Health Sector Coordinating Council
Cybersecurity Working Group

Secure Medtech

Health Industry Cybersecurity -

Medical Device and Health IT Joint Security Plan Version 2.0

MARCH 2024

Table of Contents

I. Executive Summary

Software-based medical technologies have demonstrably improved patient care and outcomes. However, as these products become more connected, cybersecurity becomes increasingly important since cyber threats may disrupt care or otherwise impact patient safety. As cybersecurity is a shared responsibility, a wide range of healthcare stakeholders under the umbrella of the Healthcare and Public Health Sector Coordinating Council (HSCC), have drafted this Joint Security Plan (JSP) to address these cybersecurity challenges. These challenges include but are not limited to: transparent communication between suppliers and stakeholders, robust security by design taking stakeholder needs into adequate consideration, and the effective management of security risk throughout the total product lifecycle (TPLC). Specifically, the JSP is a secure product development framework for developing, deploying, and supporting cybersecure technology solutions in healthcare environments. It includes:

- Incorporating product security controls and processes in Design & Development of medical technology products.
- Managing product security risk throughout the lifecycle of medical technology.
- Assessing the maturity of a medical technology product security program.

The JSP is voluntary and seeks to aid medical technology organizations, including medical device manufacturers and healthcare information technology (IT) suppliers, in enhancing their product security program capabilities, irrespective of organization size or maturity. It is intended to be globally applicable, inspire organizations to raise the bar for product security, and is expected to evolve as product security practices evolve. Therefore, future versions of JSP are expected, and feedback on this version is welcome. Version 2 updates the JSP to include important new and updated references as well as a general refresh of content.

It is important for medical technology providers, including medical device manufacturers (MDMs) and healthcare IT suppliers (collectively referred to as "suppliers") to consider the JSP's voluntary framework when building or evaluating product security program maturity. Cybersecurity can be difficult to integrate into existing processes for a variety of reasons, such as organizations not recognizing its importance, lack of executive management support, not knowing where to start, or insufficient resources and/or internal expertise. The components of the JSP framework can help assess, prioritize, and accelerate product security maturity by providing terminology to help communicate concepts and map to the available supplemental resources that provide additional detail. Our primary request of manufacturers is to make a commitment to leveraging the JSP to understand and evolve their product security capability maturity, as this will ultimately enhance patient safety. The industry has a joint responsibility to ensure the use of medical technology provides the intended benefits, while significantly reducing the cyber risks that could degrade care and, in the worst cases, harm patients.

II. Background

In the Cybersecurity Information Security Act of 2015 ("CISA 2015"), the United States Congress established the Health Care Industry Cybersecurity ("HCIC") Task Force to identify the challenges that the healthcare industry faces when securing and protecting itself against cyber threats. Industry participation in the task force brought to light

critical gaps warranting focus, and a year-long discussion and analysis culminated in the release of a set of recommendations and action items to address six high-level imperatives.

In 2017, a group of MDMs stepped up to address the recommendations and action items set forth under Imperative 2 of the HCIC Task Force Report: *"Increase the security and resilience of medical devices and health IT*" by engaging healthcare delivery organizations (HDOs) in a collaborative effort that would produce a Joint Security Plan. This effort was further formalized under the auspices of the Healthcare Sector Coordinating Council's (HSCC) Joint Cybersecurity Working Group (CWG) public-private partnership, as the JSP was broadly socialized with healthcare providers, trade associations, security professionals, and government organizations during development and prior to its release. The U.S. Food and Drug Administration, in its role as a key public sector partner, also assisted with the development of the JSP.

Imperative 2 of the HCIC Task Force Report states:

Imperative 2. Increase the security and resilience of medical devices and health IT.

The Health Care and Public Health (HPH) Sector is charged with keeping patients safe and that includes protecting patients from physical harm, as well as privacy-related harms that may stem from an exploited known cybersecurity vulnerability. If exploited, a vulnerability may result in medical device malfunction, disruption of healthcare services (including treatment interventions), inappropriate access to patient information, or compromised Electronic Health Record (EHR) data integrity. Such outcomes could have a profound impact on patient care and safety. Some foundational challenges that will need to be addressed in order to enhance the cybersecurity of medical devices and EHRs include legacy operating systems, secure development lifecycle, strong authentication, strategic and architectural approaches to product deployment, management, and maintenance on hospital networks.

The relatively short lifespan for operating systems and other relevant platforms such as commercial off the shelf software is inherently misaligned in healthcare as medical devices and EHRs may be utilized for 10, 15, 20, or more years. This misalignment may occur for a variety of reasons. Hospitals operate on thin budgets and cannot replace capital equipment like Magnetic Resonance Imagings (MRIs) as quickly as new operating systems are released. Product vendors have a product development lifecycle that may take several years and they may start development using one operating system and by the time the product comes to market, newer operating systems may be available. Creative ways of addressing the aforementioned challenge areas may be found by engaging key clinical and cybersecurity stakeholders, including software vendors.

The JSP is expected to evolve over time. In 2024, this version (version 2) represents the first major refresh of the JSP since its initial drafting. In this version, the following priorities were addressed:

- Adjusted to reflect new, updated, or relevant standards and guidelines impacting healthcare. A listing of standards and guidance can be found in Appendix G.

- Applied learnings and improvements identified from the MDIC Medical Device Cybersecurity Maturity: Industry Benchmarking Report 2022, as well as a review from the HSCC conducted by a cross-industry group of manufacturers, HDOs, and regulators.1
- Refreshed framework and diagram to drive clarity and consistency. Specifically, framework components were updated to be "activities" in a secure product development framework and aligned directly to supporting content describing those activities.
- Increased integration of external standards and guidance into relevant activities. This supports easy referencing to find additional supporting information as the JSP is leveraged to mature product security capabilities. Traceability from JSP version 2 sections to relevant external standards and guidance can be found in Appendix G.

The JSP version 2 was not intended to be a complete rewrite, but rather integration of useful resources and clarification of the framework to make this a more useful tool for medical technology providers, regardless of size and scale. The HSCC CWG Medtech Task Group encourages medical technology suppliers to leverage this resource and provide feedback on how it can continue to evolve to meet future sector needs.

III. JSP2 Alignment with Health Industry Cybersecurity Strategic Plan

The health industry has evolved significantly since the Health Care Industry Cybersecurity Task Force report was published in 2017 and it will evolve still more beyond the 2024 publication of JSP2. In recognition of this dynamic, the HSCC Joint Cybersecurity Working Group published in February of 2024 a five year "Health Industry Cybersecurity Strategic Plan, (HIC-SP)" which projects health industry trends leading to 2029 and the associated cybersecurity challenges, and prescribes broad goals and objectives to measurably improve healthcare cybersecurity preparedness and resiliency by 2029. In alignment with this roadmap, every ensuing HSCC cybersecurity initiative and publication will address one or more of the 10 major goals and their 12 implementing objectives in order for our work to be focused on achievement of the Strategic Plan and the long term cybersecurity health of the sector. Accordingly, JSP2 is directly aligned with two Goals: Goal 6 and Goal 7:

- *Goal 6: Healthcare technology used inside and outside of the organizational boundaries is secure-by-design and secure-by-default while reducing the burden and cost on technology users to maintain an effective security posture; and*
- *Goal 7: A trusted healthcare delivery ecosystem is sustained with active partnership and representation between critical and significant technology partners and suppliers, including non-traditional health and life science entities.*

1 For example, supplier management was noted as a low maturity capability, and therefore JSP version 2 expanded supplier management content.

Appendix D of the HIC-SP maps how the pursuit and achievement of 6 implementing Objectives will address Goals 6 and 7, providing medical technology companies and the users of their products with an envisioned end-state for secure development and use of medical technology in the clinical environment.

Other freely-available related HSCC publications include: MedTech Vulnerability Communications Toolkit; Managing Legacy Technology Security; and Model Contract for Medtech Cybersecurity.

To provide feedback on the Joint Security plan: JSPFeedback@HealthSectorCouncil.org

IV. About the Health Sector Coordinating Council Cybersecurity Working Group

The Healthcare and Public Health Sector Coordinating Council (HSCC) is a coalition of private-sector critical healthcare infrastructure entities organized under the National Infrastructure Protection Plan to partner with and advise the government in the identification and mitigation of strategic threats and vulnerabilities facing the sector's ability to deliver services and assets to the public. The HSCC Cybersecurity Working Group (CWG) is the largest HSCC working group of more than 440 healthcare providers, pharmaceutical and medtech companies, payers and health IT entities partnering with multiple federal, state, international, and local government agencies to identify and mitigate cyber threats to health data and research, systems, manufacturing and patient care. The CWG membership collaboratively develops and publishes freely-available healthcare cybersecurity best practices and policy recommendations, and produces outreach and communications programs emphasizing the imperative that cyber safety is patient safety.

V. Acknowledgments

The following individuals constitute the membership of the committee established in 2022, responsible for the development of version 2.0 of the Medical Device and Healthcare Information Technology Joint Security Plan.

Chris Reed

Task Group Co-Chair
Vice President of Product Security
Medtronic

Debra Bruemmer

Task Group Co-Chair
Senior Director of Clinical Security
MedSec (formerly Senior Manager Office of Information Security, Mayo Clinic)

Aftin Ross

Task Group Co-Chair
Deputy Office Director
Office of Readiness & Response (OCR), Office of Strategic Partnerships and Technology Innovation (OST), Center for Devices and Radiological Health (CDRH) at US Food and Drug Administration (FDA)

Matt Vorhees

Content Leader (Editor)
Program Manager of Product Security
Medtronic

Arvin Eskandarnia

Content Leader (Editor)
Cybersecurity Specialist
Division of Medical Device Cybersecurity (DMDC), Office of Readiness and Response (ORR), Office of Strategic Partnerships and Technology Innovation (OST), Center for Devices and Radiological Health (CDRH) at US Food and Drug Administration (FDA)

Ken Hoyme

Content Leader (Risk Management)
Retired Sr. Fellow of Product Security
Boston Scientific

Axel Wirth

Content Leader (Design & Develop)
Chief Security Strategist
Medcrypt

Jason Sinchak

Content Leader (Verification & Validation)
Principal
Level Nine Group

Eirene Shipkowitz Smith

Content Leader (Maintenance)
Technical Consult of Product Security
Baxter

Ramakrishnan Pillai

Content Leader (Supplier Management)
Sr. Director of Product Security
LivaNova

Chris Gates

Content Leader (Standards)
Director of Product Security
Velentium

Colin Morgan

Content Leader (Assessing Maturity)
Managing Director
Apraciti

Edison Alvarez

Communications Leader
Sr. Director of Regulatory Strategic Planning
BD

Greg Garcia

Executive Sponsor
Executive Director at Healthcare Sector Coordinating Council

Jessica Wilkerson

Senior Cyber Policy Advisor and Medical Device Cybersecurity Team Lead
Division of Medical Device Cybersecurity (DMDC), Office of Readiness and Response (ORR), Office of Strategic Partnerships and Technology Innovation (OST), Center for Devices and Radiological Health (CDRH) at US Food and Drug Administration (FDA)

Matthew Hazelett

Cybersecurity Policy Analyst
Office of Product Evaluation and Quality (OPEQ), Center for Devices and Radiological Health (CDRH) at US Food and Drug Administration (FDA)

Nastassia Tamari

Division Director
Division of Medical Device Cybersecurity (DMDC), Office of Readiness and Response (ORR), Office of Strategic Partnerships and Technology Innovation (OST), Center for Devices and Radiological Health (CDRH) at US Food and Drug Administration (FDA)

Joe Burgoyne

Sr. Director of Cybersecurity
GE Healthcare

Uma Chandrashekhar
Global Product Information Security Officer
Alcon

Anura Fernando
Principal Security Advisor and Global Head of Medical Device Security
UL Solutions

Lev Frayman
Director of Product Security
Abbott

Les Gray
Sr. Director Enterprise and Product Cybersecurity
Abbott

Michelle Jump
CEO
MedSec

Dan Lyon
Director of Product Security
Boston Scientific

Kyle Munn
Sr. Director Cybersecurity – Business Development
GE Healthcare

Eddie Myers
Director of Cybersecurity
Crothall

Robert Rajewski
President
Critech

Inhel Rekik
Sr. Director of Product Security
Bracco

Matt Russo
Global Security Lead
Olympus Corporation

Andrew Sargent
Director of Product Cybersecurity and Privacy
Werfen

Kevin Scott
Shrinenet

Michael Seeberger
Sr. Manager of Product Security
Boston Scientific

Nick Sikorski
Head of Product Security
Deloitte

Bill Proffer
Master Solution Architect
Leidos

Satya Singh
Cybersecurity Architect
Siemens Healthineers

Christine Sublett
President & Principal Consultant
Sublett Consulting, LLC

Jitesh Veetil
Sr. Program Director
MDIC

Varun Verma
Regulatory Standards Manager
Phillips

Oleg Yusim
Sr. Director of Product Security
Edwards

Mark Snyder
Product Security Engineering
Intuitive

The HSCC Cybersecurity Working Group Medtech Task Group co-chairs would also like to thank all the individuals and organizations within the Healthcare Sector Coordinating Council (HSCC) that reviewed and contributed to the plan.

VI. Purpose and Objectives

The HSCC believes that, because medical technology is integral to patient safety and clinical operations, product security in medical technology is a shared responsibility among healthcare stakeholders. This document is titled the "Joint Security Plan" because it was written with involvement of stakeholders across the healthcare ecosystem and intends to reflect "joint" expectations of what a mature product security capability should look like for medical technology. The JSP represents a consensus based TPLC reference guide for developing, deploying, and supporting cybersecure technology solutions in the healthcare environment.

The JSP is not a regulatory document, nor is it a standard. Rather, it is a set of recommendations that may be leveraged across an organization's product portfolio and is intended to be globally applicable. The recommendations provided in the JSP are intended to help organizations of various sizes and stages of maturity to enhance their product security posture by addressing key cybersecurity challenges.

This voluntary plan is intentionally forward-leaning and seeks to inspire organizations to raise the bar for product security. Integrating cybersecurity into an organization necessitates organizational and process changes that come with considerable time and monetary investments. The JSP provides a framework for making these organizational and process-related changes and then measuring the maturity of implemented changes.

One of the main themes of the JSP is the concept of continuous improvement. The HSCC CWG encourages MDMs, health IT suppliers, and healthcare providers to make a commitment to leveraging the JSP to aid in developing, deploying, and supporting cybersecure technology solutions in healthcare environments. The adoption of the JSP, with the integration into current practices, is expected to provide safer and more resilient medical technologies that improve patient care and result in overall improved product quality.

VII. JSP Product Security Framework Overview

The JSP Product Security Framework is primarily for medical technology organizations, specifically MDMs and healthcare IT suppliers, and establishes that effective cybersecurity is integrated into an organization's quality system processes and incorporated throughout the TPLC. Figure 1 provides a framework for incorporating product security activities into existing quality system processes throughout the TPLC. The core of this framework aligns to traditional medical device quality system concepts leveraged in the development of medical technology. Design controls, risk management, design requirements, testing and postmarket management can be aligned with multiple

software development methodologies (not shown). Documentation of the product security activities in the JSP framework can demonstrate that the framework has been applied consistently and is adequately followed. Additional guidance and detail are provided for each product security activity or process identified in the JSP framework in Section IV of this document. Acronyms and term definitions used throughout the JSP may also be found in Appendix A and Appendix B, respectively.

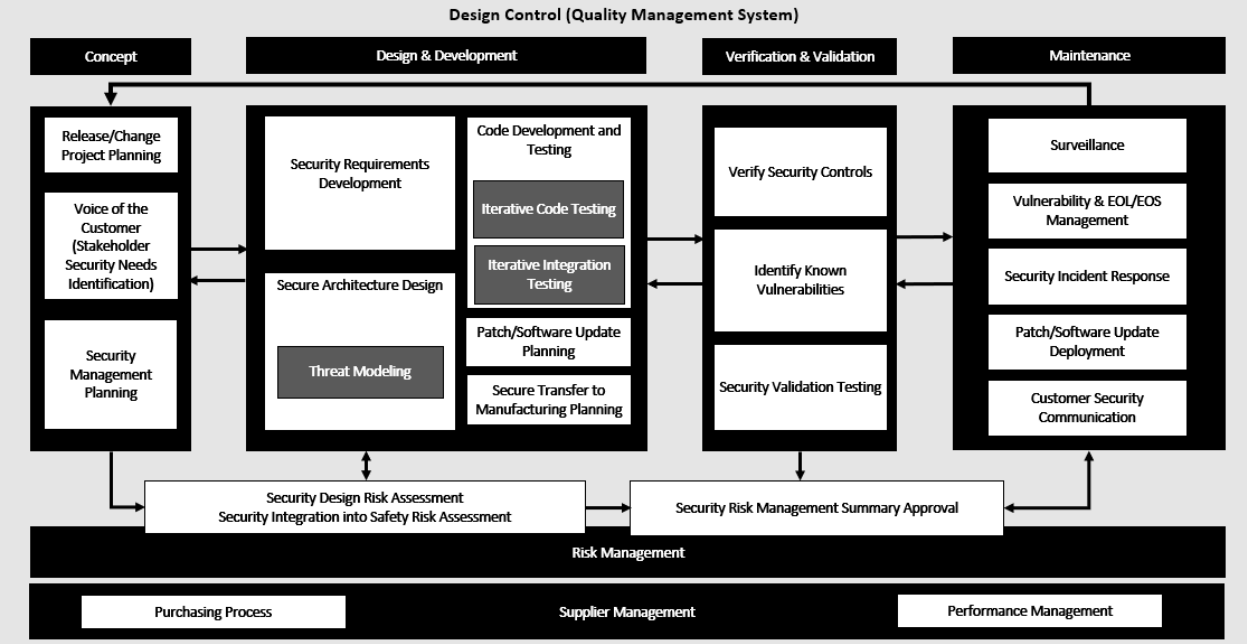

Figure 1. JSP Product Security Framework: Total Product Life Cycle Security Activities

The JSP Product Security Framework is a broad overview of security activities that are generally accepted as recommended practices throughout the TPLC of medical technology products. The black boxes in the framework represent the product lifecycle phase or capability where the activity generally takes place, and the white and grey boxes identify relevant product security activities.

However, it's important to note that this framework is a guide and not a rigid rulebook. Its purpose is to inform and prompt appropriate security activities at different stages of a product's lifecycle, rather than to dictate a specific order of activities or to cover every possible security consideration. It will often be the case that an activity earlier in the framework may need to be repeated as a product is developed and commercialized, especially when scope or major underlying technology changes occur. The framework is a starting point; it's important to adapt and supplement it according to the unique needs and risks of each medical technology product.

The following sections of the document are structured according to the phases of a product's lifecycle and include further details on the best practices for the security activities identified. These sections serve as an in-depth exploration of the concepts indicated in the framework, providing practical insights and recommendations for application in real-world scenarios. The framework is intended to enable medical technology organizations to assess,

prioritize, and mature product security capabilities to help ensure delivery of safe and secure medical technology products.2

VIII. Organizational Governance to Support JSP Implementation

To successfully use the JSP, an initial step is to define the governance process as it relates to organizational roles and responsibilities, and the needs for personnel training. Governance may include oversight of policies and procedures to support product security capabilities, monitoring product inventory alignment with the framework, and tracking of maturity against the framework and strategic investment decisions to improve product security. Framework adoption should be driven by mapping each of the framework cybersecurity activities into existing processes and minimizing the creation of separate or redundant processes. Effective governance oversight requires an accountable leader with required resources. In larger organizations, a product security function may exist that leads these activities, but the specific team that leads these activities may vary based on the organization.

Beyond organizational leadership, various members of the organization have a shared responsibility for product security, and thus benefit from the implementation of the JSP. For example, a manufacturer may share its evaluation of maturity against the JSP with customers. The manufacturer may also share this information with the HSCC and its partners such as MDIC with the intent of informing future iterations of the JSP. Additional granularity regarding stakeholder roles and responsibilities as well as potential organizational structures for implementing security are found in Appendix E.

In addition to establishing the activities outlined in the JSP framework, a culture must be established to ensure the activities can be successfully executed. Organizations adopting this framework should consider providing existing personnel with necessary training to achieve focused incorporation of cybersecurity expertise.3 Maintaining functional competency can best be achieved by establishing a routine training regimen and periodic reassessment of need.

IX. JSP Product Security Framework Implementation

The following sections further detail product security capabilities expected from a mature medical technology organization. For organizations whose capabilities are still maturing, the activities detailed in this section may be aspirational; however, such organizations should continue to mature their capabilities toward this desired state. The sections are organized by product lifecycle phases and capabilities, along with the underlying activities necessary in a

2 Healthcare delivery organizations (HDO) seeking further guidance on the secure operation of medical devices and other information technology used to run their healthcare operations may refer to the HSCC's "Health Industry Cybersecurity Practices (HICP): Managing Threats and Protecting Patients" publication, which also stemmed from CISA 2015. This industry published document outlines voluntary, consensus-based guidelines detailing practical, cost-effective cybersecurity practices an HDO can take to protect network connected medical devices.

3 See Appendix F for additional granularity regarding organizational training.

quality system to support safe and secure product development. This framework is not meant to be exhaustive, but rather provides a high-level summary of product security activities and relevant references where more information can further clarify the activity. To support ease of use, each section will map directly to Figure 2 and make it easier to find supporting content.

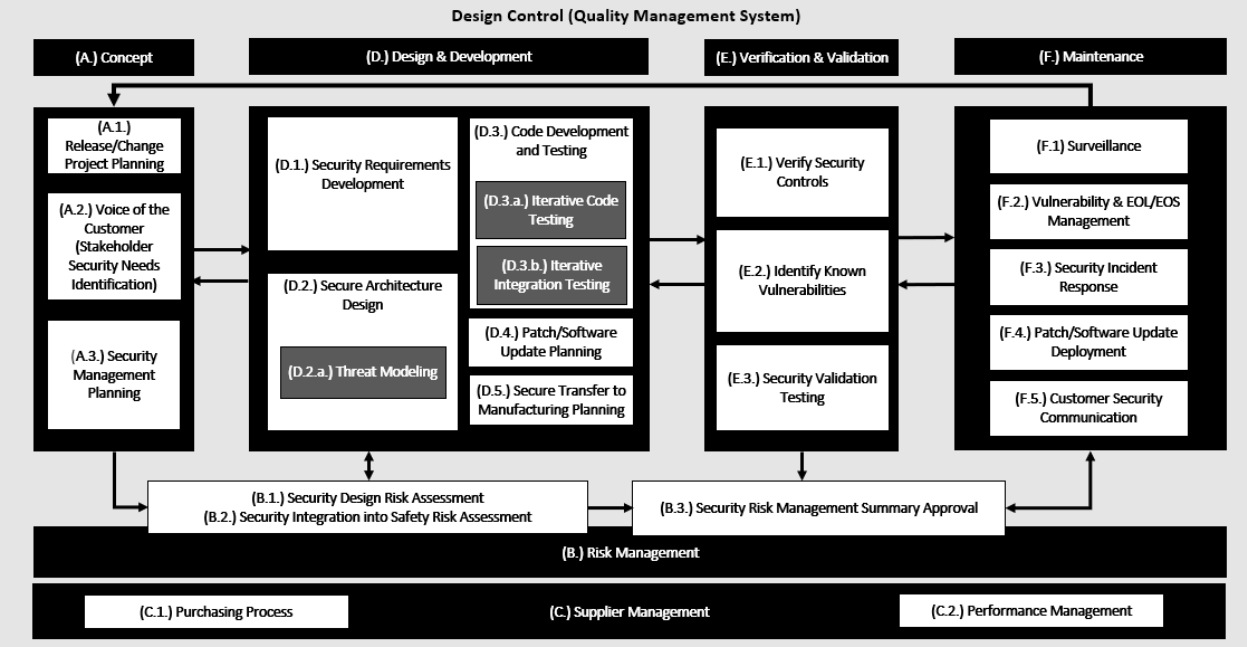

Figure 2. JSP Product Security Framework: Total Product Life Cycle Security Activities with Indexes

Figure 3 shows a subset of Figure 2 and represents the first product security activity detailed in Design & Development (Security Requirements Development) and Verification & Validation (Verify Security Controls). Section D contains the product security activities associated with Design & Development, while section E contains the product security activities under Verification & Validation. Each sub-section in section D and E represents a product security activity and directly maps to a "box" depicted in Figure 2.

