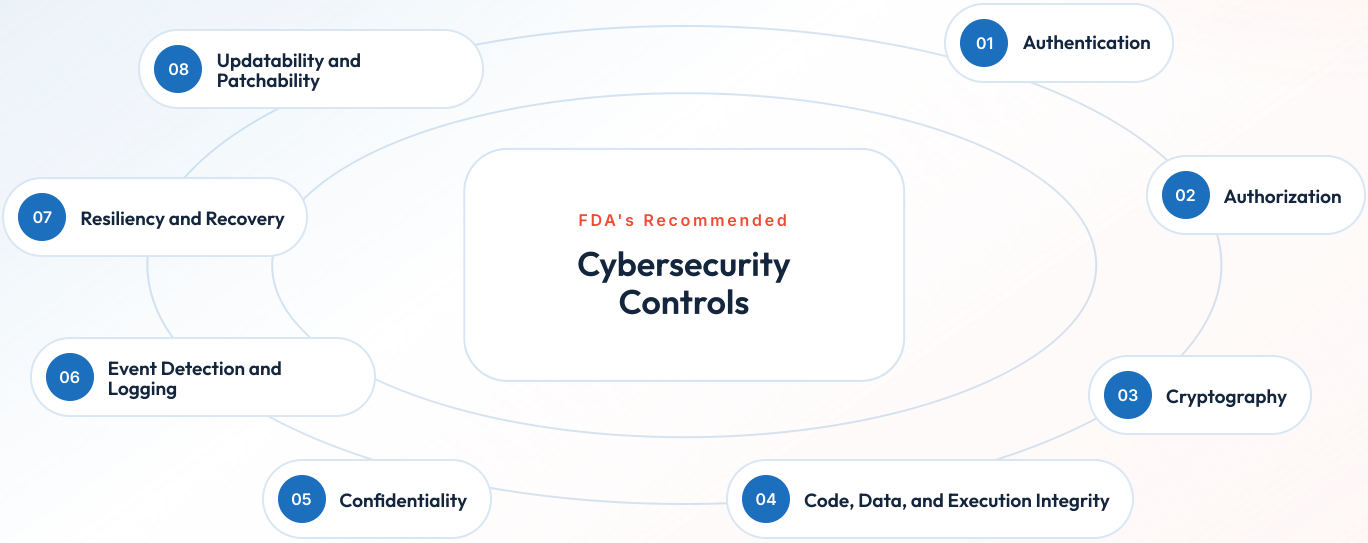
CYBERMED

2026 EDITION

A PLAIN-LANGUAGE GUIDE FOR MEDICAL-DEVICE TEAMS

The eight FDA cybersecurity control categories

A practical introduction for regulatory affairs, quality assurance, product, and engineering leaders.



WHAT IS A SECURITY CONTROL?

A safeguard that changes how a system prevents, detects, limits, or recovers from a cybersecurity event.

WHAT IS A CONTROL CATEGORY?

A group of controls with a related purpose. Each category can contain several device-specific controls.

FDA recommends a risk-based set of controls across the full medical-device system. The implementation depends on the architecture, intended use environment, clinical workflow, and cybersecurity risks.

REFERENCE

1. U.S. Food and Drug Administration. *Cybersecurity in Medical Devices: Quality Management System Considerations and Content of Premarket Submissions.* February 3, 2026. Section V.B.1 and Appendix 1.

Definitions and recommendations are paraphrased from FDA. Examples and RA/QA questions are CyberMed educational synthesis.

TALK WITH CYBERMED

Schedule a call with CyberMed

Scan the QR code or visit:

cybermed.ai/jb30

CONNECT WITH JOSE

linkedin.com/in/jose-bohorquez

info@cybermed.ai

01

SECURITY CONTROL CATEGORY

Authentication

Who or what is this, and where did this information come from?

DEFINITION

Authentication provides evidence that an entity is who or what it claims to be and that information came from a known, trusted source without unauthorized alteration.

