforms)
  - Applications, hardware, and/or other supporting assets that directly interact with the targeted device, such as configuration, installation/upgrade, and data transfer applications
  - Health care facility-operated assets
  - Communications/networking assets
  - Manufacturer-controlled assets, including any servers that interact with external entities (e.g., a service that collects and redistributes device data, or a firmware update server).
- For every communication path that exists between any two assets in the security use case view (and/or explanatory text), including indirect connections when there is at least one intermediate asset (e.g., an app), the following details should be provided:
  - A list of the communication interfaces and paths, including communication paths (e.g., between two assets through an intermediary), including any unused interfaces
  - An indication of whether the path is used for data, code, and/or commands, and type of data/information/code being transferred

- Protocol name(s), version number(s), and ports/channels/frequencies

- Detailed descriptions of the primary and all available functionality for each system asset, including assessment of any functionality that is built in but not currently used or enabled (e.g., dormant application functionality or ports), including assurance that this functionality cannot be activated and/or misused

- Access control models or features (if any) for every asset (such as privileges, user accounts/groups, passwords)

- Users' roles and levels of responsibility if they interact with the assets and communication channels

- Any "handoff" sequences from one communication path to another (e.g., from asset to asset, network to network, or Bluetooth to Wi-Fi), and how the data, code, and/or commands are secured/protected during handoff (i.e., how is their integrity/authenticity assured)

- Explanations of intended behavior in unusual/erroneous/unexpected circumstances (e.g., termination of a connection in the middle of a data transfer)

- Authentication mechanism (if any), including the algorithm name/version (if available), "strength" indicators (e.g., key bit length, number of computational rounds) and mode of operation (if applicable)

- Descriptions of the cryptographic method used and the type and level of cryptographic key usage and their style of use throughout the system (e.g., one-time use, key length, the standard employed, symmetric or otherwise). Descriptions should also include details of cryptographic protection for firmware and software updates

- Detailed analyses by cryptography experts if a cryptography algorithm is proprietary, or a proprietary modification of a standard algorithm

- For each authenticator created, a list of where it is verified, and how verification credentials (e.g., certificates, asymmetric keys, or shared keys) are distributed to both endpoints

- A precise, detailed list of how each type of credential (e.g., password, key) is generated, stored, configured, transferred, and maintained, including both manufacturer- and health care facility-controlled assets (e.g., key management and public key infrastructure (PKI))

- Identity management (if any), including how identities are managed/ transferred and configured (e.g., from manufacturer to programmer and from programmer to device)

- If communication sessions are used or supported, a detailed explanation of how sessions are established, maintained, and broken down, including but not limited to assurances of security properties such as uniqueness, unpredictability, time-stamping, and verification of session identifiers

- Precise links between diagram elements (or explanatory text), associated hazards and controls, and testing

- Explanations or links to the evidence that may be used to justify security claims and any assumptions

- Traceability to the SBOM described, above, for proprietary and third-party code

Appendix 3: Submission Documentation for Investigational Device Exemptions

FDA acknowledges the need to balance innovation and security in designs especially during clinical trials. In order to ensure security is addressed early in the device design, FDA has identified a subset of the documentation recommended throughout this guidance to submit with IDE applications. Under 21 CFR 812.25, manufacturers must provide an investigational plan as a part of their IDE application.

For devices within the scope of this guidance, FDA recommends that this investigational plan include information on the cybersecurity of the subject device.

Specifically, FDA recommends the following documentation be included as part of IDE applications:

- Inclusion of cybersecurity risks as part of Informed Consent Form (21 CFR 50.25(a)(2) and 21 CFR 812.25(g))

- Global, Multi-patient and Updateability/Patchability views (21 CFR 812.25(c), (d))

- Security Use case views for functionality with safety risks (e.g., implant programming) (21 CFR 812.25(c), (d))

- Software Bill of Materials (21 CFR 812.25(c), (d))

- General Labeling – Connectivity and associated general cybersecurity risks, updateability/process (21 CFR 812.25(f))

FDA intends to review this information in the context of the overall benefit-risk assessment of investigational devices as outlined in Factors to Consider When Making Benefit-Risk Determinations for Medical Device Investigational Device Exemptions. Therefore, approval of an IDE based on the documentation recommended above does not preclude the possibility of future cybersecurity questions or concerns being raised during review of a subsequent marketing application. This is, in part, due to the understanding that design changes may be needed and the temporal nature of security. Security improvements will likely be needed between the time of clinical trials and the device submitted for marketing authorization (e.g., operating system no longer supported or nearing end of support, third party software updates, etc.).

Sunbird
MD | MEDICAL DEVICES

www.sunbirdmedicaldevices.com