Figure 3. Example JSP Framework Mapping to JSP Product Security Framework Implementation Content

A. Concept

The concept phase of the development lifecycle occurs as the scope of a development effort is being planned. This could be a brand-new product development effort, including early concept devices such as devices being planned for clinical trial use under Investigational Device Exemption (IDE), a major update to an existing product, or even a minor maintenance change to an existing product.

Failure to adequately consider security risk in this phase can result in lack of resources, both financial and human, required to ensure adequate security is built into the product. At this point in development, security can be introduced early on to address potential risks introduced by the scope of the current project.

The essential component of this phase is that adequate security considerations are planned into the development project. Effectively managing potential security risk in this phase is essential to ensuring adequate resourcing and planning throughout development efforts. For the purpose of this document, "cybersecurity" refers specifically to addressing security related to the safety and effectiveness of the device.

It should be noted that the concept phase may need to be revisited during development if major changes to scope are introduced that alter the risk profile of the device and may require additional security activities to be planned.

A.1. Release/Change Project Planning

Potential inputs:

- Project Charter/Change Proposal (project initiation)
- Joint Security Plan (for recommended product security activities for consideration)
- Initial User/Stakeholder/Architectural Needs

Potential outputs:

- Development Plan (as a project plan and/or official design history file document)
- Security Management Plan (create or update, section VII, A, iii. provides further details)
- Change Impact Assessment (for change)
- Estimation and commitment of financial and professional resources necessary to support identified product security activities

To ensure effective security planning, including resource allocation, it is essential to identify the relevant security management support activities specific to the project scope. Table 1 lists possible project scopes, and potential associated security activities.

Release/Change Type	Example Illustrations	Potential Security Activities	Comments

Brand-new, Including Investigational Device Exemption (IDE)	• New imaging device used to support a diagnosis for detection of prostate cancer. • New mobile application to interface with an existing imaging system with AI algorithms to support clinical evaluation of imaging.	All recommended JSP activities, unless sufficient justification exists for exclusion such as the activity happening at a system level of a product the in-scope component integrates into.	Often new Design History File (DHF) documents leverage DHF documents from an existing product in the market to serve as a starting point. An organization may have templates that can be leveraged as a starting point.
Major Update	• Change to intended use and/or contexts. • New Electronic Health Record (EHR) integration for an existing device to relay imaging results into a patient electronic medical record. • An additional AI algorithm added to filter clinical output and reduce noise that is displayed to the device user.	Security staff will have to assess the scope of the proposed project and prudently plan secure development activities appropriately. Additionally, any normal security activities such as security verification testing, and potentially penetration testing, will need to be planned based on the scale and type of changes being proposed.	Typically, major revisions are made to existing DHF documents, in addition to performing significant Verification & Validation activities.
Maintenance Update	• Consolidated patches for 3rd party components integrated and verified to ensure essential performance is preserved. • Replacement of a hardware component due to supply chain changes with a component of similar or better performance characteristics.	Assessing the extent of change introduced by in-scope maintenance activities is essential. Many security activities may be reduced or even bypassed if the changes being introduced are not significant from a security perspective. Security controls still must be reasonably verified to ensure changes did not adversely affect security control and the essential performance of the system.	Typically, minor revisions to existing DHF documents are performed with less significant verification activity to ensure minor changes do not introduce significant anomalies into the system. Documentation on why specific security activities are or are not performed are typically captured in the change impact assessment.

Table 1 Release/Change Type Impacting Secure Development Activities

Additionally, when integrating the development work into a larger system, careful consideration must be given to how changes in a component can impact the existing ecosystem. Assessing the potential impact of component changes on the ecosystem should also be incorporated into the project plan.

As part of Release/Change Project Planning, commitments on deliverables are typically documented in a Development Plan or a Change Impact Assessment. To adequately account for any new security risks introduced, it is imperative to involve a knowledgeable security professional in assessing the project's scope, resourcing, and project activities. In cases where the full set of JSP recommended activities are not planned, it is important to create defensible documentation explaining why normal secure development activities were reduced or eliminated for each Release/Change Type.

It is worth noting that this planning and assessment of secure product development activities aligns with the JSP Product Security Framework. While the framework provides guidance, it emphasizes the need for adaptation and supplementation based on the unique needs and risks of each product. The suggestions and recommendations presented in the following sections of the document delve deeper into the concepts outlined in the framework, providing practical insights for their application in real-world scenarios, all of which can be tailored to fit the specific security requirements of each product.

A.2. Voice of the Customer (Stakeholder Security Needs Identification)

Potential inputs:

- Contractual obligations from existing contracts or model contract language, such as that found in the HSCC Model Contract Language for Medtech Cybersecurity resource.
- Market research including specific connectivity (e.g., PACS, mobile apps).
- Regional markets planned for the product and associated regulatory, legal or standards required for the targeted markets.

Potential outputs:

- User Needs

Voice of the Customer is the Stakeholder Security Needs Identification process, specifically to identify security expectations of the audiences that will end up interacting with the product once it is in the market, in addition to the overall system level expectations that impact security.

It is essential to consider stakeholder voices that will interact with the product along with the various security needs they may have, as failure to do so may result in security controls that are cumbersome or difficult to use, potentially impacting their effectiveness and/or market adoption. Security needs identified in this step should be evaluated for inclusion in the security requirements process in the Design & Development phase of the project, and security requirements should be incorporated into the overall process of identifying user needs when defining the project scope. This ensures that, while addressing market demands, security considerations are not overlooked. If the Stakeholder Security Needs Identification process heavily relies on interviews with prospective users to define the User Needs, they may not provide detailed security needs or identify security as a need at all; however, it is always an implicit expectation of users of medical technologies and should be captured in this process.

Table 2 illustrates stakeholders and potential considerations to consider along with potential resources that could help identify specific details to feed into stakeholder needs.

Stakeholder	Considerations	Resources
Patients	• Methods to notify patients of necessary updates (potentially built into the product). • Security control considerations to ensure ease of patient use.	FDA PEAC materials including *FDA's Whitepaper on Best Practices for Communicating Cybersecurity Vulnerabilities to Patients4* Market Research
Healthcare Personnel (HCP)5 6	• Security control considerations to ensure ease of HCP use. • User roles required to perform device functions (e.g., surgeon, nurse). • Workflow integration considerations. • Data exchange interoperability capabilities. • Elements to convey during informed consent processes.	Market Research
HDO Support Staff	• Baseline controls that integrate into HDO's network. • Frequency of medical device updates. • Privileged access roles required to perform support activities (e.g., clinical engineering, general IT support). • Remote access or service support integration	IEC 80001 Series documents, HSCC HICP, HSCC Model Contract Language

4 https://www.fda.gov/about-fda/division-patient-centered-development/best-practices-communicating-cybersecurity-vulnerabilities-patients

5 https://www.hhs.gov/hipaa/for-professionals/security/index.html

6 https://www.hhs.gov/hipaa/for-professionals/privacy/index.html

	• Critical patch and incident response.	
Manufacturer Support Staff	• Key support capabilities identified for the product. • Necessary information from product inventories to provide support capabilities.	
Manufacturer Service Staff (Field Support)	• Privileged access roles required to perform support activities (e.g., field engineering). • Service models supported. • Remote access capabilities.	
Regulators	• Jurisdictions planned for the product and related requirements.	Expected security standards for the targeted regions

Table 2 Stakeholder Considerations

A.3. Security Management Planning

Potential inputs:

- Development Plan
- User Needs
- Change Impact Assessment
- Risk Management Plan (safety)
- Security Risk Procedure and/or Product Security Policy

Potential outputs:

- Security Risk Management Plan

A Security Risk Management Plan should document and establish how security will be managed throughout the TPLC. The plan should reference supporting policies and procedures where applicable, and detail what product security capabilities will be leveraged and how they will integrate with development efforts, including what DHF deliverables will be established.

The Security Risk Management Plan should be driven by a preliminary security risk analysis of what is known about the system in the concept phase to ensure the right level of rigor is planned. A clear understanding of this integration helps ensure adequate planning occurs, along with evaluation of whether activities in a particular phase of development can be considered complete.

Most importantly, the security management plan should document established risk acceptability expectations across the product development lifecycle, and specifically how risks will be evaluated to determine if they must be reduced. Definition of acceptable and unacceptable risks must be established to ensure security risk evaluation can effectively drive security design decisions.7 Additionally, the approach for integration of security risk with safety risk must be explicitly detailed.8

The following content should be considered in a security management plan, and may vary depending on the specifics of an organization's QMS structure and approach:

- Security Risk Acceptability, e.g., how security design risks will be assessed for acceptability, and known vulnerabilities will be triaged for confirmation and mitigation (this section)
- Risk Management approach and deliverables (see section B)
- Supplier management approach and deliverables (see section C)
- Design & Develop phase activities and deliverables (see section D)
- Verification & Validation phase activities and deliverables (see section E)
- Product Release and Maintenance deliverables (see section F)

Security Risk Acceptability is an essential activity that supports teams objectively evaluating security risk to determine if security controls are necessary. This activity requires that organizations define the criteria that will be used to evaluate security design risks in order to drive design decisions regarding whether additional controls are necessary, as well as known vulnerabilities to drive remediation decisions and timelines.

Section B—Risk Management further details the distinction and corresponding security activities that should be captured in the plan to manage security risks. In addition, Appendix I provides Common Security Risk Assessment methods that can be consulted to establish security risk acceptability.

In addition, the following items should be considered for coverage in the security risk management plan, if not already addressed through the items previously listed:

- Justification for the secure development framework being utilized, e.g., the JSP or other frameworks.
- Roles and responsibilities identified and documented at each phase of the TPLC.

---

7 Reference to QMS policy or procedure containing appropriate definitions is also acceptable.

8 Refer to AAMI/ANSI SW96:2023, Standard for medical device security—Security risk management for device manufacturers.

- Process for monitoring and identifying vulnerabilities in third-party software components including sources (e.g., Cybersecurity & Infrastructure Security Agency (CISA) Known Exploited Vulnerabilities Catalog, National Institute of Science and Technology (NIST) National Vulnerability Database), methods (e.g., Software Bill of Material (SBOM) scanning), and frequency.
- Process for assessing vulnerabilities in third-party software components, including interaction with the Corrective and Preventative Action (CAPA) process.
- Process for mitigating vulnerabilities in third-party software components.
- Contingency plan if a supplier of a third-party software component no longer supports the component, including cases where the manufacturer does not have access to the source code.
- Plan for providing updates to fielded devices, including expected periodicity; the maximum amount of time needed to provide updates; risks to the devices that are not able to be updated; and how the build environments are to be archived and maintained to allow the possibility of updates in the future.
- Process for creating and issuing coordinated disclosures of vulnerabilities affecting fielded products.
- Process for performing security testing in a sustained way on a consistent timeline.

This security management plan should be cross-functionally reviewed and approved by business leadership in addition to R&D and quality teams, including sufficiently qualified product security resources. Additionally, the document should be a living resource that is updated as new scope or learnings occur in development and maintenance of a product.

B. Risk Management

Security risk management is an integral component of overall product risk management. There are specific considerations necessary for ensuring that cybersecurity risks identified during design, development, or post launch are properly analyzed, evaluated, and controlled to adequately reduce and document acceptable residual risk. This section describes security risk management activities from product concept through end of support and how they should integrate into overall risk management activities.

Risks to the security of a product can be broad, and as a result the JSP leverages two specific categories of security risks to help provide clarity on where certain risks are addressed within the TPLC management process and resulting documentation. This distinction is established in *ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers* section B6, Sources of Risk, and is further emphasized in this document. Two categories of risks to security are also defined in Appendix B**:**

- ***Security Design Risk:*** An error in design that may result in degraded security. Errors or weaknesses are often categories or general potential weaknesses and not specifically known vulnerabilities.9 Common Weakness Enumeration (CWE) could be a useful resource for identifying types of these Security Design Risks.

9 ANSI/AAMI SW96:2021, Section B6 Sources of Risk

- ***Known Vulnerability:*** A publicly identified specific weakness in the computational logic (e.g., code) including configuration found in software and hardware components that, when exploited, results in a negative impact to confidentiality, integrity, or availability. A known vulnerability in a released product or software should have an identified CVE and associated inherent CVSS score.10

Practices in this space are still evolving, and specific security risk management documentation practices may vary across manufacturers. However, it is essential that organizations distinguish these categories of security risk, as different security activities are leveraged to identify, evaluate, and often document them.

Risk assessments combine two factors, one that represents frequency of occurrence, and the second addressing the impact when the risk occurs. Medical device safety risk assessment as defined in ISO 14971:2019 and ISO/TR 24971:2020, uses probability as the occurrence factor and severity of harm as the impact factor. The intent of this analysis is to have an estimate of real probability using the math of reliability analysis to arrive at a precise value. However, it is well understood that estimation of the mathematical probability of security risk occurrence is difficult, if not impossible, as it involves estimating the human behavior of an attacker, which cannot be realistically estimated with mathematical precision.

Instead, many security risk standards use "likelihood" as the measure of risk occurrence, defined as:

- ***Likelihood of Occurrence***: A weighted factor based on a subjective analysis of the probability that a given threat is capable of exploiting a given vulnerability or a set of vulnerabilities.11

However, some manufacturers have noted that in this usage "likelihood" and "probability" are not proper nouns. Instead, FDA has explained it its *2016 guidance, Postmarket Management of Cybersecurity in Medical Devices*[12]*, and in its 2023 guidance, Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions*[13], that "cybersecurity risks are difficult to predict, meaning that it is not possible to assess and quantify the likelihood of an incident occurring based on historical data or modeling."

The FDA introduced the term "exploitability," defined as:

- ***Exploitability:*** The feasibility or ease and technical means by which the vulnerability can be exploited by a threat.

The use of exploitability as the occurrence factor maps it to a subjective range of values. At the highest value would be known vulnerabilities that are known to be exploited in the real world, for which the device has no compensating

10 NIST CVD, also consistent with UL 2900-1.

11 NIST SP800-30:2012, ANSI/AAMI SW96:2021

12 https://www.fda.gov/regulatory-information/search-fda-guidance-documents/postmarket-management-cybersecurity-medical-devices

13 https://www.fda.gov/regulatory-information/search-fda-guidance-documents/cybersecurity-medical-devices-quality-system-considerations-and-content-premarket-submissions

controls in place.14 Alternatively, if a device effectively implements a defense in depth design strategy, a complex chained exploit may be required to activate a known vulnerability.

Furthermore, the terms *exploitable* and *exploitability* have very specific meanings and their usage in security risk assessment contexts must be precise. For a known vulnerability to be *exploitable*, a specific repeatable method must exist that successfully leverages the known vulnerability to realize a risk. On the other hand, *exploitability* is "the feasibility or ease and technical means by which the vulnerability can be exploited by a threat". In the latter, a specific repeatable method does not exist and instead exploitability serves as an estimation of the likelihood.

It is important for organizations to note that the impact factor for security risk analysis is more complex than just harm (safety). Other impact factors include privacy, confidentiality, integrity, and availability aspects, and business impacts (cost/reputation), among others.

B.1. Security Design Risk Assessment

Potential inputs:

- User Needs
- Development Plan
- Change Impact Assessment
- Security Requirements
- Risk Management Plan
- Security Management Plan
- Architecture Documentation

Potential outputs:

- Security Design Risk Assessment

Security Design Risks15 are identified, managed, and documented through three key product security activities:

- Security Risk Assessment
- Security integration into Safety Risk Assessment, and
- Security Risk Management Summary Report

Security Design Risks are managed through activities documented in this section and are part of the overall required risk management for a device, including safety risk. Most of this activity occurs during the Design & Development phase of product development, but new Security Design Risks that require further consideration and documentation can be identified at any time, including during postmarket maintenance activities.

---

14 For example, the WannaCry malware exploited known vulnerabilities in Microsoft's SMB v1.0 protocol. If a device had the port/service open to the network, the device would be highly exploitable for this known vulnerability. If the device was hardened that the SMB service and ports were disabled, the exploitability would be much less.

15 In ANSI/AAMI SW96, these risks are termed "security risks", but the word "design" is added in the JSP to help further clarify the errors in design specified in ANSI/AAMI SW96.

The objective of security design risk assessment is to identify, evaluate, control, and accept security design risks, informed by a determination of those risks' comprehensive impact, for example, to clinical safety, business operations, intellectual property, patient privacy, contractual requirements, regulations, and laws. The risk assessment also enables the risks and vulnerabilities to be prioritized for response. Security Design Risk assessment should be done in a way that reflects the target use environment and use cases of the product, and security design risks should be mapped to the security control requirements used to manage the security design risks to acceptable levels as defined in a security management plan or other policy/procedure reference.16

Security Risk Acceptability of both security design risks and known vulnerabilities should have been established in a security management plan or other shared policy/procedure documentation, and organizations may have two different methods for evaluating security risk between the two different categories. Additionally, the method(s) for integration of security risk with overall risk management, including safety, should also be established.17

Once the security design risk assessment has been documented, it can be populated while other product security activities are performed.18 Threat modeling documentation is a primary input and identification methodology to the security design risk assessment, and threat modeling is performed periodically throughout the product design process.19 Accordingly, the security design risk assessment should be periodically revised as the product design evolves and threat modeling activities establish an updated understanding of the product's attack surface and design vulnerabilities.

For each security design risk identified, manufacturers should identify controls (if any) which mitigate the identified risk. Mitigating controls should take the form of system or software requirements to support Verification & Validation during subsequent phases of the product lifecycle. Manufacturers may find that minimal mitigating controls are present during the early stages of Design & Development and mitigating controls should be added to the security design risk assessment artifact as requirements are developed, documented, and refined.

As noted in Figure 1, it is possible for other lifecycle phases, including maintenance phase activities, to cause reconsideration of Security Design Risks. Also, it is generally expected that security risk files are reviewed periodically, and any new information should result in appropriate updates to relevant security risk documentation.

Known vulnerabilities should be identified during the development activities of Design & Development phase whenever possible. However, vulnerabilities must be formally identified, documented, and dispositioned in the Verification & Validation phase, as well as the Maintenance stages. While vulnerabilities fully mitigated during

---

16 The security design risk assessment artifact commonly takes the form of a table with security Design Risk and Known Vulnerabilities assessed according to the organizations criteria, mitigations with links to the requirements and potentially verification artifacts as well as an assessment of the residual risk. Verification traceability can also be handled following existing requirement verification documents.

17 See Section B2

18 To ensure that learnings are brought forward into newly developed products, additional inputs to the security design risk assessment should include security design risks in any previous versions of the product or similar products.

19 See Section D(2)(a)—Threat Modeling

Design & Development do not necessarily require formal documentation and disposition, those identified during Verification & Validation or Maintenance of a released product must be documented.

An impact analysis should be conducted for these vulnerabilities, and their disposition should be based on the defined Security Risk Acceptability. For risks accepted without further mitigation, a risk-benefit analysis must also be conducted, with its rationale thoroughly documented. The method for documenting known vulnerabilities might differ across organizations, but any remaining vulnerabilities and their dispositions should be part of the Security Risk Management Summary Report, as outlined in section B3. This could lead to periodic updates to the residual Security Risk Report, but at the very least, it should be updated during the product's formal Verification & Validation phase.

In the FDA's *Postmarket Management of Cybersecurity in Medical Devices*, their recommendation is to measure security risk via the axes of "exploitability" and "impact." Several means for estimating exploitability are possible. A subset of the CVSS scoring components address exploitability (e.g., attack vector, attack complexity, privileges required, and user interaction). Other exploitability measures may be considered.20

Cybersecurity impact should be assessed along several categories, including but not limited to:

- Confidentiality, Integrity, and Availability, which may result in impact to:
  - Privacy risk including patient information
  - Business risks including intellectual property and business operations
- Patient safety including integrity and essential performance, therefore requiring linkage of security risks that impact patient safety to the manufacturer's safety risk assessment processes.

Note that a single vulnerability may have multiple impacts, and the risk should be managed for the worst potential impact. Figure 4 illustrates this process, where:

- Low risk means negligible or no impact to safety, privacy, confidentiality, integrity, availability, or business risk to the patient, user, manufacturer, or customer environment. From an FDA safety perspective, this may be considered controlled risk.
- Medium to high risk means potential vulnerabilities that may result in adverse events impacting safety, privacy, confidentiality, integrity, availability, or business risk to the patient, user, manufacturer, or customer environment. From an FDA safety perspective, this may be considered uncontrolled risk, depending on impact to safety and efficacy.
- Critical risk introduces potential for harm to patients or users of products, including impact to sensitive information and data or critical functions, or critical impact to business risk. From an FDA safety perspective, this may be more likely to be considered uncontrolled risk.

[20] See Appendix I for a discussion of options.