IN PLAIN LANGUAGE

Authentication is an identity check. It applies to people, devices, applications, servers, messages, commands, clinical data, software, and updates.

WHY IT IS IMPORTANT

- A device must distinguish a clinician, patient, service technician, connected device, or server from an impostor.
- A receiving system must be able to trust the source of commands, clinical data, and software updates.
- Weak authentication can let an attacker impersonate a trusted person or system and use valid device functions in an unsafe way.

WHAT THIS CATEGORY PROTECTS

People and roles | Devices and services | Messages and software

HOW THE CONTROL WORKS

CLAIM

User • device • server • message

An identity or information source is presented.

PROOF

Credentials • certificates • signatures

The device checks cryptographic or user evidence.

TRUSTED RESULT

Session • command • telemetry • update

The system accepts valid proof and handles failure.

CONTROL 01

Authentication in practice

01

FOUR PRIMARY IMPLEMENTATION APPROACHES

1 User authentication

Passwords, passkeys, smartcards, biometrics, or multi-factor authentication selected for the user, privilege level, and clinical context.

2 Device and service authentication

Certificates or other cryptographic credentials that allow devices, applications, and servers to verify each other.

3 Information authentication

Digital signatures or message authentication codes that verify the source and integrity of messages, telemetry, software, and firmware.

4 Session and replay protection

Session limits, nonces, sequence values, or equivalent measures that prevent reuse of captured valid communications.

MEDICAL-DEVICE EXAMPLES

- **Implant programmer**
  A clinician programmer and an implanted device authenticate each other before the programmer can change therapy settings.
- **Automated insulin delivery**
  A glucose monitor and insulin pump verify the source of glucose readings before the readings influence insulin delivery.
- **Home monitoring**
  A home-use device authenticates the manufacturer cloud service before it uploads data or accepts instructions.
- **Firmware update**
  A device verifies the digital signature on a firmware package before installation.

QUESTIONS RA AND QA CAN ASK

Which people, devices, services, messages, and updates must be authenticated?

What happens when authentication fails or a credential expires?

Could a security step delay urgent clinical access, and how has that risk been addressed?

REMEMBER

Authentication establishes identity and source. Authorization defines the actions available to that identity.

02

SECURITY CONTROL CATEGORY

Authorization

What is this authenticated entity allowed to do?

DEFINITION

Authorization grants or denies an authenticated entity permission to access a resource or perform an action.

IN PLAIN LANGUAGE

Authentication checks identity. Authorization checks permissions. A hospital badge may identify a person, while access rules determine which rooms that person may enter.

WHY IT IS IMPORTANT

- A valid account or trusted device should have access only to the functions and data it needs.
- A compromised low-privilege account should remain separated from clinical configuration, maintenance functions, and system administration.
- Clear permission boundaries limit the impact of mistakes, stolen credentials, and compromised components.

WHAT THIS CATEGORY PROTECTS

Functions | Resources and data | Privilege boundaries

HOW THE CONTROL WORKS

IDENTITY

Authenticated user, device, or process

The system knows which entity is making the request.

POLICY DECISION

Role • least privilege • context

Permissions are evaluated with a deny-by-default posture.

ENFORCED RESULT

Allow approved action • deny other access

Session and privilege limits remain in force.

CONTROL 02

Authorization in practice

02

FOUR PRIMARY IMPLEMENTATION APPROACHES

1 Role or attribute rules

Assign permissions based on roles such as patient, clinician, service technician, and administrator, or on defined contextual attributes.

2 Least privilege

Give each user, process, and component the minimum access needed to complete its assigned task.

3 Deny by default

Reject connections, functions, and requests unless the design expressly permits them.

4 Session and privilege controls

Use timeouts, renewed approval for sensitive actions, separation of duties, or physical signals of intent where appropriate.