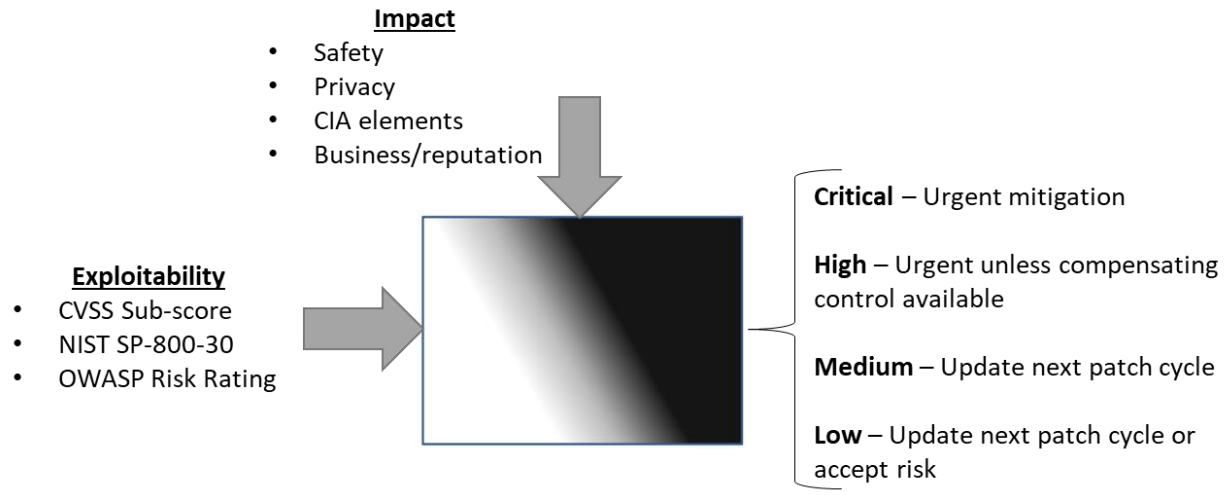

Figure 4. Security Risk Assessment Is A Combination of Exploitability And Impact

B.2. Security Integration into Safety Risk Assessment

Potential inputs:

- Risk Management Plan
- Security Requirements
- Security Management Plan
- Security Design Risk Assessment
- FDA Guidance: Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submission

Potential outputs:

- Risk Assessment Report (e.g., Failure Mode and Effects Analysis (FMEA), Software Failure Mode and Effects Analysis (sFMEA))

Security risk assessment is performed as part of a wholistic approach to risk that is rooted in safety risk.21 While safety and security risk management follow similar paths, the differences in assessing exploitability (for security risk) versus probability (for safety risk) and assessing multiple classes of impacts for security risk makes a parallel but linked process easier to manage.

As shown in Figure 5, ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers documents that there are specific links between the security risk and safety risk management processes:

21 ISO 14971:2019 and ISO/TR 24971:2020

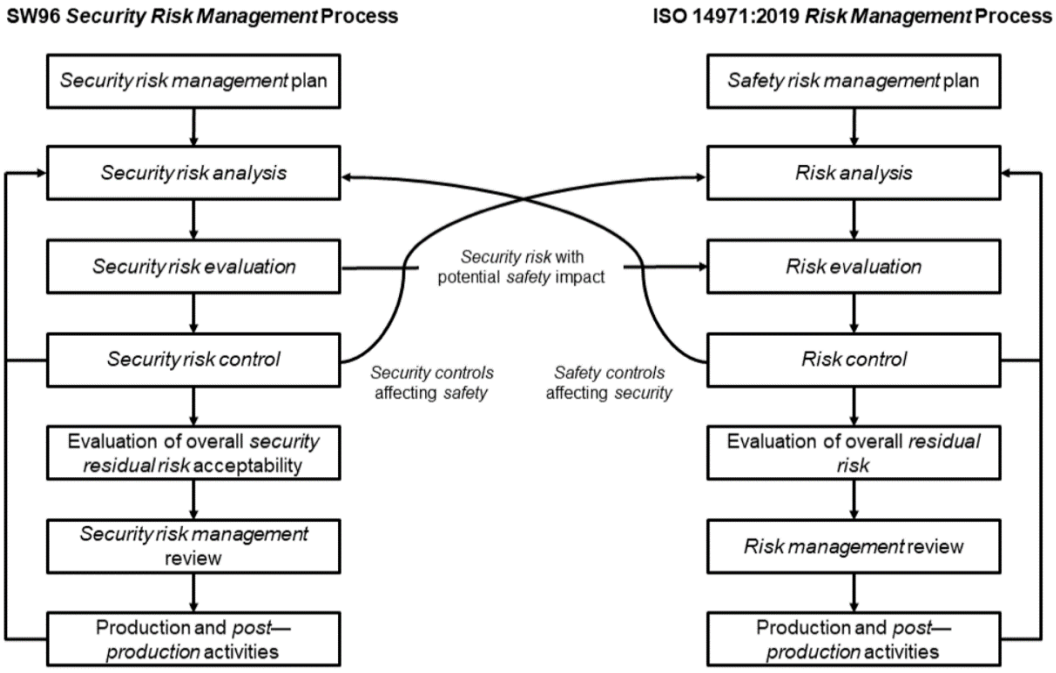

Figure 5. Relationship Between Security Risk and Safety Risk Management Processes (Reprinted with permission from ANSI/AAMI SW96:2023; STANDARD FOR MEDICAL DEVICE SECURITY—SECURITY RISK MANAGEMENT FOR DEVICE MANUFACTURERS. Arlington, VA: Association for the Advancement of Medical Instrumentation; Figure 2. © AAMI.)

Any security risk which can have a safety impact should be propagated to the safety risk management process. Any risk control introduced in either process should be assessed in the parallel process to see if it introduces new risks.

It should be noted that risks that are propagated from security risk management process into the safety risk management process will likely require association with a harm and hazardous situation. For example, denial of service on wireless communications in the security risk assessment may need to be associated with a hazardous situation of being unable to start and/or stop therapy on a medical device, if the device's command and control capabilities are entirely wireless. Safety risk management typically uses probabilities as the occurrence factor for hazardous situations, but in this case the occurrence factor should be exploitability. Various methods exist to make this translation, and currently there is not an established consensus method.

The FDA recognizes this in the 2023 premarket cybersecurity guidance, Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions which states:

"the associated acceptance criteria as well as the method for transferring security risks into the safety risk assessment process should also be provided as part of the premarket submission."

Regardless of the approach, quantitative probabilities must not be leveraged for assessing the likelihood of a security risk. Exploitability must be leveraged in a rational way as a qualitative or semi-quantitative approach and this method must be documented. Furthermore, basic tests of the methodology should yield defendable results.

Manufacturers should attempt to proactively identify where the absence of mitigating controls could result in an unacceptable risk as defined in the Security Management Plan, Risk Management Plan, or relevant organizational risk management policy/procedure. Potential unacceptable risk should drive design changes or implementation of additional mitigating controls to control risk to an acceptable level.

Known vulnerability related security risks may also need to be assessed for potential safety impact so decisions on mitigations can be made comprehensively. Often a known vulnerability will be triaged and assessed, including reviewing whether an overall risk management review, including safety, should occur based on the results of the known vulnerability risk acceptability established by the organization.

B.3. Security Risk Management Summary and Approval

Potential inputs:

- Risk Management Plan
- Security Management Plan
- Security Design Risk Assessment
- Safety related risk assessments (e.g., PHA, FMEA, Safety Assurance Case)
- Penetration and Verification testing reports (including security verification)
- Scan reports from tools looking for known vulnerabilities

Potential outputs:

- Residual Security Risk Report (residual security risks can be captured in the residual risk report, and therefore could either be included in that report or separately)
- Residual Risk Report (potentially updated with security-related content contributing to the overall residual risk of the product)
- Security Risk Management Summary Report

An artifact such as a Residual Security Risk Report, Residual Risk Report, or Security Risk Management Summary Report should be created that summarizes the Threat Model, Security Design Risk Assessment, and any security-related anomalies identified during Verification & Validation, along with their disposition. This includes the identification of known vulnerabilities. The artifact should be updated to reflect the final residual security risk in the device.

Management approval of this artifact should be documented prior to regulatory submission. Annex C in ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers provides a good summary of the content expected in this document, which serves a dual purpose. First, it acts as a condensed version of more comprehensive documents, aiding leadership and essential product development and support teams in comprehending the overarching residual risk. Second, it is suitable for inclusion in regulatory submissions.

It is not intended to be shared publicly given the level of risk evaluation details, but it may be useful to inform customer-facing documentation of summarized residual risks that stakeholders should consider in their risk management programs.

C. Supplier Management

A supplier is any individual or entity that provides any type of product and/or service, including firmware/software, that is part of a medical device system.22 The word supplier may include, but is not limited to, contracted developers or service providers (both individual and companies), component suppliers (both hardware and software), and open-source library developers.23

Supply Chain Risk Management (SCRM) is a systematic process for managing security risk exposures, threats, and vulnerabilities throughout the supply chain as well as developing strategies to respond to the risks presented by the supplier, the supplied products and services, or the supply chain. Manufacturers should have procedures including security risk management leveraged to evaluate, contract, and manage performance of prioritized suppliers.

SCRM should cover components used as part of the product and the tools used in the development, manufacturing, and distribution processes. An insecure component used in any of these processes could adversely impact the security of producing the code, creating the product, or getting the product to users.

Manufacturers should cover the different activities of the SCRM process described above as part of their QMS documents. The following sections describe best practice security activities which manufacturers should include as part of a SCRM program.

C.1. Purchasing Process

Potential inputs:

- ISO 13485 (section 7.4 Purchasing in ISO 13485:2016+A11:2021)
- Organizational procedures for supplier management
- ISO IEC 81001-5-1 Health software and health IT systems safety, effectiveness and security – Part 5-1: Security – Activities in the product life cycle Ed. 1.0 4.1.5 Software Items from third-party suppliers, 5.2.3 Security risks for Required Software
- HSCC Health Industry Cybersecurity – Supply Chain Risk Management (HIC-SCRiM) Guide v2.024
- NIST SP 800-161r1 Cybersecurity Supply Chain Risk Management Practices for Systems and Organizations25

22 ISO 13485:2016+A11:2012 section 7.4 Purchasing as well as other references of supplier throughout

23 IEC 62304:2006 & A1:2016 section 7.1.3

24 https://healthsectorcouncil.org/wp-content/uploads/2023/10/HIC-SCRiM_2023-2.pdf

25 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-161r1.pdf

- Health Information Sharing and Analysis Center (Health-ISAC) Medical Device Cybersecurity Lifecycle Management, Sec.5 - Main Lifecycle Phases – Supply Chain26
- NIST Cybersecurity Framework (CSF) 2.0 Appendix A: CSF Core, Supply Chain Risk Management (GV.SC) – Specifically GV.SC-01, GV.SC-02, GV.SC-0327
- HSCC Health Industry Cybersecurity – Managing Legacy Technology Security (HIC MaLTS) 202328, section VIII, Challenges and Recommendations, G. Third Party Component Risk Management
- SBOM and/or SOUP list(s)

Potential outputs:

- Supplier contract and/or license
- Supplier Service Level Agreements (SLAs)
- Supplier information collected and approved as part of the approved supplier list
- Supplier Risk Management Plan
- Relevant Supplier Risks Assessment(s)

SCRM activities should be applied to all supply chain components, regardless of whether payment is required or not (e.g., use of open-source software) to ensure that supplier assessment, contracting/licensing, and ongoing risk management is adequately performed. For manufacturers, an approved supplier list (ASL) is often maintained to satisfy required quality record requirements. The scope and nature of suppliers can vary widely, but the activities documented in this section should be applied appropriately. Manufacturers will likely have their own supplier management procedures, and they should be leveraged to ensure that all suppliers involved in a product have been adequately managed for risks.

Activity	Description/Reference
Security Assessment	Perform a security review of the supplier organization and the component to ensure the supplier is following industry best practices around secure software development, and the supplier has a strong information security management program in place.

26 https://h-isac.org/wp-content/uploads/2020/10/IHE_MDSISC-Lifecycle-Management-Working-Group-Whitepaper.pdf

27 https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.29.pdf

28 https://healthsectorcouncil.org/wp-content/uploads/2023/03/Health-Industry-Cybersecurity-Managing-Legacy-Technology-Security-HIC-MaLTS.pdf

Contracting/Licensing	All supplier contracts should have product requirements and security requirements included. Necessary privacy requirements should also be put into place if the supplier will have any access to personal information that are relevant for the geography, such as: • Business Associate Agreement (BAA) - If supplier is handing US Protected Health Information (PHI). • Data Processor Agreement (DPA) - If supplier is handling personal information of European citizens or residents (data subjects under GDPR). There are several security considerations for contracts, but the following should be specifically considered: • Require coordinated vulnerability disclosure and specify how vulnerabilities will be communicated, and on what timeline. • Require SBOMs for any software provided by the supplier, especially when it may be included in a finished product. • Require security incident management procedures, including how suppliers will notify manufacturers and within what timeline. • Require Secure Development Life Cycle (SDLC) activities, including requirements, architecture, code review, testing, and training. Additional considerations are provided in HSCC Model Contract Language for Medtech Cybersecurity. Additional recommendations can be found in the resources included in the "inputs" list at the beginning of the section. Depending on the scope of work for the supplier, a Quality Agreement may be required to ensure the manufacturer understands their role and is able to satisfy regulatory requirements and customer commitments related to product security.
License/Security/Export Control Review	Use a checklist to ensure the components comply with intellectual property, licensing, security, and export control requirements. This is especially important in the case of open-source components which may bypass the contracting process.
Determine and Implement Risk Controls	Based on security assessment and contracting/licensing review, and in alignment with organizational procedures, establish necessary controls to manage risk to organizationally aligned risk acceptability thresholds. This may include creating a supplier risk management plan and/or establishing the frequency at which supplier reviews will be conducted.

C.2. Performance Management

Potential inputs:

- Supplier Risk Management Plan(s)
- Supplier SLAs
- Supplier contract(s)/license(s)
- Product Security Incident Response Plan(s)

- Vulnerability monitoring procedure(s)

Potential outputs:

- Records supporting identification and disposition of supplier vulnerabilities
- Periodic supplier review report(s)
- Supplier Performance Review Meeting(s)

The performance of a supplier should be managed appropriately based on the initial supplier review and should be added to an ASL. Manufacturers' procedures and any relevant commitments to manage supplier risk should be executed and documented appropriately. The following activities may be appropriate depending on the scope of the supplier's work.

Activity	Description/Reference
Vulnerability Monitoring	It is a regulatory requirement to monitor cybersecurity vulnerabilities in all third-party components used in a manufacturer's medical devices. This includes security updates from third party suppliers, vulnerability advisories/disclosures, and security testing29 of products. The supplier's method for sharing information should be integrated into vulnerability monitoring for affected products.
SBOM	The supplier should provide an updated SBOM for every release of their products. Monitoring for updates to new versions of SBOMs from suppliers should be performed as part of Vulnerability Management activities. Reference section F.2 for further details.
Incident Management	Notification from suppliers should be integrated into incident response processes. How suppliers notify, including who they notify and how fast, should be documented and maintained.
Supplier Risk Review	Periodic assessment of suppliers' security risks is crucial to ensure that supplier controls are implemented and maintained throughout the lifecycle of usage of the supplier in an MDMs product.

D. Design & Development

The Design & Development phase establishes detailed specifications for a product, including implementation. In the formal language of design controls, design inputs and design outputs are created in this phase, which often involves iterative work as initial requirements are refined through architecture reviews and practices such as threat modeling.

29 It is recommended that security testing of 3rd party components be integrated into Design & Development (D.3) and Verification and Validation (E). Integrated testing could include SCA looking for third-party known vulnerabilities in supplier provided code. See Appendix H for more information on security testing tools that may be appropriate.

This section details the significant security activities that occur in this phase.

D.1. Security Requirements Development

Inputs:

- ISO IEC 81001-5-1 Health software and health IT systems safety, effectiveness, and security – Part 5-1: Security – Activities in the product life cycle Ed. 1.0 Section 5.2 Health Software Requirements Analysis
- ISO 13485 7.3.3 Design and Development Inputs
- FDA Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions, Appendix 1. Security Control Categories and Associated Recommendations
- User Needs (see Section VII A ii and Appendix L).
- Security Controls
- Organizations should identify, apply, and maintain system hardening standards provided by a third-party component supplier or an authoritative source for securely configuring all products and components used in a manufacturer's product.

Outputs:

- System and/or Software Requirements

This section describes how security requirements can be integrated into the existing requirements process in a fashion that is conducive to ensuring appropriate implementation during development and appropriate testing of the requirements in the existing Verification & Validation operation.

Due to the complexity of security requirements, the variety of implementation methods available, and the involvement of cybersecurity subject matter experts (SMEs), it is vital for the development team to work closely with Product Security SMEs. This collaboration is essential to identify and rectify weak or ineffective design aspects in the early phases before product development commences. Ultimately, all security requirements should be testable and traced to design elements and test cases that verify proper implementation. System, software, and security requirements should be reviewed by designated security experts to provide input on the appropriateness and correctness of specified security controls.

The application of formal design control principles, codified in policies and procedures, aid with the development of product security requirements, ensure that product security requirements are appropriate for the device type, technology, use case, and use environment. It also ensures that product security requirements can be verified as successfully implemented.

Security requirements are based on user and market needs, laws and regulations, and security best practices, including recognized standards. The output—collectively referred to as product design inputs—leads to the formulation of design requirements. Security requirements are a subset of overall product requirements. It is typically useful to tag or indicate security requirements in some fashion to help reviewers easily identify security controls and to allow for traceability of these requirements through implementation and testing.

As a subset of product requirements, security requirements should be established based on the following:

- Legal and regulatory requirements specific for the security of technology or medical technology.

- Customer requirements and feedback relating to security, potentially in the form of user needs or laws/requirements with which the user is obligated to comply.
- Recognized security standards and best practices.30
- An organization's own security requirement baselines and practices.

These requirements should be assessed for applicability to a product during the Design & Development processes. In addition to the sources listed above, an initial set of baseline requirements and principles to prompt for potential requirements can found in Appendix C. The requirements definition process is often iterative, and updates will occur as security architecture is refined, including security practices such as threat modeling and security risk management. Any security control that is eventually deemed necessary to manage an identified security design risk should have an associated requirement.

Legal and regulatory requirements specific for the security of technology or medical technology.

Based on the intended target markets for the product and in consultation with personnel responsible for legal, regulatory, and privacy duties, regional security requirements that apply to the product should be identified. Depending on the target region where the product is intended to be sold or used, unique requirements should be accounted for from a regulatory compliance and customer acceptance perspective.

Regulatory compliance requirements can be driven both by medical device regulators (e.g., United States Food and Drug Administration) as well as laws defining regional requirements (e.g., data protection and residency, or specific localization such as EU's GDPR).

Customer requirements and feedback relating to security.

Customer acceptance requirements vary and depend on the functionality and risk of the product. These requirements can become more complex when talking about digital health products as third-party certifications (e.g., HITRUST, ISO 27001) of underlying infrastructure may become applicable and introduce additional considerations both in the design of the product and in the capabilities and processes that need to exist around it.

Recognized security standards and best practices.

Relevant recognized security standards and best practices should be considered for security requirement generation. A listing of relevant standards and practices is provided in Appendix G, with the entry for VII A ii being the most relevant for consultation.

An organization's own security requirement baselines and practices.

The development team should also reference internal security best practices and guidelines. For instance, an organization may also choose to create an internal standard on a broadly applicable topic like hardening an operating system image.

30 A listing of relevant standards and practices is provided in Appendix G, with the entry for 7.A.2 being the most relevant for consultation.

D.2. Secure Architecture and Design

Inputs:

- ISO IEC 81001-5-1 Health software and health IT systems safety, effectiveness and security – Part 5-1: Security – Activities in the product life cycle Ed. 1.0 Section 5.3 Software architectural design
- ISO 13485 7.3.4 Design and Development Outputs
- FDA Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions, Section V. B., Security Architecture, Appendix 2. Submission Documentation for Security Architecture Flows
- MDIC/MITRE Playbook for Threat Modeling Devices
- Security system and software requirements

Outputs:

- Security Use Cases
- Security Architecture Views
- Security Design Risk Assessment
- Architecture Reviews

As implementation progresses, specifics of system design become further clarified. Often this work results in various design diagrams, charts, and other useful reference material that developers and engineers—and regulators—can reference to understand the system, its components, and how they come together.

Ultimately, while diagrams and architectures are only representations of the actual implementation, they serve an essential purpose in communicating and ensuring both proper implementation as well as future supportability. These materials often accelerate the learning curve for new team members engaging with a product through both development and eventual support activities and are useful to other staff from other disciplines such as security engineers, regulatory, quality/safety, legal, and management.

The initial architecture and design of the product should consider the threats and countermeasures identified through threat modeling, along with the latest product requirements.

D.2.a. Threat Modeling

It is imperative that manufacturers leverage threat modeling to improve security aspects of device architecture and design, and to provide input(s) into a robust security risk assessment.

Threat modeling is the systematic and structured methodology of analyzing the security posture of a system, and it serves as one of the primary inputs to the security design risk assessment artifact.

Threat modeling:

- Enables the identification of threats and the identification, enumeration, and prioritization of vulnerabilities.
- Helps in providing an overview of how a system should function.
- Aids in delineating design choices and rationales.
- Facilitates addressing security risks with risk mitigations to achieve an acceptable security posture.

- Supports the identification of architectural or design security weaknesses that can be addressed in this early stage through architecture and design changes.

Threat modeling should involve the systematic identification of threat vectors and assets that may be most sought after by an unauthorized actor. It should also include a detailed analysis of the system, including but not limited to:

- processes
- data stores
- data flows
- trust boundaries
- intended and undesired interfaces with external entities (e.g., unauthorized actors)
- possible intended and undesired data flows between entities, data stores, and processes

This enables decomposition of the system and supports the identification of specific threat vectors and assets at risk, leading to an identification of the types of threats the system could be exposed to. Trust boundaries should be used to separate trustworthy elements of the system from other elements considered untrustworthy. The MDIC/MITRE Playbook for Threat Modeling Devices should be consulted for additional detail.

It is advisable for individuals who will be leading threat modeling efforts to undergo specialized training in this area.

Ultimately, threat modeling results in the identification of security architecture and design risks. These risks should be tracked in the security design risk assessment, along with any resulting architectural or design changes. Additionally, the assessment should include the implemented security controls that manage these identified risks. Traceability should be clearly established and documented from identified risks, through applied security controls and design implementation, and design verification testing.

D.3. Code Development and Testing

Input:

- ISO IEC 81001-5-1 Health software and health IT systems safety, effectiveness and security – Part 5-1: Security – Activities in the product life cycle Ed. 1.0 section 5.1 Software Dev Planning
- Secure coding standards (organizationally maintained and can include references to vetted public secure coding standards)
- NTIA Multistakeholder Process on Software Component Transparency document “Framing Software Component Transparency: Establishing a Common Software Bill of Materials (SBOM)”
- Secure Coding Techniques
- Security Related Software Requirements
- Security Related Software Designs

Output:

- Source code, system builds including any build scripts leveraged to assemble the system
- Source code reviews
- Unit tests (ideally as code)
- Security tool and automated unit testing results

- SBOM
- Other relevant security documentation

Systems that are secure from a requirement and design perspective can still be flawed due to vulnerabilities introduced by third-party components or during implementation or code development. Modern software development quality is not a guarantee, but achieved through the careful incorporation of developer guidelines, training, code reviews, and pervasive software testing. Each of these activities should also incorporate a security viewpoint to ensure the final system is appropriate for its intended use. See the Verification & Validation subsection for more complete descriptions of test types. For the purposes of this section, testing may be done informally and may aid development staff to identify and fix security issues prior to the start of formal Verification & Validation.

D.3.a. Iterative Code Testing

Security testing should be an activity that occurs throughout the development process and takes a variety of forms:[31]

- While code is being developed, code analysis should be performed during code check-in to proactively identify security vulnerabilities and insecure coding practices.
- Iteratively or upon successful completion of development, scanning for known vulnerabilities appropriate for the architecture in question should also be performed to ensure known vulnerabilities have been discovered and addressed and security controls selected to reduce the attack surface are sufficient and implemented correctly.

Organizations should apply secure coding standards during the development of software, and these standards should outline secure coding practices that apply in two ways:

- Standards that are generic to any programming language.
- Standards that are language specific.

Developers should be trained to secure coding standards. Examples of general developer guidelines relating to security that could be included in a secure coding standard are:

- Input validation
- Output sanitation
- Robust error handling
- Minimizing information leakage (error handling, non-essential development artifacts)
- Using proven implementations of security functions vs custom implementations whenever possible
- Logging of security relevant actions
- No hardcoded credentials
- Appropriate use of least privilege
- Code integrity protection

31 This includes, but is not limited to, the integration of security into code development, code scanning, and requirements definition.

Technology selection can also have an impact on the types of vulnerabilities possible in a software application. Organizations should carefully consider technology choices based on security implications associated with those technologies. Examples may include:

- Using memory safe languages as encouraged by CISA, when reasonable.32
- Selecting libraries that have ongoing support and a track record for timely software updates.
- Establishing SBOMs early, which can aid in a comprehensive assessment of appropriate libraries for use.
- Leveraging technology-specific secure development guidelines, such as use of compiler switches that augment security, avoiding object serialization, or use of parameterized queries for database operations.

Organizations should leverage code reviews to ensure security standards are applied. Code reviews can be performed in an automated or manual fashion, and they can be done on each software check-in or on specific software baselines. Generally, code reviews may be completed by peers (peer review) or by a team of reviewers.

The approach should be scaled in rigor to handle the complexity and associated risk of the code under review. To address high-risk code, organizations should conduct a design review. During design review, elements of the code with a high vulnerability to security threats are thoroughly examined, including the source code itself. Code reviews should be extended to explicitly consider security and to ensure proper application of secure coding techniques by reviewers with appropriate training in security. Code reviews should also ensure that implementation correctly reflects the documented product requirements and design, and that no additional interfaces or attack surfaces were added.

It is best practice to integrate Static Analysis Security Testing (SAST) into the code development pipeline, which can be effective at finding various classes of errors and can give repeatable and reliable results. SAST findings should be resolved as soon as possible but are required to be addressed prior to freezing code for Verification & Validation testing. Often, a final SAST report can be filed into the DHF as part of Verification & Validation, where developers are addressing findings as early in the development process as possible to avoid Verification & Validation findings and delays.

Finally, unit testing is the earliest level of testing in software applications, where individual building blocks of a program are tested one at a time to ensure proper functionality with a high degree of control and rigor. Unit testing is a critical opportunity to perform functional testing on security primitives such as encryption or authentication, and automated unit testing can be effective in ensuring secure functionality is not broken by code changes during development or over the product lifetime. It is also an ideal time to verify the robustness of functional interfaces for concerns such as comprehensive input validation, output sanitation, and error handling. Unit testing can be done by development teams themselves or by independent teams, and unit tests should be reviewed to ensure complete coverage, paying special attention to testing of security critical parts of the system.

D.3.b. Iterative Integration Testing

Organizations should perform iterative testing of integrated components (third party or developed) to ensure that the components are tested at various integration stages. Integration testing involves building running versions of all

32 https://www.cisa.gov/news-events/news/building-blocks-digital-world-coding-must-be-memory-safe-and-secure

components and their interfaces. Ideally, the build and test processes are automated into continuous integration pipelines, but less automated methods are also acceptable as long as the test system consists of the latest versions of developed components. Test plans can be consulted for specific tests that should be run iteratively throughout development and may include the following:

- Host vulnerability scanning, using tools like Nessus, to identify known vulnerabilities and insecure configurations
- Dynamic Application Scanning Testing (DAST)
- Malformed Input Testing (e.g., Fuzz testing)

D.4. Patch/Software Update Planning

Successful implementation of TPLC cybersecurity considerations require organizations to plan, incorporate, and resource processes used to maintain system resilience such that systems can continue to be safe and effective through their entire lifecycle. During Design & Development it is essential that any postmarket management and maintenance features that have been specified during requirements development are built into the product.

Patch/software update planning is often a significant part of the overall maintenance plan, but there can be other activities in maintenance plans such as refreshing cryptographic certificates (e.g.,SSL certificates) or re-signing/pushing software to app stores. Systems that include software require regular updates to ensure the technical and security debt of the system does not become a barrier to keeping the system safe, secure, and effective.33

This section will highlight key considerations when planning for patch/software updates to ensure the TPLC of the product is adequately maintained through planned end of support, and beyond to meet regulatory requirements. Resulting processes will be leveraged during the maintenance phase to ensure TPLC cybersecurity considerations are adequately managed for on-market products.

Potential inputs:

- Secure architecture design
- SBOM
- Threat model and resulting Security Risk Assessment
- Product Lifecycle Policy
- Contractual obligations from existing contracts or the HSCC Model Contract Language for Medtech Cybersecurity, specifically on patching timelines and incident response
- AAMI TIR97:2019 Principles for medical device security – Postmarket risk management for device manufacturers
- ISO IEC 81001-5-1 Health software and health IT systems safety, effectiveness and security – Part 5-1: Security – Activities in the product life cycle Ed. 1.0
- FDA Postmarket Management of Cybersecurity in Medical Devices

33 AAMI TIR 97 provides further details and recommendations related to product update planning.

- IMDRF Principles and Practices for Medical Device Cybersecurity
- MDCG 2019-1 Rev 1 Guidance on cybersecurity for Medical Devices
- ISO 29147 Information Technology – Security techniques - Vulnerability Disclosure
- ISO 30111 Information Technology – Security techniques – Vulnerability handling process
- MITRE Rubric for Applying CVSS to Medical Devices
- CISA Stakeholder-Specific Vulnerability Categorization Guide (Nov 2022)34
- UL 5500 Standard for Safety for Remote Software Updates
- Enhancing Vulnerability Prioritization: Data-Driven Exploit Predictions with Community-Driven Insights35

Potential outputs:

- Maintenance Plan
- Cybersecurity Signal Monitoring Procedure
- Coordinated Vulnerability Disclosure Policy
- Product Security Vulnerability Triage Procedure
- Product Security Incident Response Team (PSIRT) Procedure

Security by Design and TPLC planning must include adequate consideration for how a product will be maintained in the market within the designated timeframe until end of support is determined, communicated, and executed, as well as additional regulatory requirements that may extend past declared end of support, such as complaint handling, adverse event and problem reporting (i.e., 21 CFR Part 803 Medical Device Reporting) and recalls (i.e., 21 CFR Part 806 Correction and Removals).

The following topics should be addressed as part of TPLC planning.

Update Cycle (Standard and Critical) Planning

Given the increasing reliance on software, manufacturers should account for known anomalies, vulnerabilities, and defects in software, and they should plan postmarket activities thoughtfully to ensure products are safe and maintain essential performance, as well as to protect the environments within which they operate.

Not all known vulnerabilities are created equal and performing emergency actions to immediately fix all known vulnerabilities identified in software would result in an unsustainable practice. Instead, Manufacturers must determine an effective way to prioritize vulnerabilities that require immediate action, and then establish reasonable routine software update cycles to catch up on most vulnerabilities.

Given regulatory requirements and patient safety concerns, manufacturers and their customers may not be able to apply third party component updates as they become available. In certain cases, Verification & Validation activities must be performed prior to the application of the update to ensure the device maintains safety and essential

[34] https://www.cisa.gov/sites/default/files/publications/cisa-ssvc-guide%20508c.pdf
[35] https://arxiv.org/pdf/2302.14172.pdf

performance prior to update application.36 As this activity can be extensive, organizations should ensure resources are planned and available to perform update-related tasks on a rational cycle, based on key device characteristics such as the chosen third-party components, and how effectively threat surface reduction has been accomplished in design.

The deployment of software updates that are verified and validated by the manufacturer (as opposed to third party) are also non-trivial and may have significant cost in terms of resources and may require specific considerations to minimize downtime required for the update by the operator.

Prior to regulatory submission, a Postmarket Vulnerability Management Plan should be developed by manufacturers to demonstrate how the manufacturer intends to sustain an appropriate level of security throughout the TPLC, as well as how cybersecurity can be maintained for the product post-release.

Successful development and implementation of a Postmarket Vulnerability Management Plan ensures that vulnerabilities are evaluated and helps organizations to prioritize and understand potential applicability and impact. The principles for decisions and actions taken should be well documented and integrated in the development process. Further, it can provide guidance on how vulnerabilities that are commonly addressed through routine patches can be added to a sustaining update or patching cycle, while vulnerabilities that are determined to have potential for significant impact can be further assessed for patching outside the routine patching cycle.

In addition to process considerations, the device design should support future maintenance needs, such as resource-efficient or automated patching and code signing for patch security, among other characteristics of an effective patching process.

Because the potential for the existence of a future exploit for a given vulnerability cannot be anticipated, manufacturers should leverage existing known methods to further define potential risk, such as through threat modeling, and they should prioritize potential for patient harm as determining factors, as additional actions may be required.

There are two key designations that need to be designed into the product during development to support postmarket needs to be discussed in terms of planning maintenance lifecycles.37

- Standard update: The planned maintenance activity, including updates to address known vulnerabilities, on a reasonably justified regular cycle where justification appropriately reflects the system's architecture and associated risks over time.
- Critical update: An out of cycle software update to address an uncontrolled risk or critical security vulnerability.

36 Organizations should review the [Summary of FDA guidance on patching] on Page 59 of the HIC-MaLTS for more information.

37 Designations are consistent with updates to the FD&C Act, specifically 21 USC 360n-2(b)(2)(A) 21 USC 360n-2: Ensuring cybersecurity of devices (house.gov)

Most maintenance activities, including addressing lower risk software vulnerabilities, should fit within the standard update cycle for a properly designed device. A critical update, requiring pathing outside the standard update cycle, should only occur when the standard update cycle is insufficient to address a clear and present threat.38

The essential planning required for a medical device is to design the targeted standard software update cycle. The frequency will vary by device, but the objective is to ensure security debt is effectively managed to ensure the benefit of the device continues to outweigh the risk to the device and its operating environment. The resilience of a particular device's security controls, such as disabling unnecessary services that may be vulnerable, could help justify a longer frequency. If a specific cybersecurity vulnerability will cause an imminent threat before the standard update cycle would complete, it should be treated as a critical update.

Figure 6 below illustrates this concept, and notes that at times a specific cybersecurity vulnerability may pose a risk that requires an accelerated critical update. In addition to the accumulating security risk, it should be noted that the amount of work to bring a system to a patchable level also increases if updates are not regularly performed. For instance, a specific security patch may require many other updates to be on the system to successfully install or operate.

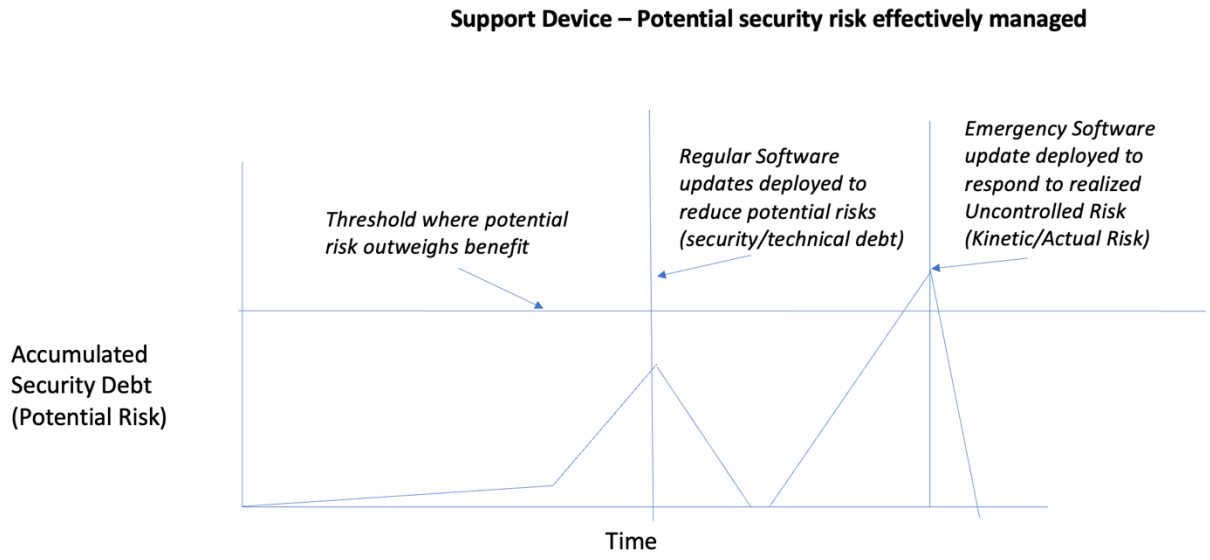

Figure 6. Security Debt Over Time Managed With Standard Update

The mechanism to deploy updates should similarly reflect the velocity required to support the standard update cycle frequency and desired target timeline for critical updates. In addition to frequencies, the design of the device and

38 It should be noted that the longer the standard update cycle, the more likely a critical update may be necessary to protect safety and security of a product and the environments within which it operates.

supporting customer support documentation should indicate roles and responsibilities in the update process, including planned update frequencies and how organizations will communicate that updates are available.

Table 3 suggests standard update frequency and critical update timeline capabilities based on different device types.39 The suggested timeframes are a starting point, and it is recommended that deviations from these suggested norms be adequately supported with rationale that is shared with regulators and customers in customer security documentation.

Device Type	Example Device(s)	Suggested Reasonably Justified Regular Cycle Frequency	Target Timeline for Critical Updates*	Rationale
Standard Windows Build	Surgical Planning software Image viewing software	N/A	N/A	Operates like a standard Windows system but requires validation to ensure essential performance. Maintained by hospital staff according to IT or workstation policies, software should be resilient to updates occurring through healthcare delivery staff.
Kitted Windows – Manufacturer Supplied Hardware	Console for monitoring medical devices	Quarterly	30 days	Manufacturer provides hardware and updates to the operating system to ensure integration with a related medical device operate correctly.
Windows-based, highly customized (embedded and/or highly integrated essential performance)	Surgical robot console bedside patient monitor	Bi-annual w/in 1 quarter of a consolidated windows update (released twice annually)	60 days	Directly operates a medical device and has high availability/reliability requirements, validation is essential to maintain safety and essential performance.

39 These suggested standard update frequencies are starting points that have been deemed reasonable by the manufacturers and HDOs working on this document.

Mobile device, commercial build	Programmer Insulin pump controller (mobile app)	Bi-annual	30 days	Manufacturers should understand the mobile device OS planned lifecycle and plan to keep pace with updates as part of their postmarket operations.
Mobile device, highly customized	In-home cardiac monitor	Major releases from the supplier	60 days	Manufacturer has reduced attack surface and can apply updates that are critical when needed. Ability to precisely apply necessary patches is a needed capability through the supported lifetime of the device.
RTOS, highly customized	Surgical robotics system	Major releases from RTOS supplier	60 days	Manufacturer has reduced attack surface and can apply updates that are critical when needed. Ability to precisely apply necessary patches is a needed capability through the supported lifetime of the device.
Embedded firmware	Surgical tool Sensor that is part of a system such as a CGM	Highly variable based on device components, possibly only on major releases from MDM	60 days	Manufacturer has reduced attack surface and can apply updates that are critical when needed. Ability to precisely apply necessary patches, or replacement when devices may be short term use including recall of unused devices, is a needed capability through the supported lifetime of the device.
Embedded firmware, implanted	Pacemaker Deep Brain Stimulator	None	Months	Manufacturer has designed the device for maximum reliability for the entire TPLC of the implanted device. Selected components are designed specifically for the long lifecycles and attack surface is minimized and fails safe. Updates are possible but given the real time life supporting capabilities are rare and coordinated with HCPs.

Cloud-based Software Device Function	HCP alert processing, AI algorithms analyzing images or sensor data for medical relevant observations	Quarterly	Weeks	Manufacturer has responsibility for the entire environment. Threat Modeling should identify which areas of the Cloud the Manufacturer can update and which areas are under the control of Cloud providers to update. For portions of the Cloud which the manufacturer can update, they should be able to validate and deploy updates quickly. For portions of the Cloud which the manufacturer cannot update, mitigating controls such as agreements with Cloud providers should be established to mitigate associated risks.

Table 3: Suggested Starting Points for Standard Update Cycle Planning

**Based on FDA Postmarket Management of Cybersecurity in Medical Devices "uncontrolled risks" and suggested timeframes*

EOL/EOS planning

In addition to planning the update lifecycle for product maintenance, manufacturers also need to assist customers in planning the useful lifetime of their products and the required replacement lifecycles. Products that do not plan for EOS can result in significant risks that may not have clear solutions to manage. For instance, if a product depends on a third party to provide operating system updates, and those updates are no longer available for the version the product depends on, a situation may occur where a cybersecurity risk cannot be effectively addressed in the field.

It is essential that EOL and EOS be a planned activity for devices as part of product Design & Development. If choices are made during product Design & Development that insufficiently account for future supportability challenges, cybersecurity risks may cause unacceptable risks to the product's safety and essential performance as well as the environments in which they operate. Figure 7 illustrates how accumulating debt can result in unacceptable risk over time.

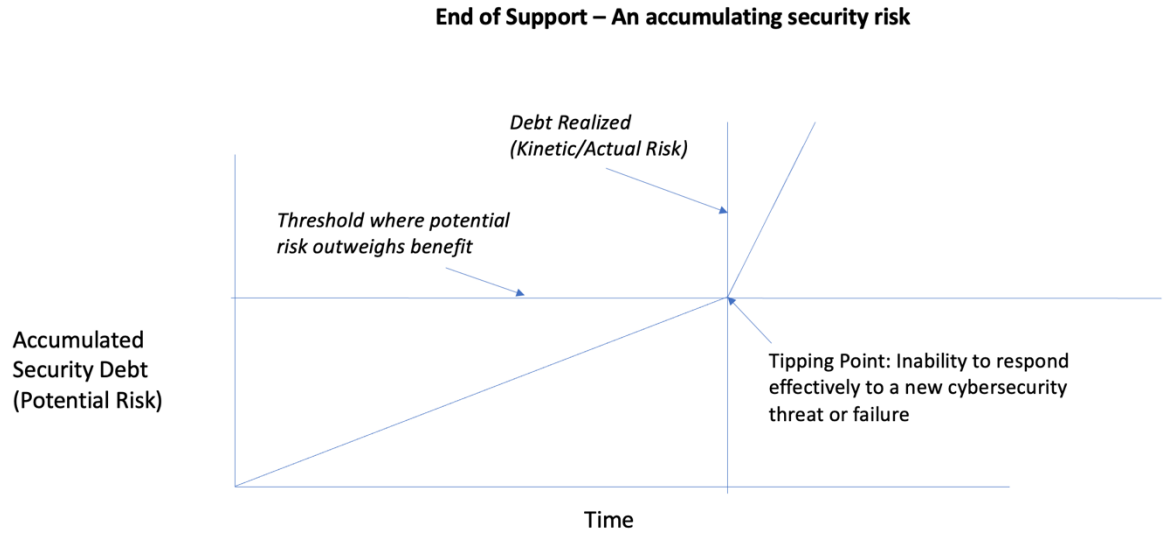

Figure 7. End of Support – An Accumulating Cybersecurity Risk

It should be noted that the EOL/EOS dates for product components need to be closely monitored and considered in managing a product's potential useful lifetime.

The following general principles are suggested when planning and communicating product EOL/EOS:40

- EOL should be communicated prior to when the manufacturer plans to stop selling a product. The expected period of support should match the expected lifespan for a device from the last date of sale.
- EOS should, at a minimum, cover the expected lifetime of a device from the last date it was sold (EOL date). There are limiting factors that may reduce support capabilities such as EOS components, but manufacturers are expected to be accountable to address cybersecurity risks through the expected lifetime from the last sale date.
- Communication of EOL and EOS should occur with sufficient notice to allow downstream customers adequate notice to plan and manage any cybersecurity risks that may result. Methods for communication will vary, but minimally EOL and EOS information should be present in customer support documentation and should be communicated through normal customer communication channels leveraged by the product.

Planning for Cybersecurity Signal Monitoring

40 See the HSCC HIC-MaLTS and IMDRF Medical Device Legacy best practices document for more information to help inform the management of EOL and EOS. The HSCC HIC-MaLTS also provides a significant amount of information on how manufacturers can avoid premature, "unexpected legacy" status where a product cannot tolerate existing threats presented in the environments in which the product operates.

Manufacturers should have a cybersecurity signal monitoring process to identify and triage cybersecurity signals effectively.41 This process should be designed to enable prioritization of resources in response to standard (planned) or critical (unplanned) updates.

The following sections detail key considerations for planning postmarket cybersecurity signal monitoring and response.

Potential Sources for Cybersecurity Signal Monitoring

Examples of potential sources for cybersecurity signal monitoring include, but are not limited to:

- Complaints: Ensure complaint handling processes have criteria to identify and categorize potential cybersecurity issues along with escalation paths that engage key personnel qualified to assess and respond.
- Coordinated Vulnerability Disclosure42 Process: Companies should have a method for security researchers and other relevant parties to report potential security vulnerabilities.
- Product Anomalies: Security resources should be integrated into the assessment of product anomalies to ensure that potential security vulnerabilities are considered when product issues are identified.
- Supplier Monitoring: Key suppliers should be identified and monitored for vulnerability disclosures originating from both the supplier and public databases both from the third party and in any public database such as the US National Vulnerability Database maintained by NIST.
- ISAOs: Information Sharing and Analysis Organizations (ISAOs) like Health-ISAC should be monitored for new threats to the healthcare sector, including those that may affect in-scope products.
- Retesting: Periodic vulnerability testing and penetration testing should be performed in a relevant fashion for the product in question to identify new vulnerabilities.
- Logs/Alerts: Signals originating from the product in the form of logs or alerts should be monitored for potential indications of exploited vulnerabilities.

Triage and Assessment of Cybersecurity Signals

Factors that organizations should consider related to triage and assessment of cybersecurity signals include, but are not limited to:

- Roles and Responsibilities: Roles and responsibilities for collection, triage, and assessment of potential cybersecurity signals should be clearly defined and resourced appropriately.
- Process: Initial triage criteria and response timing expectations should be documented, including:
- Triage: Initial triage should be completed quickly (measured in hours or days) to confirm status in the product and give an initial rating that determines assessment and remediation timelines.
- Security and Safety Assessment: Triaged vulnerabilities confirmed should be assessed for both security and safety impacts. Existing anomaly and impact assessment processes should be leveraged.
- Remediation Plans:

[41] See the HIC-MaLTS Patching Lifecycle section for additional discussion and recommendations.
[42] ISO 29147 Information Technology – Security techniques - Vulnerability Disclosure

- Standard Updates: Cybersecurity vulnerabilities that do not rise to the level of a critical update should be remediated in the next standard update cycle.
- Critical Updates: Out of cycle updates are often escalated for focused response, sometimes leveraging an incident management process, to ensure cross functional coordination of resources necessary to drive necessary remediation activities.