MEDICAL-DEVICE EXAMPLES

- **Patient application**
  A patient can view personal therapy history. Clinician-only dosage limits remain outside the patient role.
- **Service access**
  A technician can run diagnostics while patient records that are unrelated to service remain restricted.
- **Drug library**
  An authorized administrator can publish an approved library. A bedside user selects from approved entries.
- **Cloud tenancy**
  A cloud API restricts each healthcare organization to its own devices and data.

QUESTIONS RA AND QA CAN ASK

Is there a documented permission matrix for every role, component, and sensitive function?

Are unnecessary ports, interfaces, and commands denied by default?

Can a low-privilege role reach a function that can affect safety, security, or another customer?

REMEMBER

Authentication proves identity. Authorization limits access to approved resources and actions.

03

SECURITY CONTROL CATEGORY

Cryptography

How does the system establish and protect digital trust?

DEFINITION

Cryptography uses mathematical algorithms, protocols, and keys to protect information, verify authenticity and integrity, and support other security controls.

IN PLAIN LANGUAGE

Cryptography provides special locks, seals, and identity credentials for digital systems. The mathematics matters, and key handling often determines whether the protection works.

WHY IT IS IMPORTANT

- Medical-device systems exchange sensitive data and safety-relevant commands across networks they may not control.
- Cryptography supports trusted connections, protected storage, authenticated updates, and resistance to message tampering.
- The full key lifecycle affects security, including generation, storage, rotation, revocation, and replacement.

WHAT THIS CATEGORY PROTECTS

Data in transit | Data at rest | Credentials and trust

HOW THE CONTROL WORKS

ASSETS

Data • commands • credentials • updates

The design identifies where digital trust is needed.

CRYPTOGRAPHIC SYSTEM

Current algorithms • protocols • protected keys

Keys are generated, stored, rotated, and revoked securely.

PROTECTION

Encrypted data • authenticated exchange

The design resists tampering and unauthorized downgrade.

CONTROL 03

Cryptography in practice

03

FOUR PRIMARY IMPLEMENTATION APPROACHES

1 Current algorithms and protocols

Use recognized, appropriately strong cryptographic standards and avoid deprecated algorithms or configurations.

2 Key lifecycle management

Define how keys are generated, provisioned, stored, used, rotated, revoked, recovered where appropriate, and destroyed.

3 Protected data

Apply suitable cryptographic protection to communications, stored sensitive data, credentials, and configuration.

4 Agility and downgrade resistance

Support safe migration to newer algorithms or protocol versions and prevent unauthorized rollback to weaker protection.

MEDICAL-DEVICE EXAMPLES

- **Device-to-cloud connection**
  A device and cloud service use mutually authenticated transport encryption for each session.
- **Unique device identity**
  Each device receives a unique private key stored in hardware-protected storage instead of sharing one fleet-wide secret.
- **Diagnostic workstation**
  The workstation encrypts locally stored patient data and protects the associated keys separately.
- **Trust update**
  An update system accepts a new signing key through a controlled process before the old key expires.

QUESTIONS RA AND QA CAN ASK

Is there an inventory of cryptographic uses, algorithms, protocols, keys, owners, and expected service lives?

What happens if a key is exposed, a certificate expires, or an algorithm becomes unsuitable?

Does compromise of one device expose credentials that can compromise other devices?

REMEMBER

Cryptography is a lifecycle system for algorithms, protocols, keys, and trust decisions.

04

SECURITY CONTROL CATEGORY

Code, Data, and Execution Integrity

Is the system using approved code, valid data, and an expected running state?

DEFINITION

Integrity controls help ensure that code, data, configuration, and the state of running software remain authentic, complete, valid, and protected from unauthorized change.

IN PLAIN LANGUAGE

Integrity means the device uses approved instructions, trustworthy data, and an expected operating state. Think of checking both the seal on a package and the contents inside it.