- Records: Integration with existing anomaly and risk management processes including regulatory notifications. Confirmed vulnerabilities are often tracked in existing anomaly and risk management processes.

Coordinated Vulnerability Disclosure and Information Sharing

Coordinated Vulnerability Disclosure (CVD) is a process or set of processes that allows an entity to communicate details about a security risk associated with a device or technology. A mature and comprehensive CVD program is meant to promote transparency about known device vulnerabilities and provide details on how to manage, mitigate, and/or fix the issue to reduce risk to an acceptable level to ensure the device remains safe and effective. CVD is recognized internationally as a critical component of device cybersecurity risk management programs and is required by law in the United States for certain types of medical devices. Organizations should have a coordinated vulnerability disclosure policy that is publicly shared and accompanying procedure that details how vulnerabilities are identified by the organization or received from external reporters along with the process of how these vulnerabilities are triaged, acknowledged, analyzed, mitigated and communicated publicly in a timely fashion.

D.5. Secure Transfer to Manufacturing Planning

Potential inputs:

- Secure Architecture Design artifacts
- SBOM (raw from development release)
- ISO IEC 81001-5-1 Health software and health IT systems safety, effectiveness and security – Part 5-1: Security – Activities in the product life cycle Ed. 1.0 section 5.8
- Design transfer to manufacturing process

Potential outputs:

- Release Management Process Design
- Release Management Procedure
- Customer security documentation, including SBOM (release version)

Release Management - Design Transfer to Production/Manufacturing

The product release process includes design transfer to production/manufacturing and involves the process to ensure documentation is complete and the specific procedural activities to repeatable and secure deployment of artifacts used to build the product including code or manufacturing specifications. Specifically, this process needs to consider aspects and define requirements for the design transfer to reduce the risk of security compromise to an acceptable level. This section details documentation considerations, including processes that should be planned for the design transfer process. The design transfer happens after all Verification & Validation activities are completed and the design has been formally approved for transfer. Planning for a smooth design transfer is essential to ensure

the product's state of security at release is accurately captured, the release process is adequately secured, and adequate resources for maintenance activities are in place.

Documentation should include the following:

- Residual Risk Acceptance: Residual risks will exist in a product to be released. To ensure that these risks are understood and monitored, a formal risk acceptance process should be followed. At a minimum, this process should include a definition of the underlying vulnerability, any threat(s) that may exploit it, rationale as to why the risk is being accepted, the period of time (as applicable) that the risk is accepted without requiring reapproval, and the signature/approval of the senior leadership individual that is accepting the risk.
- Risk Transfer Decisions: The product security management plan should be cross-functionally reviewed and approved by business leadership in the organization. Components of this plan necessary for operation and management of product security should be provided to customers in customer security documentation, user manuals, and contractual agreements between the manufacturer and customer.
- DHF Documentation Completion Review: A quality review to ensure all committed and required DHF documents are complete and appropriately approved/stored, including any required operational procedures, should be performed.
- Device Master Record (DMR) Management Process: A process should be in place to associate security information specific to a device with its DMR.
- Regulatory Considerations: A regulatory review to ensure the release has met any required regulatory notifications and/or premarket approvals should occur.
- Operational Support: Responsibilities for postmarket maintenance work should be defined, resourced, and accepted.
- Build Process:
    - Regional and environmental configurations for target deployments, such as country-specific markets, should be documented and applied.
    - Build sources and environment(s) should be appropriately documented to ensure repeatable and consistent builds with full integrity.
    - Build output(s) should be cryptographically signed for downstream verification.
- Transfer to Manufacturing for New Devices: The specific steps to securely distribute updates to manufacturing and/or product environments should be documented.
- Installation/Deployment on Fielded Devices: The specific process and plan to deploy updates on existing devices in the field and on the shelf should be documented.
- Securing Cryptographic Materials: Methods should be documented that establish that cryptographic material(s) installed in each device are not compromised during the manufacturing process.

Customer Security Documentation

For any commercialized product, it is critical that the manufacturer develop and maintain documentation that describes all pertinent security information. Furthermore, customer security documentation should be updated when significant changes occur to existing or new product versions. This documentation should be prepared for external distribution and consumption by customers during product procurement to perform risk assessments.

Customer security documentation provided by manufacturers should include:

- Description of secure configuration(s).
- Information on secure device operation, including logs, maintenance, and patching/updates.
- SBOM
- EOL/EOS information on the product and any included components
- Information on secure decommissioning.
- Data flow diagrams that capture items flowing in and out of the device, open network ports and active services, as well as any requirements for network connectivity.43
- Remote access methods and tools, if used.
- Access control design, including privileged access controls and manufacturer maintenance and/or service accounts.
- Comprehensive description(s) of the control measures implemented.
- The patch management plan developed by the manufacturer, identifying any customer responsibility as part of the plan.
- The required cybersecurity controls.
- Logging and audit capabilities to support customer security operations.
- Summary of known security risks and considerations, including unmitigated findings from penetration testing.
- Contact information for the manufacturer to report incidents, vulnerabilities, or for general inquiries regarding security.
- Manufacturer Disclosure Statement for Medical Device Security (MDS2)44

For context regarding what may be included in customer security documentation and what it might look like, see Appendix D.

E. Verification & Validation

This section decomposes cybersecurity testing into the discrete activities that may occur during the Verification & Validation stage of Design & Development. The described activities may be executed throughout development: iteratively during early development stages, comprehensively during later development stages, and formally during final release. A specific emphasis is placed on separating diverse testing activities based on their objective, involved tools, skillsets, and discrete outcomes.

An organization's ability to possess a broad spectrum of testing activities is key to employing cybersecurity testing during various stages of development when various components require specific levels of evaluation. Formally

43 See Appendix 2 of the 2023 FDA Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions.

44 https://www.nema.org/standards/view/manufacturer-disclosure-statement-for-medical-device-security

defining the purpose and output of each activity further enables the creation of mature cybersecurity engagement processes and automation. With a proper engagement process to prescribe and govern the associated mitigations, cybersecurity testing may be performed at different product stages and across product lines to deliver a comprehensive set of technical cybersecurity risks.

E.1. Verify Security Controls

Potential inputs:

- Security Risk Assessment Report
- Security Management Plan
- Security Test Plan
- Security Unit Test Scripts
- Security Controls

Potential outputs:

- Security Risk Traceability Matrix (may be included as a subset of overall traceability, but must trace to identified controls identified during security risk assessment)
- Security Test Execution Records (may be included as a subset of overall test execution records)
- Security Software Anomalies (may be included as a subset of overall anomalies)
- Security Test Report (may be included as a subset to overall test report)

Security requirements, like functional requirements, are described through system and software level detailed requirements and expressed to the development team for implementation. Due to the strong resemblance in how cybersecurity and functional requirements are described and implemented, it's essential to adopt a similar approach to ensure their correct implementation through unit testing.

This approach includes the following types of testing:

- Unit testing that includes security requirements
- Integration testing that is inclusive of security requirements

These tests must be traced to the requirements they verify. Ideally these tests are automated wherever possible and can be included in regression test suites that could be leveraged to verify requirements after any change. When automation is not possible, clear test scripts enabling manual verification should be created to ensure consistent and accurate verification of the requirements. Output from security verification testing should include:

- Security Risk Traceability Matrix (may be included as a subset of overall traceability, but must trace to identified controls identified during security risk assessment)
- Security Test Execution Records (may be included as a subset of overall test execution records)
- Security Software Anomalies (may be included as a subset of overall anomalies)
- Security Test Report (may be included as a subset to overall test report)

E.2. Identify Known Vulnerabilities

Potential inputs:

- Security Management Plan
- CISA's Known Exploited Vulnerability list
- National Vulnerability Database (NVD)

Potential outputs:

- Vulnerability test records
- Software anomalies (may be included as a subset of overall anomalies)
- Test reports (records to demonstrate the evaluate of the product for known vulnerabilities)

Identification of known vulnerabilities is focused on quickly and consistently identifying known cybersecurity gaps using primarily automated and updateable tools. Vulnerability testing can be performed at high velocity and due to the automated nature of the activities, can often be integrated in an automated fashion into pre- and postmarket processes.

Every manufacturer is expected to possess the ability to identify known vulnerabilities in their product at any point in time. Due to the fast-moving nature of the cybersecurity threat landscape and associated vulnerability data, knowledge of current exposures is a point-in-time state. The security posture of a product may change significantly in response to newly discovered vulnerability information.

Mature product security programs have processes and methodologies for identifying vulnerabilities across various components through a variety of activities:

- Managing known, reported vulnerabilities: Known, reported vulnerabilities are associated with a specific version of the software/hardware module and are typically—but not always—reported to NVD by suppliers and researchers, among others.
- Identifying secure coding violations**:** Secure coding violations are logical coding omissions (e.g., unparametrized SQL queries, executing user input via OS/platform command line function, weak encryption routines). These flaws can be identified through automated tools.
- Identifying insecure configurations: Components can be misconfigured, creating unintentional vulnerabilities that can be easily exploited.
    - Exposing interfaces to untrusted networks
    - Retaining unused functionality
    - Retaining unneeded development functionality
    - Using improper authentication/authorization of accessible resources

See Appendix H for more details on vulnerability scanning tools and techniques.

Automated vulnerability and exposure identification can lead to a variety of initial findings, with severity and exploitability ratings that need to be evaluated within the full context of the product and the product's planned or implemented security controls. These evaluations may then be used to determine if the subset of the component where the vulnerability or exposure exists is used by the product or accessible by a threat actor. A revised rating may be appropriate if the evaluation finds that the severity or exploitability is different within the context of the product.

E.3. Security Validation Testing

Potential inputs:

- Architecture/Design Documentation
- Security Design Risk Assessment

Potential outputs:

- Raw penetration test report
- Manufacturer penetration test disposition report
- Anomalies resulting from penetration test findings

All security requirements should be subject to Verification & Validation testing, alongside the device's functional requirements. The validation testing should focus on how users interact with security controls and how a cyber adversary (hacker) may exploit vulnerabilities. The purpose is to ensure that clinical functionality is not made ineffective by the implemented controls and that device safety and efficacy cannot be compromised by an attacker.

Penetration Testing

Penetration testing45 should be conducted to identify security weaknesses in the product that could be exploited to cause the device to function outside of its intended use, or to cause patient harm. Penetration testing can be part of validation testing, in that it can help validate that certain actions that could otherwise cause a safety or effectiveness risk are unlikely to occur.

Testing can be performed at three different levels based on access and information provided to the testers.

- A "closed box" or "black box" assessment provides minimal information (e.g., IP address of the target on the network).
- An "open box" or "white box" assessment provides a large amount of information (e.g., IP address, source code, customer documentation, confidential information).
- A "balanced box" or "grey box" assessment provides enough information to expedite the test, assuming with enough time and patience the attacker would eventually obtain the provided information.

Elements to consider when conducting penetration testing include duration, personnel availability and resources, tool set, scope, test environment, and the vulnerability landscape of the system. Figure 8 outlines the three phases of a targeted penetration test, including actions to complete each step:

45 Penetration testing is a deep and targeted assessment conducted with a specific focus on areas of the product that the assessment entity and associated team believe are likely to possess a vulnerability or exposure that may lead to risk. Identified products are assessed with a specific objective over a fixed period of time, and the assessment concludes with a report indicating identified vulnerabilities and exposures that were leveraged to advance or obtain the objective. The quality of a penetration test is a product of the testing scope, the time allocated, and the skill of the involved assessment team.

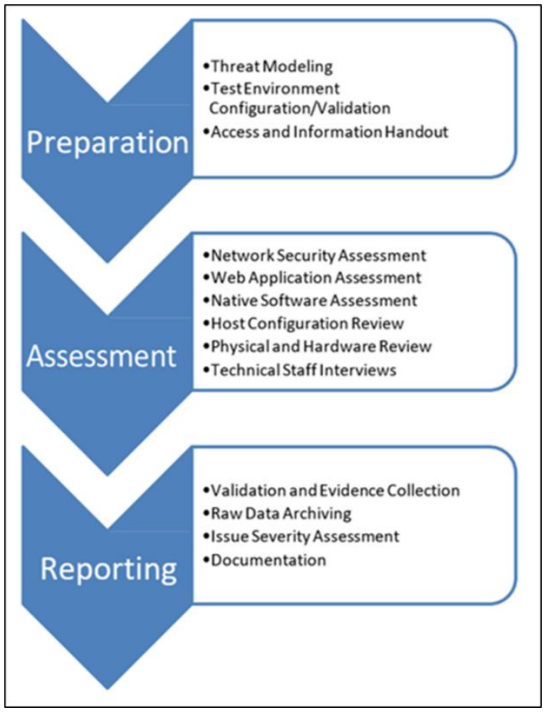

Figure 8. Three Phases of a Penetration Test

A key element of a successful penetration testing is the skill level of the staff performing the test. Tools alone are insufficient to perform adequate security testing. Testers need system thinking, network knowledge, coding skills, and the ability to think like a threat actor.

Penetration testing is a deep and targeted assessment conducted with a specific focus on areas of the target that the assessment entity and associated team believe are likely to possess a vulnerability or exposure that would advance their objective. Identified targets are assessed with a specific objective over a fixed period of time, and the assessment concludes with a report indicating identified vulnerabilities and exposures that were leveraged to advance or obtain the objective. The quality of a penetration test is a product of the testing scope, the time allocated, and the skill of the involved assessment team.

Traditional penetration testing provides specific insights and real-world attack simulation across an objective with focus and issues reported based on severity.

Manual deep-dive testing, in which attempts to bypass security controls iteratively and vulnerabilities are intelligently chained, is the hallmark of a quality penetration test and key to obtaining and demonstrating the highest impact possible across the target scope. Penetration testing employs human intuition and experience to:

- Make an informed decision regarding where weaknesses may reside. Decision-making may be informed by the feedback of robustness or resiliency testing.
- Intelligently chain seemingly low-risk issues to create a high-risk issue.
- Identify and exploit logical flaws that require the context of multiple aspects of the target.
- Intelligently overcome security controls or iteratively attacks in response to a failure.

Objective-based penetration testing begins with defining objectives; what information, key components, processes, or functionality are we trying to protect? What threats are we trying to simulate? Typical objectives include obtaining access to high-security components, impacting key clinical functions, access to sensitive information, or control over a target platform. The assessment team will develop a plan to reach the objective through any possible attack technique to locate the weakest links and exploit them in concert.

A penetration testing methodology is expected to be documented to address how the penetration test will marshal testing activities around a specific objective or set of objectives and consistently evaluate any target scope regardless of the individuals performing the assessment.

Robustness Testing

Robustness testing focuses on the breadth of cybersecurity controls for specific components or the entire target. A robust target or component is one that can resist a broad range of potential threats.

In contrast to penetration testing, robustness testing includes a specifically cataloged set of technical test cases that are executed against a product to identify vulnerabilities and exposures relative to a known framework of tests. The output of robustness testing qualifies the target component to a level of cybersecurity contextualized by the test cases executed and the protocol for which each test was conducted. An applicable framework may be employed that provides technical test cases for robustness testing of each component (e.g., OWASP for web applications).

Resiliency Testing

Resiliency testing assesses how a system may sustain damage when threats exceed its robustness and evaluates if it can recover. A resilient system invests in the ability to recover, change, and adapt to meet unanticipated threats. In medical device cybersecurity, targeted resiliency testing focuses on maintaining system availability and integrity during a DoS attack. Resiliency testing methods must be validated to ensure the adequacy of depth and format. Specific types of resiliency tests performed are selected based on dataflows, interfaces, and related functionality viewed as input for manipulation. A key target of resiliency testing is often proprietary data parsers and protocols.

See Section D.5: Secure Transfer to Manufacturing Planning for information on Design & Development activities that are done once Verification & Validation is completed to facilitate release of the product.

F. Maintenance

Maintenance of connected and software-based products is an essential TPLC activity. Tasks can include regular required activities such as refreshing pinned SSL certificates in a mobile application, but also must include essential capabilities to maintain the security of products over their lifecycle. For purposes of the JSP, maintenance is broken down into the component parts. The processes leveraged in this section should have been planned and developed as part of product development. During the maintenance phase, these processes should generate records that demonstrate that TPLC activities are being effectively executed to ensure maintenance of products. Each of these topics is further defined and described in the following sections.

- Surveillance
- Vulnerability & EOL/EOS Management

- Patch/Software Update and Deployment
- Security Incident Response
- Customer Security Communication

F.1. Surveillance

Potential inputs:

- Maintenance Plan
- Cybersecurity Signal Monitoring Procedure

Potential outputs:

- Records generated that show triage and potentially escalation of vulnerability signals relevant to the product

Procedures for surveillance including monitored sources for cybersecurity signals, as detailed in a maintenance plan or cybersecurity signal monitoring procedure, should be staffed and executed. These sources should include externally reported vulnerabilities via a Coordinated Vulnerability Disclosure process, threat intelligence sources such as ISAOs and relevant software supplier known vulnerability communications. Monitoring performance of surveillance with leadership reviews is recommended. Once a potential vulnerability is identified, it should be assessed, verified, remediated, and communicated as appropriate. Records should demonstrate how potential vulnerabilities are identified and triaged leveraging documented procedures in the Maintenance Plan or Vulnerability Triage Procedure.

F.2. Vulnerability & EOL/EOS Management

Potential inputs:

- Maintenance Plan
- Cybersecurity Signal Monitoring Procedure
- Vulnerability Triage Procedure
- Supplier lifecycle policy and EOL/EOS communications

Potential outputs:

- Customer notices indicating changes in a product's EOL/EOS dates, including recommended mitigations for safe continued use

Vulnerability management46 describes a set of processes to intake, assess, remediate, release, and communicate about vulnerabilities for the purpose of reducing risk to an acceptable level in order to ensure the device remains safe and secure to provide it's intended clinical use. Organizations should have a Cybersecurity Signal Monitoring Procedure in place that identifies sources that will be monitored for vulnerabilities. Similarly, organizations should have a maintenance plan and/or vulnerability triage procedure that can be leveraged to process identified

46 ISO 30111:2019

vulnerabilities or significant changes to the support (EOL/EOS) of included software components. The major phases of this capability include:

- Receipt and Intake
- Verification and Triage
- Remediation Development
- Release

Potential vulnerability evaluation or investigation by the manufacturer should include steps to determine if a vulnerability constitutes a product-related cybersecurity vulnerability or incident.

- A cross-functional team should be assembled to ensure a coordinated investigation and appropriate response.
- Specifically, the investigation should include close coordination with the affected customers or stakeholders and appropriate clinical parties.
- The manufacturer should ensure effective escalation and triage by having adequate procedures and classification for potential cybersecurity issues for handling by key support personnel to assess the issue, identify mitigation, communicate the matter to affected customers or stakeholders, and follow up, as needed.
- Customers and suppliers should perform information sharing during an investigation to support meeting FDA Postmarket Management of Cybersecurity in Medical Devices recommendations.

For commercialized products, security risk assessment and remediation planning are performed as part of a postmarket management process.

- Low risks may be addressed separately in a reasonable amount of time, but at minimum during the next product or software update.
- Medium to critical risks, which may align with uncontrolled risks per FDA Postmarket Management of Cybersecurity in Medical Devices Guidance, include communicating with the customer and user community about the vulnerability.
    - 30 days: communication to customers identifying the devices which could be impacted and providing interim control measures to mitigate risk and a remediation plan within 30 days of learning of the vulnerability.
    - 60 days: Remediation and control measures, including patches, must be available with at least one of the deployment methods promptly and within a maximum of 60 days after learning of the vulnerability. As soon as possible but no later than 60 days after learning of the vulnerability, the manufacturer should fix the vulnerability, validate the change, and distribute or ensure availability for the deployable fix to its customers and user community such that the residual risk is brought down to an acceptable level.

All disposition decisions should be captured in appropriate records such as software anomalies or periodic disposition reports. Where applicable, Corrective and Preventive Action (CAPA) plans may need to be established in compliance with manufacturer CAPA policy/procedures to evaluate the need to correct existing or potential quality issues that impact the security or safety of products and to develop actions to prevent their occurrence or recurrence.

Risks resulting in unauthorized disclosure of PHI or PII will require data breach investigation and potential notification to customers in accordance with local laws and regulations. Other sensitive information and data such as intellectual property will require data breach investigation and potential notification to stakeholders. Any potential reported patient harm or safety risk should also be investigated and potential notification to regulars should be done in accordance with local laws and regulations.

F.3. Security Incident Response

Potential inputs:

- Escalated vulnerabilities, including security researcher reports to the CVD process.
- Escalated potential security incidents.

Potential outputs:

- Records indicating actions related to a security incident including triage, escalation and ongoing actions including lessons learned for significant events.
- Records demonstrating the security incident process have been tested to ensure roles and responsibilities are defined and understood, and that users are adequately trained to perform their role successfully.

An incident is defined as an attempt to access and/or adversely affect device functionality, data, systems, services, or networks. A manufacturer may have complimentary processes for vulnerability management and incident response, in which case the plans for both should cross-reference each other as needed.

Manufacturers should provide timely responses and communications to all stakeholders impacted by exploited vulnerabilities and security incidents for commercialized products as described below.

- Priority should be given to evaluating, mitigating, and communicating product security matters to patients, clinicians, and other stakeholders.
- Manufacturer Product Security Incident Response Team (PSIRT) should establish capabilities to work with hospital Security Incident Response Team to contain, eradicate, and recover from security incidents. This includes subject matter expertise and forensic evaluation for access logs, amounts of data held on device, and integrity of application, among others.
- Manufacturer incident response activities should include customer communication pertaining to patient safety, remediation, patch, upgrade, or updated documentation to control or mitigate the risk.
- If the security incident is associated with PHI or PII, then privacy considerations must be accounted for (e.g., privacy notifications, breach investigation). Potentially affected constituents must be notified (e.g., patients, physicians, care providers). The impacted party must provide information needed for proper incident response to enable successful breach determinations, notifications, and response.
- Any potential reported patient harm or safety risk should also be investigated and potential notification to regulars should be done in accordance with local laws and regulations.

F.4. Patch/Software Update Deployment

Potential inputs:

- Escalated vulnerabilities from vulnerability management processes, or backlog of known vulnerabilities.

- Maintenance Plan
- Release Management Procedure

Potential outputs:

- Records showing release of the product along with actions taken to address known vulnerabilities.

Prior to commercialization, a manufacturer should establish a Maintenance Plan to detail the plan product update process. The plan should address routine patching throughout the product's lifecycle, as well as emergency patching when warranted. Standardizing a predetermined frequency for routine patches and updates is recommended. Publishing and coordinating patches in a timely manner to mitigate medium to critical risk vulnerabilities is of prime importance to any vulnerability and patch management program.