WHY IT IS IMPORTANT

- Modified code can change device behavior or add hidden functions.
- Unchanged data can still be malformed, out of range, or unsafe, so integrity and validation must work together.
- A device can start with approved software and still be compromised while the software is running.

WHAT THIS CATEGORY PROTECTS

- Software and firmware
- Inputs and settings
- Runtime state

HOW THE CONTROL WORKS

WHAT MUST REMAIN TRUSTWORTHY

Code • data • configuration • execution state

Each asset has a defined approved or valid condition.

INTEGRITY CHECKS

Signatures • secure boot • validation • monitoring

The system checks source, structure, range, and runtime state.

EXPECTED OPERATION

Approved software • valid inputs • protected execution

Failure blocks use or triggers a defined safe response.

CONTROL 04

Code, Data, and Execution Integrity in practice

04

FOUR PRIMARY IMPLEMENTATION APPROACHES

1 Code integrity

Use authenticated software, secure boot, signed updates, version protection, and signature-based allow-listing where appropriate.

2 Data and configuration integrity

Protect data at rest and in transit, validate structure and safe ranges, and control changes to safety-relevant settings.

3 Execution integrity

Use isolation, runtime protections, monitoring, and careful handling of external input to protect software while it runs.

4 Hardware and physical integrity

Restrict debug ports, use hardware-backed security where appropriate, and apply tamper-evident measures when physical access is relevant.

MEDICAL-DEVICE EXAMPLES

- **Infusion safety limit**
  A pump rejects a drug-library value outside the approved range even when the file arrived through an authenticated channel.
- **Verified startup**
  A device verifies the bootloader, operating system, and application signatures before clinical operation.
- **Medical-image input**
  Software checks file structure, required dimensions, and identifiers before processing an incoming image.
- **Production hardware**
  Production devices disable or restrict JTAG and UART debug access before release.

QUESTIONS RA AND QA CAN ASK

How does the system detect unauthorized changes to software, firmware, configuration, and critical data?

Which external inputs are checked for both integrity and validity?

How are debug interfaces and runtime protections handled in the production configuration?

REMEMBER

Authentic data came from a trusted source. Valid data is also well-formed and within acceptable limits.

05

SECURITY CONTROL CATEGORY

Confidentiality

Who or what is allowed to see this information?

DEFINITION

Confidentiality controls prevent unauthorized disclosure of information whose exposure could enable harm or misuse.

IN PLAIN LANGUAGE

Confidentiality makes sensitive information available only to the people and systems that are allowed to see it.

WHY IT IS IMPORTANT

- Exposed credentials or cryptographic keys can let an attacker impersonate a trusted user, device, or service.
- Disclosed configuration or security information can make attacks easier.
- Medical-device systems may also handle protected health information and other data subject to legal, contractual, or organizational requirements.

WHAT THIS CATEGORY PROTECTS

- Patient information
- Credentials and keys
- Security-sensitive data

HOW THE CONTROL WORKS

SENSITIVE INFORMATION

Patient data • credentials • keys • configuration

Threat modeling identifies disclosure that could enable harm.

PROTECTION CHOICES

Minimize • restrict • encrypt • protect secrets

Controls apply in storage, transit, logs, backups, and support data.

AUTHORIZED DISCLOSURE

Approved people and systems only

Access and retention match the intended purpose.

CONTROL 05

Confidentiality in practice

05

FOUR PRIMARY IMPLEMENTATION APPROACHES

1 Data identification and minimization

Identify sensitive information, collect only what is needed, and define retention and deletion rules.

2 Access restrictions

Use authentication, authorization, separation, and administrative controls to limit access.

3 Encryption

Protect sensitive information in transit and at rest with appropriate key management.

4 Secret protection

Keep passwords, tokens, private keys, and other credentials out of source code, logs, exposed storage, and unnecessary interfaces.