Critical elements of an updated deployment include the ability to:

- Validate the remediation and successful patching of vulnerabilities, including impact to performance and clinical use.
- Perform proper version control to ensure patches can be identified once deployed on products.
- Deploy remediation, including routine and emergency software patches, by implementing at least one of the following secured methods that are then documented by both manufacturer and customer:
    - Remote Update: Patches applied via secure authorized remote service and support platforms provided by the manufacturer.
    - Customer Administered: Validated patches made available for customer retrieval and installation from a designated source, including direct download from the third-party that provides the product or component.
    - Service Visit: Local service administered cybersecurity patches. Note that this method is less optimal due to the time required to deploy local service personnel to customer facilities. However, it has utility in cases where faulty patching has foreseeable and serious safety risk and local service personnel may be required for resolution.
    - Ad-hoc Patching: Customers may accept engineering and technical risk for all other deployment mechanisms and/or application of cybersecurity patches not validated by the manufacturer. Note that this method is not advised due to the lack of validation by the manufacturer and potential impact to system performance or patient safety.
- Make customers aware of the availability of cybersecurity patches and upgrades for products through a public webpage and/or direct customer notification (e.g., email followed by letter).
    - For manufacturer-managed remote updates and service visits, routine reporting to customers of failures to patch products in the field is necessary, including products and components provided by third party entities that are no longer supported by their supplier.
    - It is essential that customers establish processes and/or technical means for routinely monitoring the designated communication channels predefined by the manufacturer for new information or changes regarding patches.
- Monitoring the performance of deployment after updates are released

- Key metrics relevant to the platform should be identified to monitor the effectiveness of update to released addressed vulnerabilities47
- Status of update deployment performance should be regularly communicated to leadership along with potential residual risk
- Metrics should be captured in annual risk management reporting as part of the product's design history file including potential required regulatory filings

F.5. Customer Security Communication

Potential inputs:

- SBOM
- Architecture & Design Documents
- Security Summary Report

Potential outputs:

- Customer Security Documentation, including Instructions for Use, Security Whitepapers, MDS2, SBOM

Manufacturers should make certain security documentation easily available to customers to support successful operation of products in their operating environments. Appendix D provides a list that should be considered. Additionally, the recommendations in the Design & Development section identified security documentation that should be created as part of the design transfer process. Any materials leveraged for customer security communications should be regularly reviewed and maintained.

In the interest of strengthening cybersecurity within the connected medical technology ecosystem, it is essential for manufacturers to communicate cybersecurity information, including cybersecurity vulnerabilities, to affected entities, including healthcare delivery organizations, the patients they serve, and other stakeholders, when necessary.

Additionally, other governmental agencies and organizations should be kept in the notification and disclosure process. Stakeholders include cyber emergency response teams (CERTs) and groups that share medical technology vulnerability and threat information, such as Health-ISAC and information sharing analysis organizations, (ISAOs). Manufacturers should also be aware of any additional reporting and remediation requirements imposed by regulators in the jurisdictions in which they operate, as these vulnerabilities may pose patient safety concerns.

A manufacturer should incorporate cross-functional stakeholders (including customer facing teams, customer support teams, public relations teams, and legal functions) in developing a communication plan for a vulnerability. The manufacturer should:

- Produce targeted customer bulletins or notifications and post to a public webpage or deliver via other available mechanisms to customers in a timely manner.

47 FDA guidance Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions section on TPLC Security Risk Management details specific metrics that should be considered.

- Evaluate related customer security documentation to determine if updates are indicated and provide an update if appropriate.
- Provide status updates to customers and third parties reporting vulnerabilities and incidents, with a routine cadence established by the cross-functional team while complaint handling investigation is in progress.

Achieving the timing for bulletins or notifications by the manufacturer during incidents may be dependent on timely and accurate communication with customers. Communication may occur even if recommended compensating controls or remediation steps are not yet available and are staged as they become available.

Recommendations for vulnerability disclosures can be found in the HSCC Medtech Vulnerability Communications Toolkit. They should typically include a summary of the vulnerability, including the CVSS score, the product(s) impacted, any remediation activities carried out, or compensating controls to mitigate the risk if no remediation is available. In addition, any information pertaining to potential patient safety risks, if known or applicable, should be communicated.

X. Evaluating JSP Progress and Maturity

Performing a periodic maturity assessment against the JSP can provide organizations valuable input when determining where product security program investments may be required to ensure secure product development. Organizations should strongly consider benchmarking themselves against the JSP and participating in the Medical Device Innovation Consortium (MDIC) JSP benchmarking survey (described below), to help target and prioritize program investments, and inform sector progress.

In the initial JSP, a series of questions were provided to enable organizations to assign Capability Maturity Model Integration (CMMI) maturity ratings to the JSP framework activities. These questions were removed in this version,48 as the MDIC now maintains a JSP benchmarking survey, and their approach is considered easier and will be maintained more effectively outside of the JSP.

In addition to the survey, MDIC annually collects shared results and produces a report on overall industry progress. This report is invaluable to organizations to benchmark themselves and to inform industry efforts on where more support on security product development activities may be required.

More information on MDIC JSP survey and benchmarking work can be found at https://mdic.org/program/cybersecurity/.

Note that other maturity assessments may be of value, and additional information on the CMMI maturity assessment can be found at https://cmmiinstitute.com/learning/appraisals/levels.

48 See https://cmmiinstitute.com/learning/appraisals/levels for more information.

Appendix A: Acronyms

This appendix section provides an overview of the acronyms used in this document.

C-I-A	Confidentiality Integrity Availability
CISO	Chief Information Security Officer
DHF	Design History File
DHS	U.S. Department of Homeland Security
EHR	Electronic Health Record
EU	European Union
FDA	U.S. Food and Drug Administration
GDPR	General Data Protection Regulation
HDO	Healthcare Delivery Organization
HCIC Task Force	Health Care Industry Cybersecurity Task Force
HHS	U.S. Department of Health and Human Services
HIMSS	Healthcare Information and Management Systems Society
HIPAA	Health Insurance Portability and Accountability Act
HPH	Healthcare and Public Health
IDE	Investigational Device Exemption
IT	Information Technology
ISAO	Information Sharing and Analysis Organization
ISAC	Information Sharing and Analysis Center
MDM	Medical Device Manufacturer
NIST SP	National Institute of Standards and Technology Special Publication
NIS	Network and Information Security 2 (NIS2) Directive (EU) 2022/2555
Health-ISAC	Health Information Sharing and Analysis Center
NCCoE	National Cybersecurity Center of Excellence
NSA	National Security Agency
PHI	Protected Health Information
PII	Personally Identifiable Information
R&D	Research and Development

SDL	Security Development Lifecycle
SDLC	Software Development Life Cycle
U.S.	United States

Appendix B: Glossary

Various cybersecurity and healthcare centric terms are used throughout this document. This appendix section provides an overview of what is meant by some of these key terms. Note that some of these terminologies and definitions were derived from authoritative sources listed in Appendix G, which describes the drafting of the Joint Security Plan.

Code Analysis: Source code analysis is the automated testing of a program's source code with the purpose of finding faults and fixing them before the software is sold or distributed. [SOURCE: Adapted from ISO IEC 62443-4-1]

Common Platform Enumeration (CPE): An industry standard structured naming scheme for information technology systems, software, and packages. [SOURCE: https://nvd.nist.gov/products/cpe]

Common Vulnerability Exposure (CVE): The Common Vulnerabilities and Exposures (CVE) system provides a reference method to identify, define, and catalog publicly disclosed cybersecurity vulnerabilities. [SOURCE: NIST SP 800-128]

Common Vulnerability Scoring System (CVSS): The CVSS score provides a way to capture the principal characteristics of a vulnerability and produce a numerical score [SOURCE: https://www.first.org/cvss/]

Compensating Controls: A cybersecurity compensating control is a safeguard or countermeasure deployed, in lieu of, or in the absence of controls designed by a device manufacturer. These controls are external to the device design, configurable in the field, employed by a user, and provide supplementary or comparable cyber protection for a medical device [SOURCE: Adapted from NIST SP 800-53A Rev. 4 and FDA Postmarket Management of Cybersecurity in Medical Devices]

Complaint Handling: Process for receiving, reviewing, and evaluating complaints. [SOURCE: 21 CFR 820.198]

Coordinated Vulnerability Disclosure: Process through which researchers and other interested parties work cooperatively with a manufacturer in finding solutions that reduce the risks associated with disclosure of vulnerabilities.

[SOURCE: AAMI TIR97:2019 Principles for medical device security – Postmarket risk management for device manufacturers]

Controlled Risk: Controlled risk is present when there is sufficiently low (acceptable) residual risk of patient harm due to a device's particular cybersecurity vulnerability. [SOURCE: FDA Postmarket Management of Cybersecurity in Medical Devices]

Critical Functions: Any product functionality which impacts the clinical safety or significantly disrupts the business operations of Customers. [SOURCE: Defined in this document]

Customers: Includes healthcare providers and patients. [SOURCE: Defined in this document]

Customer Complaint: Complaint means any written, electronic, or oral communication that alleges deficiencies related to the identity, quality, durability, reliability, safety, effectiveness, or performance of a medical device or health information technology after it is released for distribution. [SOURCE: Adapted from 21 CFR 820.198]

Customer Incident: An occurrence from a customer's use of software, products or services that actually or potentially results in adverse consequences to (adverse effects on) (poses a threat to) an information system or the information that the system processes, stores, or transmits and that may require a response action to mitigate the consequences. [SOURCE: Defined in this document]

Customer Security Documentation: Security information provided to customers to enable more robust risk assessments, identify configurable security controls, and allow them to better protect their systems. [SOURCE: Adapted from HSCC Model Contract Language for Medtech Cybersecurity]

Customer Security Requirements: A user, or potential user, of a system's functional and non-functional requirements that achieve the security attributes of a system. [SOURCE: Defined in this document]

Decommissioning: The first physical process in the disposition process and includes proper identification, authorization for disposition, and sanitization of the equipment, as well as removal of Patient Health Information (PHI) or software, or both. [SOURCE: Adapted from FDA Postmarket Management of Cybersecurity in Medical Devices]

Design: A process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. [SOURCE: Adapted from ISO/IEC/IEEE 24765:2017]

Design control: The application of a formal methodology used to conduct product development activities. [SOURCE: FDA Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions]

Design Input Requirements: The physical and performance characteristics of a product that are used as the basis for product design. [SOURCE: FDA Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions]

Dynamic Code Analysis: The testing and evaluation of a program by executing data in real-time. The objective is to find errors in a program while it is running, rather than by repeatedly examining the code offline. [SOURCE: Adapted from NISTIR 8011 Vol. 4]

End of Life: Point in time in the life cycle of a product starting when the manufacturer no longer sells the product beyond its useful life as defined by the manufacturer and the product has gone through a formal EOL process including notification to users. [SOURCE: Adapted from IMDRF Principles and Practices for the Cybersecurity of Legacy Medical Devices]

End of Support: Point in time in the life cycle of a product starting when the manufacturer terminates all service support activities and service support does not extend beyond this point. [SOURCE: Adapted from IMDRF Principles and Practices for the Cybersecurity of Legacy Medical Devices]

Exceptions: An instance when a cybersecurity risk is identified (both pre- and post-launch of the product) and the manufacturer determines that no action is needed.

Exploitable: The state of a vulnerability that has an actual proven compromise that works effectively. [SOURCE: Adapted from IMDRF Principles and Practices of Cybersecurity for Legacy Medical Devices]

Exploitability: the feasibility or ease and technical means by which the vulnerability can be

exploited by a threat. [SOURCE: Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions]

Failure Mode and Effects Analysis (FMEA): A step-by-step approach for identifying all possible failures in a design, a manufacturing or assembly process, or a product or service. [SOURCE: IEC 60812]

Fuzz Testing: A software testing technique, often automated or semi-automated, that involves providing invalid, unexpected, or random data to the inputs of a computer program. The program is then monitored for exceptions such as crashes, failing built-in code assertions or for finding potential memory leaks. Fuzzing is commonly used to test for security problems in software or computer systems and is a type of robustness testing. [SOURCE: Adapted from Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions]

Harm: Injury or damage to the health of people, or damage to property or the environment. [SOURCE: Adapted from ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers]

Hazard: Potential source of harm. [SOURCE: Adapted from ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers]

Hazard Analysis: The first step in the process is used to assess risk and identify different types of hazards. [SOURCE: ISO 14971]

Incident Response: Actions taken to mitigate or resolve a security incident. [SOURCE: Adapted from NIST SP 800-61 Rev. 2]

Internal/External Security Audit: Review and examination of data processing system records and activities to test for adequacy of system controls, to ensure compliance with established security policy and operational procedures, to detect breaches in security, and to recommend any indicated changes in control, security policy, and procedures. [SOURCE: Adapted from NIST SP 800-12 Rev. 1]

Kitted: Manufacturer provided hardware which operates as a medical device accessory and contains any necessary applications or components (e.g., operating system) to ensure integration with a related medical device. The manufacturer is responsible for maintenance of the provided hardware and any necessary applications or components.

Known Vulnerability*:* An error in implementation. A publicly identified specific weakness in computational logic (e.g., code) including configuration found in software and hardware components that, when exploited, results in a negative impact to confidentiality, integrity, or availability. A known vulnerability in a released product or software should have an identified CVE and associated inherent CVSS score. [Source: NIST CVD, also consistent with UL 2900-1)]

Known Exploited Vulnerability (KEV): A known vulnerability with a publicly identified known exploit. [SOURCE: CISA Known Exploited Vulnerabilities Catalog - https://www.cisa.gov/known-exploited-vulnerabilities-catalog)

Medtech: Medtech, or Medical Technology, can be defined as the technologies that diagnose, treat, and/or improve a person's health and wellbeing.https://apacmed.org/the-medtech-industry/what-is-medical-technology/ [SOURCE: https://apacmed.org/the-medtech-industry/what-is-medical-technology/]

Malware: A program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the data, applications, or operating system. This includes both known and unknown (Zero Day) viruses, spyware, ransomware, and other forms of malicious code that exploit vulnerable systems. [SOURCE: Adapted from Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions]

Patch Management: The systematic monitoring, identification, assessment, remediation, deployment, and verification of operating system and application software code updates. These updates are known as patches, hot fixes, and service packs to operating systems, third-party products and components, and in-house developed software. [SOURCE: Adapted from NIST SP 800-137]

Patient Harm: Physical injury to the health of patients, including death. Cybersecurity exploits (e.g., loss of authenticity, availability, integrity, or confidentiality) of a device may pose a risk to health and may result in patient harm. [SOURCE: Adapted from Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions]

Patient Safety: The prevention of harm to patients, including that of cybersecurity-related events. [SOURCE: Adapted from ISO 14971]

Penetration Testing: A testing methodology in which assessors, using all available documentation such as system design and working under specific constraints, attempt to circumvent the security features of an information system. [SOURCE: NIST SP 800-137]

Preliminary Hazard Analysis (PHA): A technique used in the early stages of system design. It focuses on identifying apparent hazards, assessing the severity of potential accidents that could occur involving the hazards, and identifying safeguards for reducing the risks associated with the hazards. [SOURCE: Defined in this document]

Product Lifecycle: Managing the entire lifecycle of a product from inception, through engineering design and manufacture, to service and disposal of manufactured products. [SOURCE: Adapted from IMDRF Principles and Practices of Cybersecurity for Legacy Medical Devices]

Purchasing: The acquisition of materials or services from outside the organization following a documented process to ensure what is acquired conforms to what is expected. [SOURCE: Adapted from ISO 13485]

Remediation: Countermeasures to reduce a cyber asset's susceptibility to cyber-attack over a range of attack tactics, techniques, and procedures. [SOURCE: Adapted from FDA Postmarket Management of Cybersecurity in Medical Devices]

Remediation Planning: Planning of processes and actions by which organizations identify and resolve threats to their system. [SOURCE: Adapted from NIST SP 800-216]

Remote Access: Access to a product or an organization's non-public information system by an authorized user such as Service and Support communicating through an external network. [SOURCE: Adapted from NIST SP 800-128]

Remote Support: Support activities conducted by individuals communicating through an external network (e.g., the Internet). [SOURCE: Defined in this document]

Removable Media: Portable electronic storage media such as magnetic, optical, and solid-state devices, which can be inserted into and removed from a computing device and used to store text, video, audio, and image information. Such devices have no independent processing capabilities. Examples include hard disks, floppy disks, zip drives, compact disks, thumb drives, pen drives, and similar USB storage devices. [SOURCE: Adapted from CNSSI 4009-2015]

Residual Risk: Risk remaining after risk control measures have been implemented.

[Source: ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers]

Risk: A measure of the extent to which an entity is threatened by a potential circumstance or event, and typically is a function of:

- The adverse impact, or magnitude of harm, that would arise if the circumstance or event occurs; and
- The likelihood of occurrence.

[SOURCE: NIST SP 800-37 Rev. 2]

Note 1 to entry: The probability of occurrence includes the exposure to a hazardous situation and the possibility to avoid or limit the harm.

Note 2 to entry: For a security related risk, determining a statistically based probability is typically not possible. The use of a proxy such as exploitability or likelihood score is acceptable.

[SOURCE: ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers]

Risk Control: process in which decisions are made and measures are implemented by which risks are reduced to, or maintained within, specified levels

[SOURCE: ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers]

Risk Management: systematic application of management policies, procedures and practices to the tasks of analyzing, evaluating, controlling, and monitoring risk. [Source: ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers]

Robustness Testing: A testing methodology to detect the vulnerabilities of a component under unexpected inputs or in a stressful environment. [SOURCE: Adapted from CNSSI 4009-2015]

Secure Coding Standards: Guidelines for writing software code that mitigates common security flaws specific to a programming language or in general to all software. [SOURCE: Defined in this document]

Security / Cybersecurity: State where information and systems are protected from unauthorized activities, such as access, use, disclosure, disruption, modification, or destruction to a degree that the related risks to violation of confidentiality, integrity, and availability are maintained at an acceptable level throughout the life cycle. [SOURCE: ISO IEC 81001-5-1 Health software and health IT systems safety, effectiveness and security – Part 5-1: Security – Activities in the product life cycle Ed. 1.0 [17], 3.2.13]

Security Design Risk: An error in design that may result in degraded security. Errors or weaknesses are often categories or general potential weaknesses and not specifically known vulnerabilities. [SOURCE: ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers]

Security Incident: An event that may indicate that a device's data and security may have been compromised. This includes, but is not limited to:

- Attempts to gain unauthorized access to a system or its data
- Unwanted disruption or denial of service
- Unauthorized use of a system for the processing or storage of data
- Changes to system hardware, firmware or software characteristics without owner's knowledge, instruction or consent

[SOURCE: Adapted from ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers]

Security Management Plan: Used to document all framework components carried out through the design process and post commercialization. May also capture technical and process gaps, including exceptions. May be incorporated in a product risk management file or equivalent. [SOURCE: Adapted from Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions]

Security Requirements: A set of design-level requirements that comprise a product or other commercial offerings, ensure security issues are mitigated in both software and system components during design control, and are processed through Risk Management. [SOURCE: Adapted from ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers]

Sensitive Information and Data: Protected health information (PHI), personally identifiable information (PII), proprietary software source code or business logic, configuration parameters, user credentials, cryptographic keys, quality control and calibration results. [SOURCE: Defined in this document]

Static Code Analysis: The automated analysis of software code for security flaws and adherence to a secure coding standard. [SOURCE: Adapted from NISTIR 8011 Vol. 4]

System Hardening Practices: A documented process or mechanism for securely configuring or implementing commonly used technologies. [SOURCE: Adapted from NIST SP 800-152]

Third-Party Entities: External individuals and organizations such as suppliers involved with products or acquisition, that collaborate at any point in the product lifecycle, including acquisition, development and servicing. [SOURCE: Defined in this document]

Threat Monitoring: A methodology for optimizing system, product, network, application, and connection security by identifying objectives and vulnerabilities, and then defining countermeasures to prevent, or mitigate the effects of, threats to the system. [SOURCE: Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions]

Threat Source: The intent and method targeted at the intentional exploitation of a vulnerability or a situation and method that may accidentally trigger a vulnerability. [SOURCE: Adapted from FDA Postmarket Management of Cybersecurity in Medical Devices]

Uncontrolled Risk: Uncontrolled risk is present when there is unacceptable residual risk of patient harm due to inadequate compensating controls and risk mitigations. [SOURCE: Adapted from Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions]

Validation: Establishing by objective evidence that specified requirements conform with user needs and intended use(s). [SOURCE: IMDRF Principles and Practices of Cybersecurity for Legacy Medical Devices]

Verification: Confirmation by objective evidence that the results of the design effort meet the design input. [SOURCE: IMDRF Principles and Practices of Cybersecurity for Legacy Medical Devices]

Vulnerability: A weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source. [SOURCE: Adapted from Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions]

Vulnerability Scanning: The automated analysis and detection of vulnerabilities such as missing patches and misconfiguration in operating systems and other third-party software. [SOURCE: Adapted from NIST SP 800-115]

Appendix C: Example Design Input Requirements for Security

The controls requirements included in device design are informed by the device type, design, use environment, and intended use or functionality. As such, there is no one size that fits all set of design inputs that should be utilized. Design inputs highlighted here in this appendix section are not intended to be comprehensive; rather, they serve as examples of input requirements that could be considered within the context of use for a given device. These design input requirements are categorized by OWASP Security Design Principles.

- **Minimize Attack Surface**
  - The system shall restrict access of removable media to what is necessary for intended use.
  - Execution of software on the system shall be restricted to explicitly authorized or validated software components.
  - The system shall provide the capability to anonymize exported data such that an individual or customer is not identifiable.
  - Ports, protocols, services, and addresses available on the system and its network connection shall be restricted to the minimum necessary for intended use and configurable locally by authorized user.
  - The system shall be capable of enabling and disabling protocol stacks, individual ports, and services, and contain manageable host-based firewall.
  - The system shall provide capability to explicitly enable or disable remote access to the system.
  - The system shall notify users to change default passwords after initial use.
  - The system shall be capable of restricting repeated and failed user access attempts.
- **Establish Security By Default**
  - The system shall have the ability to require a minimum password length.
  - The system shall have the ability to require a minimum password complexity.
  - The system shall have the ability to require periodic password renewal.
  - The system shall have the ability to restrict password reuse.

- The system shall have the capability to automatically or manually backup data necessary for intended use locally or to an external location.
- All sensitive information and data shall be encrypted in transit and at rest using an industry-accepted encryption mechanism and practice.
- The system shall prominently notify users when sensitive information and data are displayed on screen or if encryption is disabled in transit.
- The system shall have routine functionality for handling exceptions, errors and aborts that does not expose sensitive information and data.
- The system shall enforce strict order of execution during system start and end.
- All remote or local user activity which interacts with sensitive information and data as well as critical functions on the system shall be recorded in an audit log.
- All audit log entries shall include a start and end date-timestamp, user ID, role/privileges at time of access, success/failure and a description of the action performed.
- The audit log shall locally retain an individual entry for a configurable period of time or allocation of file system space.
- The system shall provide capability for a user to reset their own password or administrative reset, which is logged.
- The system shall provide the ability to create and assign a unique user ID and password to each remote or local user.

- **Principle of Least Privilege**
  - Execution of software on the system shall be limited to the minimum privileges necessary.
  - The system shall support the creation and assignment of roles that grant the minimum user privileges necessary for intended use of data and functions.

- **Principle of Defense in Depth**
  - The system shall support multiple factors for user authentication and is capable of centralized authentication.
  - The system shall provide the capability to prevent the execution of known malicious software.
  - The system shall be capable of manually or automatically locking the display and requiring user authentication after a configurable period of user inactivity to continue use such that sensitive information and data are not visible.
  - The system shall provide capability for a user to reset their own password or administrative reset, which is logged.

- **Fail Securely**
  - The system shall be capable of restoring functionality to an operational state.

- **Don't Trust Services**
  - The integrity and composition of all data as input or output of the system shall be validated such that modification is detected and/or rejected.
  - All remote or local access to the system by user or an external system shall be authenticated prior to granting access to data or functions.

- **Separation of Duties**
  - The audit log shall be restricted in access to only authorized users.
  - The audit log shall be exportable and readable by authorized users and have the capability to integrate with security information and event management for real-time analysis.
- **Avoid Security by Obscurity**
  - The security of a system shall not rely upon knowledge of the source code, or shared hard coded credentials being kept secret.
- **Keep Security Simple**
  - The system shall allow security controls to be configured with no significant downtime and centrally managed by authorized users.
- **Fix Security Issues Correctly**
  - The system shall support authorized updates to mechanisms for controlling the execution of authorized or malicious software.

Components of the system shall support software updating and patches with no significant downtime using standard centralized patch management systems.

Appendix D: Example Customer Security Documentation

Customers require security documentation to enable more robust risk assessments, identify configurable security controls, and allow them to better protect their systems. This appendix section provides an overview of items that may be included in Customer Security Documentation.

The following are examples of the types of information which may be included in documentation of security for medical devices or health IT:

- Product Description
- Hardware Specifications
- Operating Systems
- Third-party Software
- Network Ports and Services
- Sensitive Information and Data Transmitted
- Sensitive Information and Data Stored
- Network and Data Flow Diagram
- Malware Protection
- Authentication
- Network Controls
- Physical Controls
- Encryption
- Audit Logging

- Remote Connectivity
- Service Handling
- End-of-Life Date
- End-of-Support Date
- Secure Coding Standards
- System Hardening Standards
- Risk Summary
- Third Party Certification or Attestation
- Manufacturer's Disclosure Statement for Medical Device Security (MDS2)
- SBOM
- Multi-factor Authentication (MFA) Usage

Product Description

[Insert basic description of function or purpose of the product or solution. Photo is optional but recommended.]

Hardware Specifications

[List hardware components and specs]

- [List]
- [List]

Operating Systems

[List hardware operating systems and versions]

- [List]
- [List]

Third-party Software

[Also referred to as a Software Bill of Materials (SBOM), includes a list of third-party software and version numbers where applicable. Having a SBOM will aid customers in mitigating cybersecurity concerns on their healthcare technologies and ultimately to the systems/networks these technologies are attached to. The following are example attributes that would enable customers to leverage a bill of materials in protecting their assets.

Detailed attributes include:

- All commercial, open source, and custom code must be included.
- Commercial technology components (e.g., processers, network cards, sound cards, graphic cards, memory) must be included.
- The software list will be codified using an industry standard, such as Common Platform Enumeration (CPE), Software Identification tag (SWID), or Software Package Data Exchange (SPDX) that allows the software list to be searched and used to check against vulnerability feeds.
- The list will be available in an electronic format that allows bulk uploading into common asset inventories, vulnerability management systems and configuration management databases.

- The SBOM will be provided to a customer both upon purchase and after significant software or hardware upgrades.
- Supplier will maintain a SBOM for all product versions that will be accessible remotely by customers.]

Supplier and Name	Version	Description
[e.g., Microsoft Windows 10]	[e.g., 1607]	[e.g., Long-Term Servicing Branch]

Network Ports and Services

[List Network Ports and Services]

Port	Protocol	Service Name	Description of Service	Encrypted	Open/Closed
XXX	XXX	XXXXX	XXXXX	XXX	XXX

Sensitive Information and Data Transmitted

[List sensitive information and data transmitted. This can include PHI/PII/Potential access to wireless credentials, etc.]

- [List]
- [List]

Sensitive Information and Data Stored

[List sensitive information and data stored. This can include PHI/PII/Potential access to wireless credentials, etc.]

- [List]
- [List]

Network and Data Flow Diagram

[Provide a diagram that describes how the product resides in a customer environment, showing the system components (1 or N computers, routers, switches, adjacent systems, remote connectivity) types of connectivity (e.g., RS232, RJ45, Serial to TCP/IP conversion), what types of data is in transit and at rest (e.g., PHI, QC, config data), and how these are secured (e.g., in transit IPSec, HTTPS/TLS, WIFI WPA2PSK; at rest BitLocker, SQL TDE)

Important: Include if the device makes PHI/PII available via network or point-to-point connection (wired/wireless)?

- Is connected data encrypted in transit?
- Does service have network or p-to-p access to PHI (remote or in-room)?]

Malware Protection

[Describe and recommend the anti-malware measures available (e.g., validated AV solutions, AV partners, how AV is managed, application whitelisting like AppLocker or McAfee Embedded Control, advanced antimalware solutions, software restriction policies)]

Patch Management

[Describe and recommend the method in which we maintain, provide, and deploy patch updates for this product. Examples include, "Patches are installed by a field service engineer during a routine service visit or during the yearly service visit. In the even that there is no patch management solution in place, also communicate this in this section.]

Authentication & Authorization

[Describe and recommend the controls that customers have with user's authenticating and granting permissions to features and functionality, how users are managed, the default use accounts on the system and how to change and configure accounts. This includes the ability to disable user accounts]

Network Controls

[Describe and recommend the firewall rules, IPSec rules, host file restrictions, browser Internet access restrictions, MAC and IP address filtering)]

Encryption

[Describe and recommend where and how encryption is applied on the system (e.g., all network traffic is TLS 1.2, at rest is BitLocker with AES 256); include rationale and definition of the selected encryption algorithm, selected key/certificate format and strength, key hierarchy, key and algorithm lifecycle management, and cryptographic processes (e.g., provisioning)]

Audit Logging

[Describe the audit logging process, where they are stored, what an auditable event entails, who has access to audit logs and any file permissions. Describe if audit logs are synchronized with reliable time sources and have the proper time zone set or no time offset (e.g., GMT or UTC).

- What is the typical and maximum number of records retained on the device when in use?
- Do users have a means to irreversibly delete audit log records in the device?
- Does Service ever retain copies of PHI/PII data (is it encrypted by service) in audit logs?
- Application Auditing
    - Audit file location: E:\PieRoot\Logfiles*.pld
    - Audit files hashed with SHA256 when complete for integrity.
    - Auditable Events:
        - Service Start/Stop
        - User login/logout
        - User session created/destroyed.
        - User login from multiple workstations.

    - Client application connect/disconnect with IP address and port.
    - Failed client connection attempts.
    - Changes in application configuration.
    - Failed/successful attempts to access, modify, or delete security objects, (e.g., roles, permissions, etc.).

- Audit file permissions:
    - Administrators group: Read.
    - Auditors group: Read.
    - DB Auditors group: Full control.
    - DB Administrators group: Full control.
    - Virtual/Managed service accounts (audit file creators): Full control.
    - Users: None.]

Remote Connectivity

[Describe the nature of remote connectivity, what ports, protocols, URLs and endpoints for communication as well as security measures applied to the remote connection (e.g., TLS). Indicate if multi-factor authentication is used.]

Service Handling

[Describe what routine maintenance service personnel perform, what security policies and procedures they follow (e.g., never take PHI or PII, on-site authorization protocol, encrypted Removable Media, hardened service laptops, whether or not service laptops connect to product, routine AV update during visit, secure installation/implementation principles, service authentication to product, decommissioning process, once decommissioned how the product hard drive is wiped, how the product is recovered from the field or destroyed, and what customer data and features service personnel interact with.)]

End-of-Life (EOL)

[Describe the life cycle of the product in relation to when it will no longer be marketed, sold, or receive major design change. Provide dates if available otherwise describe how EOL is communicated.]

End-of-Support (EOS)

[Describe the life cycle of the product in relation to when it will no longer be supported, which may include cybersecurity support. Provide dates if available otherwise describe how EOL/EOS is communicated.]

Secure Coding Standards

[Describe the secure coding standards used]

- [List the industry secure coding standards used during software development (e.g., SEI CERT Java Secure Coding Standard)]

System Hardening Standards

[Describe the secure hardening standards used, may also create appendix to list out standards used.]

Name of Standard	Version Number	Source of Standard
[Insert name of standard]	[Insert version number]	[Insert URL]

Risk Summary

[This section should contain a summary of risks found within a penetration test, remediation report, or other topics and compensating controls that correspond to additional risks outlined in the product security white paper. This may also include any findings from application scans.]

Appendix E: Example Organizational Structure

The intent of this appendix section is to provide an example of roles and responsibilities within organizations to support the adoption and continuous improvement of cyber security for medical devices and health IT:

Medical Device Manufacturers and Health IT Suppliers

- **Chief Product Security/Cybersecurity Officer:** Responsibility to drive product and solution security throughout their organization, including identifying best practices and companywide technical standards, processes, and policies, for overall governance or guidance. In addition, this individual advises executive management, product management, project management, R&D heads, and manufacturing heads with regard to security for all products, solutions and services. Responsible for implementing pre-market product security design and postmarket support including cybersecurity events and incidents for products in scope. Independent of Information Security and in cooperation with the CEO, this individual will advise appropriate processes and structures to introduce security into products, solutions, and services.
- **Product Security/Cybersecurity Engineering**
  - Security Architects: This person will work with R&D, service, and quality organizations to research common security vulnerabilities and their remediation; develop procedures to incorporate hardening into product development; work with individual product teams in securing their products; and proactively educate teams across the company on security best practices for products under development.
  - Penetration Testers: This person will perform security penetration testing, ethical hacking, and red team activities in order to identify unique and common vulnerabilities in products under development. This includes performing vulnerability analysis and research, formalizing security testing procedures in the product lifecycle, performing penetration testing with remediation plans and formal reporting, and supporting red team, covert, and security activities to test organizational readiness.
- **Product Security/Cybersecurity Incident Response**
  - Incident Responder: This person will manage technical strategy, process, timelines, resources, and progress for incidents relating to products at customer sites or with security researchers.
  - Vulnerability Manager: This person will track the escalation, follow-up, and remediation of vulnerabilities throughout the product lifecycle.

- **Product Security/Cybersecurity Program Management**
  - Policy and Compliance Analyst: This person will ensure the adoption and continuous improvement of security policies and procedures for products in compliance with industry standards and regulations.
  - Strategic Program Manager: This person will work cross-functionally to create programs and initiatives for establishing training, awareness, and fundamental capabilities for improving security of products.
- **Product Security Testing** – Responsible for assessing and testing products in development and in the market to understand cybersecurity risk and find issues before an external party does. Comprised of Product Security members and other participants (such as 3rd parties) as needed.

Larger organizations may choose to have multiple business or product-specific roles including a dedicated product security officer, manager, and/or engineers.

Appendix F: Example Organizational Training

The intent of this appendix section is to provide training information that will help organizations mature their cybersecurity programs. A comprehensive training program for cybersecurity includes the following:

- **Training Requirements**
- Training requirements for each relevant role must be established and periodically reviewed to determine if they need to be updated.
- **General Awareness Training**
- All relevant employees in the organization should understand the principles of cybersecurity, the framework of the organization's program and the different roles and responsibilities for cybersecurity.
- **Training by Roles**
  - Training for Security Practitioners
    - Engineers
      - Architecture: Security experts who participate in architecting products or contribute to the security architecture components of products should be trained in secure architecture principles and patterns.
      - Threat modeling and security risk analysis: Security experts who participate in threat modeling should be trained in the principles of threat modeling and the use of threat modeling tools, as well as methods of translating threats into a risk management framework.
      - Design: Security experts who participate in product design or contribute to the security design of products should be trained in secure design principles and patterns.
      - Testing: Security experts who perform or guide security testing of products should be trained in security testing methodologies, tools and interpretation of testing results.
      - Forensics and Incident Response: Security experts who evaluate evidence of security incidents should have training in security forensic analysis in addition to practical experience. Those who participate in the incident response process should be trained in that process and the theory of incident response, in addition to practical experience.
    - Penetration Testing: Penetration testers should have proper training in penetration testing techniques and tools as well as considerable practical experience before being qualified as a penetration tester for products.
    - Security Officers/Directors/Managers/Advocates/Champions: Non-technical security practitioners should be trained in the secure development lifecycle, the company's security framework and the company's quality system.

- Training for Related Activities – Non-dedicated Practitioners
    - Software/firmware/hardware/systems engineers
        - Secure Coding standards: Engineers involved in developing code should be trained in secure coding standards.
        - Static and dynamic code analysis tools: Engineers involved in development and/or configuration management should be trained in the use and interpretation of automated code analysis tools.
    - Sustaining engineering (maintenance for vulnerabilities): Engineers and product managers involved in maintenance of commercialized products should be trained in the interpretation of vulnerability notifications and the steps necessary to respond to vulnerabilities identified in the products.
    - Risk managers: Risk managers should be trained on the incorporation and interpretation of security risks within the existing risk management framework.
    - Requirements engineers: Requirements engineers should be trained to be able to incorporate standard security requirements into risk catalogs as well as novel requirements identified during threat modeling.
    - Deployment engineers: Those responsible for deploying products in the field should be trained on adapting the products to the IT environment as well as configuring that environment, to match the security requirements specified for the products.
    - Support and service engineers: Support and service engineers should be trained to recognize, remediate, and escalate security issues reported or discovered in fielded systems.
    - Information Security/IT/Systems Administration (infrastructure): Those responsible for defining and implementing the security infrastructure of the company's IT and physical environments should be trained in the access and protection requirements of secure development and manufacturing.

- **Periodic refreshers for awareness:** Employees who have participated in the overall awareness and more detailed training should be given periodic refresher training to remind them of the key elements of the previously acquired training.
- **Periodic updates for changes in threat landscape, technology, program:** As the threat landscape changes, as new technology is developed in cybersecurity and as the company's security program evolves, the training requirements and trainings themselves should be updated to stay in synchronization.
- **Qualification and Certification of Security Experts:**
    - Certification: Requirements for certification for security experts and practitioners should be established and upheld as minimum qualifications to participate in these activities. Certifications can be external and/or internal (based on completion and confirmation of an internal training regime).
    - On the job experience: Minimum requirements for actual experience practicing security activities should be specified for a person to be considered a security expert in a particular sub-role of expertise.
    - Mentoring and community: Participation in the community of experts within the company should be included as a requirement to be considered a security expert. This may include peer relationships as well as mentor-mentee relationships.
    - Levels of expertise: Different levels of expertise should be defined by the degree to which a practitioner has achieved these aspects of qualification. The levels should correspond to minimum requirements for specific security-related activities. For instance, a penetration tester may be allowed to be the lead tester for a product only in the case of a minimum amount of time practicing as a penetration tester.

- **Drills:** Periodic drills should be exercised, in order to ensure the ability of practitioners to apply training. These may take the form of tabletop incident response drills or full-blown red team/blue team exercises.

Appendix G: Mapping To Medical Technology Guidance and Standards

A listing of new, updated, or relevant standards and guidance impacting healthcare considered in the JSP v2 update include:

- IMDRF Principles and Practices for Medical Device Security (18-Mar-2020)
- MDCG 2019-1 Rev 1 Guidance on cybersecurity for Medical Devices (1-Jul-2020)
- ISO TR 24971:2020 Guidance on the application of ISO 14971 (Jun 2020)
- HSCC Medtech Vulnerability Communications Toolkit (Apr 2022)
- HSCC Model Contract Language for Medtech Cybersecurity (Mar 2022)
- HSCC Managing Legacy Technology Security (Mar 2023)
- Health-ISAC Medical Device Cybersecurity Lifecycle Management (Oct 2020)
- MDIC/MITRE Playbook for Threat Modeling Devices (Nov 2021)
- ANSI/AAMI SW96:2023 medical device security – Security Risk Management for Device Manufacturers (Dec 2022)
- AAMI TIR97:2019 Principles for medical device security – Postmarket risk management for device manufacturers (Sep 2019)
- ISO IEC 81001-5-1 Health software and health IT systems safety, effectiveness, and security – Part 5-1: Security – Activities in the product life cycle Ed. 1.0 (Dec 2021)
- NIS2 Directive (Aug 2023)
- MDIC Medical Device Cybersecurity Maturity: Industry Benchmarking Report 2022 (Oct 2022)
- MITRE Rubric for Applying CVSS to Medical Devices (Oct 2020)
- Health Industry Cybersecurity Practices: Managing Threats and Protecting Patients (HICP 2023 Edition)
- Consolidated Appropriations Act, 2023, Section 3305 (December 2022)
- FDA Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions (Sep 2023)

A mapping of content from version 2 of this document to both new and updated as well as original standards and guidance impacting healthcare is provided below:

JSP Version 2 Section	Healthcare Standard or Guidance	Section	Section Title
A.1 Concept – Release/Change Project Planning	ISO 81001-5-1	6.2	Problem and modification analysis
	ISO 81001-5-1	6.3	Modification implementation
A.2. Concept – Voice of the Customer (Stakeholder Security Needs Identification)	ANSI/AAMI SW96 (2023)	Annex E	Third-Party Service Organizations and Security

	FDA Postmarket Management of Cybersecurity in Medical Devices	V.A	Premarket Considerations
	FDA Postmarket Management of Cybersecurity in Medical Devices	V.B	Postmarket Considerations
	IMDRF Principles and Practices for Medical Device Cybersecurity	4.3	Shared Responsibility
	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VIII.C	Third Party Servicers
	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VIII.D	Inventory/Asset Management
	MDCG 2019-16	2.3	Basic Security Concepts: Intended use and intended operational environment for use
	MDCG 2019-16	2.5	Basic Security Concepts: Operating Environment
	MDCG 2019-16	2.6	Basic Security Concepts: Joint Responsibility - Specific expectations from other stakeholders
	MDCG 2019-16	4.2	Instructions for use
	MDCG 2019-16	4.3	Information to be provided to healthcare providers
	HSCC Model Contract-language for Medtech Cybersecurity	N/A	N/A
	IEC 80001-2-2	5	Security Capabilities
A.3. Concept – Security Management Planning	AAMI TIR97:2019	3	Postmarket considerations for security policies and security program administration
	AAMI TIR97:2019	Annex A	Sample medical device security policy statements

	FDA Postmarket Management of Cybersecurity in Medical Devices	X.	Appendix: Elements of An Effective Postmarket Cybersecurity Program
	FDA Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions	IV	General Principles
	FDA Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions	VI	Cybersecurity Transparency
	IMDRF Principles and Practices for Medical Device Cybersecurity	4.2	Total Product Life Cycle
	IMDRF Principles and Practices for Medical Device Cybersecurity	5.4	Premarket Considerations: TPLC Cybersecurity Management Plan
	IMDRF Principles and Practices for Medical Device Cybersecurity	5.6	Premarket Considerations: Documentation for Regulatory Submission
	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VII.A	Legacy Devices: Core Practices - Governance
	MDCG 2019-16	3.8	Lifecycle Aspects
	MDCG 2019-16	4.1	Documentation
	UL-2900-1	11	Product Management
	UL-2900-2-1	11	Product Management
	ISO 81001-5-1	4.1	Quality management (of security activities during the lifecycle)
	ISO 81001-5-1	5.1	Software development planning
	ISO 81001-5-1	B	Guidance on implementation of security life cycle activities

	IMDRF Principles and Practices for Medical Device Cybersecurity	5.2	Premarket Considerations: Risk Management Principles for the TPLC
	MDCG 2019-16	3.2	Security Risk Management
	ISO 81001-5-1	4.2	Security Risk Management
	ISO 81001-5-1	7.1	Risk management context
	NIS2	21	Cybersecurity risk-management measures
B.1 Risk Management – Security Design Risk Assessment	AAMI TIR97:2019	Annex B	Security risk management for healthcare networks
	ANSI/AAMI SW96 (2023)	4	General requirements for security risk management
	ANSI/AAMI SW96 (2023)	6	Security risk evaluation
	ANSI/AAMI SW96 (2023)	Annex F	Security risk scoring based on likelihood of occurrence
	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VII.C	Legacy Devices: Core Practices - Cybersecurity Risk Management
	MDCG 2019-16	3.4	Security Risk Assessment
	ISO 81001-5-1	7.3	Estimation and evaluation of security risks
	IMDRF Principles and Practices for Medical Device Cybersecurity	5.2	Premarket Considerations: Risk Management Principles for the TPLC
	MDCG 2019-16	3.2	Security Risk Management
	ISO 81001-5-1	4.2	Security Risk Management
	ISO 81001-5-1	7.1	Risk management context
	NIS2	21	Cybersecurity risk-management measures
B.2. Risk Management – Security Integration into Safety Risk Assessment	ANSI/AAMI SW96 (2023)	6	Security risk evaluation

	ANSI/AAMI SW96 (2023)	Annex F	Security risk scoring based on likelihood of occurrence
	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VII.C	Legacy Devices: Core Practices - Cybersecurity Risk Management
	MDCG 2019-16	Annex IV	Cybersecurity risk management process and safety risk management relationship
	UL-2900-2-1	12	Safety-Related Security Risk Management
B.3. Risk Management – Security Risk Management Summary Approval	ANSI/AAMI SW96 (2023)	8	Evaluation of overall residual security risk acceptability
	ANSI/AAMI SW96 (2023)	9	Security risk management review
	ANSI/AAMI SW96 (2023)	Annex C	Security risk management report
	MDCG 2019-16	3.5	Security Benefit Risk Analysis
C.1. Supplier Management – Purchasing Process	UL-2900-1	12	Vendor Product Risk Management Process
	HSCC Model Contract-language for Medtech Cybersecurity	N/A	N/A
	AAMI TIR97:2019	6	Postmarket management of fielded devices
	ISO 13485:2016+A11:2021	7.4	Purchasing
C.2. Supplier Management – Performance Management	UL-2900-1	12	Vendor Product Risk Management Process
	HSCC Model Contract-language for Medtech Cybersecurity	N/A	N/A
	AAMI TIR97:2019	6	Postmarket management of fielded devices
	ISO 13485:2016+A11:2021	7.4	Purchasing

D.1. Design & Development – Security Requirements Development	AAMI TIR97:2019	4	Design features for postmarket security risk management
	AAMI TIR97:2019	5	Installation and configuration
	ANSI/AAMI SW96 (2023)	7	Security risk control
	FDA Cybersecurity in Medical Devices: Quality System Considerations and Content of Premarket Submissions	V	Using SPDF to Manage Cybersecurity Risk
	MDCG 2019-16	3.3	Security Capabilities
	MDCG 2019-16	3.6	Minimum IT Requirements
	UL-2900-1	7	Risk Controls
	UL-2900-1	8	Access Control, User Authentication, and User Authorization
	UL-2900-2-1	7	Risk Controls
	UL-2900-2-1	8	Access Control, User Authentication, and User Authorization
	ISO 81001-5-1	5.2	Health software requirements analysis
	ISO 81001-5-1	8	Software configuration management process
	ISO 81001-5-1	A.4	Secure coding best practices
	NIS2	24	European cybersecurity certification schemes
D.2. Design & Development – Secure Architecture and Design	ANSI/AAMI SW96 (2023)	5	Security risk analysis
	ANSI/AAMI SW96 (2023)	Annex D	Threat Modeling
	IMDRF Principles and Practices for Medical Device Cybersecurity	5.1	Premarket Considerations: Security Requirements and Architecture Design

	IMDRF Principles and Practices for Medical Device Cybersecurity	6.1	Postmarket Considerations: Operating Devices in the Intended Use Environment
	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VII.D	Legacy Devices: Core Practices - Future Proofing
		VIII.A	Connectivity
	MDCG 2019-16	2.4	Basic Security Concepts: Reasonably foreseeable misuse
	MDCG 2019-16	3.1	Secure by design
	UL-2900-1	9	Remote Communication
	UL-2900-1	10	Sensitive Data
	UL-2900-1	A1	Sources for Software Weaknesses
	UL-2900-2-1	9	Remote Communication
	UL-2900-2-1	10	Sensitive Data
	ISO 81001-5-1	5.3	Software architectural design
	ISO 81001-5-1	5.4	Software design
	ISO 81001-5-1	7.2	Identification of vulnerabilities, threats, and associated adverse impacts
	ISO 81001-5-1	C	Threat Modeling
D.3. Design & Development – Code Development and Testing	IMDRF Principles and Practices for Medical Device Cybersecurity	5.3	Premarket Considerations: Security Testing
	UL-2900-1	17	Software Weakness Analysis
	UL-2900-1	18	Static Source Code Analysis
	UL-2900-1	19	Static Binary and Bytecode Analysis
	UL-2900-2-1	17	Software Weakness Analysis
	UL-2900-2-1	18	Static Source Code Analysis
	UL-2900-2-1	19	Static Binary and Bytecode Analysis

	ISO 81001-5-1	5.5	Software unit implementation and verification
	ISO 81001-5-1	5.6	Software integration testing
	ISO 81001-5-1	5.7	Software system testing
D.4. Design & Development – Patch/Software Update Planning	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VIII.B	End of Life / End of Guaranteed Support / End of Support
	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VIII.F	Patching
	ISO 81001-5-1	6.1	Establish software maintenance plan
	ISO 81001-5-1	9.1	Software problem resolution process overview
D.5. Design & Development – Secure Transfer to Manufacturing Planning	AAMI TIR97:2019	6	Postmarket management of fielded devices
	ANSI/AAMI SW96 (2023)	10	Product and post-production activities
	ISO 81001-5-1	5.8	Software release
E.1. Verification & Validation – Verify Security Controls	IMDRF Principles and Practices for Medical Device Cybersecurity	5.3	Premarket Considerations: Security Testing
	MDCG 2019-16	3.7	Verification / Validation
	ISO 81001-5-1	5.5	Software unit implementation and verification
	ISO 81001-5-1	5.6	Software integration testing
	ISO 81001-5-1	5.7	Software system testing
E.2. Verification & Validation – Identify Known Vulnerabilities	IMDRF Principles and Practices for Medical Device Cybersecurity	5.3	Premarket Considerations: Security Testing
	UL-2900-1	13	Known Vulnerability Testing
	UL-2900-1	14	Malware Testing

	UL-2900-1	15	Malformed Input Testing
	UL-2900-2-1	13	Known Vulnerability Testing
	UL-2900-2-1	14	Malware Testing
	UL-2900-2-1	15	Malformed Input Testing
E.3. Verification & Validation Testing – Security Validation Testing	IMDRF Principles and Practices for Medical Device Cybersecurity	5.3	Premarket Considerations: Security Testing
	UL-2900-1	16	Structured Penetration Testing
	UL-2900-2-1	16	Structured Penetration Testing
F.1. Maintenance – Surveillance	AAMI TIR 97: 2019	6	Postmarket management of fielded devices
	ANSI/AAMI SW96 (2023)	10	Product and post-production activities
	MDCG 2019-16	5.1	Postmarket surveillance system
	ISO 81001-5-1	7.5	Monitoring the effectiveness of risk controls
	ISO 81001-5-1	9.2	Receiving notifications about vulnerabilities
F.2. Maintenance – Vulnerability and EOL/EOS Management	AAMI TIR97:2019	6	Postmarket management of fielded devices
	AAMI TIR97:2019	7	Retirement/obsolescence
	ANSI/AAMI SW96 (2023)	10	Product and post-production activities
	FDA Postmarket Management of Cybersecurity in Medical Devices	V.C	Maintaining Safety and Essential Performance
	FDA Postmarket Management of Cybersecurity in Medical Devices	VI.A	Assessing Exploitability of The Cybersecurity Vulnerability

	FDA Postmarket Management of Cybersecurity in Medical Devices	VI.B	Assessing Severity of Patient Harm
	IMDRF Principles and Practices for Medical Device Cybersecurity	6.4	Postmarket Considerations: Vulnerability Remediation
	IMDRF Principles and Practices for Medical Device Cybersecurity	6.6	Postmarket Considerations: Legacy Medical Devices
	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VI.B	Identifying a Potential Legacy Technology for Medical Device Manufacturers
	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VIII.B	End of Life / End of Guaranteed Support / End of Support
	MDCG 2019-16	5.2	Vigilance
	UL-2900-2-1	20	Lifecycle Security Processes
	ISO 30111	N/A	
	ISO 81001-5-1	6.2	Problem and modification analysis
	ISO 81001-5-1	7.5	Monitoring the effectiveness of risk controls
	ISO 81001-5-1	9.2	Receiving notifications about vulnerabilities
	ISO 81001-5-1	9.3	Reviewing vulnerabilities
	ISO 81001-5-1	9.5	Addressing security-related issues
F.3. Maintenance – Security Incident Response	AAMI TIR97:2019	6	Postmarket management of fielded devices
	ANSI/AAMI SW96 (2023)	10	Product and post-production activities
	FDA Postmarket Management of Cybersecurity in Medical Devices	V.C	Maintaining Safety and Essential Performance

	FDA Postmarket Management of Cybersecurity in Medical Devices	VI.C	Evaluation of Risk of Patient Harm
	FDA Postmarket Management of Cybersecurity in Medical Devices	VII.A	Controlled Risk of Patient Harm
	FDA Postmarket Management of Cybersecurity in Medical Devices	VII.B	Uncontrolled Risk To Safety and Essential Performance
	IMDRF Principles and Practices for Medical Device Cybersecurity	6.4	Postmarket Considerations: Vulnerability Remediation
	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VIII.F	Patching
	ISO 81001-5-1	9.1	Software problem resolution process overview
	ISO 81001-5-1	9.5	Addressing security-related issues
F.4. Maintenance – Patch / Software Update Deployment	AAMI TIR97:2019	6	Postmarket management of fielded devices
	AAMI TIR97:2019	Annex E	Security incident handling and response
	ANSI/AAMI SW96 (2023)	10	Product and post-production activities
	IMDRF Principles and Practices for Medical Device Cybersecurity	6.5	Postmarket Considerations: Incident Response
	IMDRF Principles and Practices for Medical Device Cybersecurity	Appendix A	Incident Response Roles
	MDCG 2019-16	Annex II	Examples of cybersecurity incidents/serious incidents
	NIST 800-61	N/A	Incident handling and response

	NIS2	23	Reporting obligations
F.5. Maintenance – Customer Security Communication	AAMI TIR97:2019	Annex C	Establishing a coordinated vulnerability disclosure process
	FDA Postmarket Management of Cybersecurity in Medical Devices	IX.	Criteria For Defining Active Participation By A Manufacturer In An ISAO
	IMDRF Principles and Practices for Medical Device Cybersecurity	4.4	Information Sharing
	IMDRF Principles and Practices for Medical Device Cybersecurity	5.5	Premarket Considerations: Labeling and Customer Security Documentation
	IMDRF Principles and Practices for Medical Device Cybersecurity	6.2	Postmarket Considerations: Information Sharing
	IMDRF Principles and Practices for Medical Device Cybersecurity	6.3	Postmarket Considerations: Coordinated Vulnerability Disclosure
	IMDRF Principles and Practices for Medical Device Cybersecurity	Appendix B	Jurisdictional resources for Coordinated Vulnerability Disclosure
	HSCC Managing Legacy Technology Security (HIC-MaLTS)	VII.B	Legacy Devices: Core Practices – Communications
	MDCG 2019-16	4.3	Information to be provided to healthcare providers
	ISO 29147	N/A	Vulnerability Disclosure
	ISO 81001-5-1	A.3	Risk transfer (to customer)
	HSCC Medtech Vulnerability Communication Toolkit	N/A	
	NIS2	23	Reporting obligations
	NIS2	29	Cybersecurity information sharing

Appendix H: Vulnerability Scanning Tools and Descriptions

Vulnerability scanning is a large software tool category that performs automated vulnerability and exposure identification. All vulnerability scanning tools are only as complete as their scope of target and quality of known vulnerability database.

Vulnerability scanners are important for evaluating systems for historical vulnerabilities that could date back many years but are typically regularly updated to also include the most current vulnerabilities. As an encyclopedia of 'everything known to be vulnerable' these tools serve as a foundational floor to baseline securing a system from immediate, known issues.

The following are various forms of vulnerability identification and associated automated identification purposes and techniques:

- Software Composition Analysis (SCA)
  - Software composition analysis (SCA) is the analysis of a body of software and its decomposition into various third-party supplier software modules. SCA tools are focused on discovering known vulnerabilities with CVE assigned to them. Due to the necessary decomposition of software into its smaller components and third-party modules, SCA tools can be employed to generate a Software Bill of Materials (SBOM) and perform continuous Vulnerability Monitoring of the SBOM. Using the module names, versions, and associated suppliers, a manufacturer can become aware of a new vulnerability impacting their product during the late stages of development and perpetually in postmarket. SCA focuses on identifying three types of issues:
    - Known vulnerabilities (third-party software vulnerabilities)
    - Commercial licensing (noncompliance or conflicts between software licenses)
    - End-of-life (software modules that are very out of date or not supported anymore by the software community i.e., open-source)
  - SCA can provide significant value in an automated fashion, but a manual process must be employed to review the results and ensure problem areas are not missed:
    - SCA does not perform any vulnerability identification for the portions of software developed in-house and not derived from a third party.
    - SCA tools must be able to detect and identify third-party software components within the software and match those identified versions against a database of vulnerabilities. Any software components which the tool cannot identify will be excluded from the analysis.
    - Third-party components not linked in software code, such as a situation where the component was in-line compiled, or a substantial amount of code was copied/pasted from the third-party project.
  - A regular SCA scanning, and discovery process can be enforced through policies that require third-party components with a specific severity to be further analyzed. As with any vulnerability scanning solution, the analysis must be performed to contextualize the results and ensure the applicability of vulnerability within the component which has been flagged. For example, possessing a vulnerable module is not uncommon, yet not utilizing the impacted function or feature.
- Network Vulnerability Scanning
  - Network vulnerability scanning is a specific type of vulnerability identification aimed at network services accessible from the source of the network scanner. A scan can be carried out from within a

medical device (internal network) or outside the device against its network interfaces (Wi-Fi/Wired). An internal scan is intended to enumerate service-level vulnerabilities from within perimeter defenses and identify vulnerabilities that could be exploited by a threat actor who successfully penetrates the perimeter of the device, or potentially by insider threats. An external scan's primary purpose is to detect vulnerabilities in the perimeter defenses, such as open ports in the network firewall. In many cases, a network vulnerability scan aims to:

  - Identify open network services (i.e., services available for interaction).
  - Identify what function the service represents (e.g., SMTP, SMB, FTP) based on interrogation of the protocol available during an initial connection (e.g., fingerprinting).
  - Attempt to identify versioning of the service for ease of mapping to known vulnerabilities.
  - In many cases, versioning is not available on a networked service, and the scanner must blindly attempt known vulnerabilities (exploits) against the service which are known to be associated with a particular network port (e.g., an open port of 22 is likely to result in the execution of known SSH vulnerabilities against the target).

- Due to the nature of sending excessive irregular network traffic to an unknown service and blindly (live) attempting exploits solely based on port-level associations, network vulnerability scans may introduce the operational risk of an outage when executed against production assets. Network vulnerability scanning can also be performed in an authenticated or unauthenticated fashion.
  - Authenticated Scans: Authenticated scans allow for a scanner to access network-based assets using remote administrative methods and authenticate using provided system credentials. Once authenticated, additional network services may be available for scanning.
  - Unauthenticated scans: By default, scanners will operate unauthenticated and search for weaknesses in the targeted interfaces accessible without logging into any specific internal/device network. Unauthenticated scans suffer from a lack of versioning and meta-data pertaining to network services that can generally be obtained in authenticated modes, resulting in a higher number of false positives.

- Static Application Security Testing (SAST)
  - Static Application Security Testing (SAST) solutions provide an automated means for identifying, tracking, and remediating problematic coding patterns that go against best practices. SAST works best early in the continuous integration (CI) pipeline, scanning source code, bytecode, or binary code, and programming language dependent.
  - SAST use in the early stages of development results in the identification of issues when they are the least expensive to fix. SAST provides scan results either as static reports, or in an interactive interface that enables tracking runtime behavior per vulnerability through the code and provides guidelines for remediation.
    - Captures the majority of coding pattern-related issues early in development.
    - Ease of automation due to no UI interaction, integration with CI and build servers, bug-tracking solutions, and source repositories.
    - Scans complete source code repositories or specific files; compilation is not required.
    - Lacks context of code while its running, it's possible for one portion of code to be problematic yet another section addresses/mitigates and SAST scanning would not take the holistic view e.g., inability to correlate that sanitization happens in a different component before data is used by the application.
    - No test cases are required, which enables ease of use.
  - SAST should be employed during pre-market software development but can also be run on legacy/postmarket code, patches, and updates. SAST scan engines are often updated like any vulnerability scanning tool where they obtain updated definitions. Regular SAST scanning may help

identify issues in existing code or frameworks when new vulnerabilities are found in the associated programming language. SAST does not require executable code and therefore can be employed very early in the development process, saving time and expenses by fixing bugs early. Large projects that are otherwise hard to completely stand up in a test environment can be scanned without any environment or infrastructure.

- Dynamic Application Security Testing (DAST)
  - DAST provides additional runtime insights to the static source-code analysis. In contrast to SAST, DAST tools evaluate running code with an end-to-end perspective. A DAST tool does not know about code-level measures that have been taken, it does not have source code, and focuses directly on instrumenting the running/live application through its designated user/input interfaces. DAST tools will look at the application's input and output and try to determine whether an attack succeeded based on the pure result. It's possible that security controls exist that block malicious input, and the DAST tool identifies that through a failed attack. Conversely, malicious input from the DAST tool may succeed in executing a specific attack, yet it will be completely unclear to the DAST tool as to why that succeeded or whether any in-place controls were bypassed or simply missing.
  - The nature of DAST tools, whereby the main function is to test an application in runtime, offers several benefits and drawbacks.
    - Fewer false positives due to DAST not scanning the whole application, and vulnerabilities identified were demonstrated by the tool.
    - DAST is not programming-language-specific. DAST doesn't look at source code, bytecode, or assembly code; it checks inputs and outputs.
    - DAST can quickly retest vulnerabilities through ease of reproduction and automation through a DAST test suite. In the event a problem resurfaces, it should show up in the next scan.
    - DAST engines operate based on known attack techniques for various forms of input and also random fuzzing. In some cases, the engine may not successfully profile input and revert to sending random attack data in hopes of a successful outcome.
    - DAST is unable to identify business logic vulnerabilities due to the 'brute force' nature of the engine.
    - DAST can result in undesirable behaviors from the target application due to a lack of context (i.e., submitting a 'contact form' thousands of times, attempting to evaluate the input fields, and unknowingly flooding an inbox).
  - Successful integration of DAST generally requires more setup to ensure user interactions and application functionality is fully available to the tool across all roles, access levels, or views. DAST tests the application through the same interfaces a threat actor has access to, therefore executing real attacks on real workflows, resulting in fewer false positives due to a focus on end-to-end input/output.

- Source Code Review
  - Source Code Review is a form of Static Code Analysis executed by an individual with expertise in the target language and application architecture. Secure Code Review may employ the results of SAST and other automated techniques in concert with human intuition to evaluate business logic and other complex vulnerabilities that tools cannot readily identify. For example, only a subject matter expert can stack SAST findings in a way to demonstrate a more significant vulnerability or outcome. Source Code Review alone could technically duplicate the results of SAST, but large code bases create difficulties in manual review, forcing the involved SMEs to use expertise to focus on specific areas of the codebase likely to be high risk or known to possess vulnerabilities outlined by automated tools.

- Source Code Review could be performed at any point during development where source code can be made available to Product Security to be reviewed manually. At the completion of significant releases, Source Code Review may be employed and is regularly performed.

- Configuration Review
  - Configuration of various components such as third-party software, Infrastructure as Code, and platforms/OS involves a set of relatively unique activities. These components are not something that can be readily assessed through vulnerability scanning or source code review. A configuration review should be performed using tools specific to the component and based upon a best practice standard for securely configuring the target. As with any testing activity, the results of a configuration review are vulnerabilities that require mitigation. The following are various examples of configuration review situations:
    - Infrastructure as Code (IaC): Configuration reviews can be performed automatically to ensure that specific settings are enacted and enforced in code prior to infrastructure deployment. A configuration review may include a series of scanners for IaC that are applicable to that target (e.g., AWS/GCP/Azure/etc.).
    - Platform/OS: Configuration reviews often include scanning the platform for known configuration issues in the file system or application that could result in privileged escalation, tampering with application data, etc. Operating systems are very feature-rich and generally contain substantial functionality that should generally be 'hardened' through disablement or changes that improve the security. Every platform/OS typically contains industry-standard hardening benchmarks that can be employed as the basis for a configuration review. When a hardening standard is employed as a configuration baseline, and if the configuration baseline deviates from the hardening standard, a vulnerability has been identified.

Third-Party Software: Third-Party software may be scanned and analyzed during various testing, but in many cases, it likely contains an applicable hardening standard. As with Platform/OS, a hardening standard may be employed as a configuration baseline, resulting in vulnerabilities when the configuration baseline deviates from the hardening standard.

Appendix I: Example Exploitability Assessment Methods

There are several exploitability scoring systems in use. Some are better suited in evaluating vulnerabilities once a system is designed and implemented, and others are better suited for risk assessment during architecture analysis and early design.

a. MITRE Rubric for Applying CVSS to Medical Devices

CVSS provides a way to characterize and assess the severity of a cybersecurity vulnerability, and the IT industry has used it effectively to manage system and software vulnerabilities for many years. The purpose of this appendix section is to provide additional healthcare context for end users and manufacturers that leverage CVSS as a part of their vulnerability assessment.

CVSS defines the "Exploitability metrics" as a combination of Attack Vector (AV), Attack Complexity (AC), Privileges Required (PR), and User Interaction (UI) all from the Base Metric Group.

CVSS was developed for generic, non-industry specific information technology environments. For this reason, CVSS must be generically applied to medical devices and their intended use within the healthcare sector. A supplemental

CVSS rubric was created to explicitly consider the clinical environment and to help assess potential risks to medical devices. The intent is to use CVSS scoring and associated vector string information to provide a consistent and standardized way to communicate the severity of a vulnerability between multiple parties, including the medical device manufacturer, hospitals, clinicians, patients, Department of Homeland Security (DHS), and vulnerability researchers.

In October 2020, the FDA indicated that this rubric was qualified as a Medical Device Development Tool (MDDT) for use in postmarket vulnerability disclosures.

The MITRE Rubric for Applying CVSS to Medical Devices is found at CVSS Medical Device Rubric.

b. OWASP Risk Rating Methodology

OWASP has developed a risk rating methodology, documented at OWASP Risk Rating Methodology.

From that document, the exploitability measures include:

Threat Agent Factors

The first set of factors are related to the threat agent involved. The goal here is to estimate the likelihood of a successful attack by this group of threat agents. Use the worst-case threat agent.

- **Skill Level** - How technically skilled is this group of threat agents? No technical skills (1), some technical skills (3), advanced computer user (5), network and programming skills (6), security penetration skills (9)
- **Motive** - How motivated is this group of threat agents to find and exploit this vulnerability? Low or no reward (1), possible reward (4), high reward (9)
- **Opportunity** - What resources and opportunities are required for this group of threat agents to find and exploit this vulnerability? Full access or expensive resources required (0), special access or resources required (4), some access or resources required (7), no access or resources required (9)
- **Size** - How large is this group of threat agents? Developers (2), system administrators (2), intranet users (4), partners (5), authenticated users (6), anonymous Internet users (9)

Vulnerability Factors

The next set of factors are related to the vulnerability involved. The goal here is to estimate the likelihood of the particular vulnerability involved being discovered and exploited. Assume the threat agent selected above.

- **Ease of Discovery** - How easy is it for this group of threat agents to discover this vulnerability? Practically impossible (1), difficult (3), easy (7), automated tools available (9)
- **Ease of Exploit** - How easy is it for this group of threat agents to actually exploit this vulnerability? Theoretical (1), difficult (3), easy (5), automated tools available (9)
- **Awareness** - How well known is this vulnerability to this group of threat agents? Unknown (1), hidden (4), obvious (6), public knowledge (9)
- **Intrusion Detection** - How likely is an exploit to be detected? Active detection in application (1), logged and reviewed (3), logged without review (8), not logged (9)

c. NIST SP800-30

Traditional risk management uses Likelihood x Impact. It is understood that likelihood is difficult to estimate since threat actors do not behave in ways that are easily predictable. That is why the FDA uses "exploitability." However, NIST SP800-30 Rev1 – *Guide for Conducting Risk Assessments* (https://www.nist.gov/privacy-framework/nist-sp-800-30) in Appendix G "Determining the Likelihood of Threat Events Causing Adverse Impacts" discusses a framework for estimating likelihood. This is an additional resource for security risk assessment.

d. Exploit Prediction Scoring System (EPSS)

EPSS is a data-driven effort for estimating the likelihood (probability) that a software vulnerability will be exploited in the wild. The goal is to assist network defenders to better prioritize vulnerability remediation efforts. While other industry standards have been useful for capturing innate characteristics of a vulnerability and provide measures of severity, they are limited in their ability to assess threats. EPSS fills that gap because it uses current threat information from CVE and real-world exploit data. The EPSS model produces a probability score between 0 and 1 (0 and 100%). The higher the score, the greater the probability that a vulnerability will be exploited.

[SOURCE: https://www.first.org/epss/]