MEDICAL-DEVICE EXAMPLES

- **Protected device key**
  A device stores a private authentication key in a protected hardware element and does not export it.
- **Home monitoring app**
  The application encrypts patient data on the phone and during transfer to the cloud.
- **Diagnostic logs**
  Logs omit or mask access tokens, passwords, and unnecessary patient identifiers before export.
- **Multi-tenant service**
  A cloud service isolates each healthcare organization's data and administrative access.

QUESTIONS RA AND QA CAN ASK

Which information could enable patient harm, unauthorized control, privacy loss, or wider compromise if disclosed?

Where does each sensitive data type exist, and who or what can access it?

Do logs, backups, support packages, and decommissioned devices receive suitable protection?

REMEMBER

Confidentiality covers patient data, credentials, keys, and other information that can enable harm when exposed.

06

SECURITY CONTROL CATEGORY

Event Detection and Logging

How will the team recognize and investigate suspicious activity?

DEFINITION

Event detection and logging allow a medical-device system to recognize, record, time, communicate, and support investigation of suspected or successful security events.

IN PLAIN LANGUAGE

Detection is the alarm. Logging is the record that helps a team understand what happened. Together, they act like a security alarm and a flight recorder.

WHY IT IS IMPORTANT

- Teams need timely awareness before they can respond to suspicious activity.
- Secure records help investigators reconstruct events, determine affected devices, and choose an appropriate action.
- Fleet-level monitoring can reveal patterns that are difficult to see on one device.

WHAT THIS CATEGORY PROTECTS

- Security awareness
- Forensic evidence
- Fleet response

HOW THE CONTROL WORKS

SECURITY SIGNAL

Login • configuration • network • integrity event

The design defines activity that needs attention.

EVIDENCE

Detect • timestamp • record • protect

The record preserves actor, action, result, and device context.

ACTION

Alert • investigate • respond • compare fleet

Teams use trustworthy evidence to determine scope and response.

CONTROL 06

Event Detection and Logging in practice

06

FOUR PRIMARY IMPLEMENTATION APPROACHES

1 **Security-event definition**

Identify events that need detection or recording, such as failed logins, permission failures, configuration changes, and integrity failures.

2 **Useful, trustworthy records**

Include relevant time, actor or component, action, result, and device context while protecting logs against unauthorized access or change.

3 **Monitoring, alerts, and response**

Define which events require local notice, centralized alerting, automated action, or human review.

4 **Retention and forensic access**

Define secure storage, export, rotation, retention, time synchronization, and access for investigation.

MEDICAL-DEVICE EXAMPLES

- **Failed service login**
  A device records repeated failed service-login attempts and alerts the appropriate administrator.
- **Configuration history**
  An infusion system records who changed a safety-relevant setting, what changed, when it changed, and the result.
- **Unexpected connection**
  A connected device detects and reports repeated requests to an unknown external endpoint.
- **Fleet analysis**
  A fleet service compares events across device models to identify a common attack pattern or affected variant.

QUESTIONS RA AND QA CAN ASK

Which security decision will each event record or alert support?

Can the record establish who or what acted, what occurred, when it occurred, and the result?

Are logs protected from tampering and unnecessary collection of sensitive data?

REMEMBER

Detection creates awareness. Logging preserves evidence for investigation and response.

07

SECURITY CONTROL CATEGORY

Resiliency and Recovery

How will critical functions continue and trusted operation be restored?

DEFINITION

Resiliency allows a system to preserve critical functions and data during adverse conditions. Recovery returns the system to a known, trusted state after disruption or compromise.

IN PLAIN LANGUAGE

Resiliency describes how the device keeps essential work going when something fails or is attacked. Recovery describes how it returns safely to normal.

WHY IT IS IMPORTANT

- Medical-device availability can affect diagnosis, monitoring, therapy, and clinical workflow.
- A long service life creates a need to limit incident impact and support safe restoration.
- Connectivity loss, denial-of-service conditions, scanning, resource exhaustion, and partial compromise can affect device performance.

WHAT THIS CATEGORY PROTECTS

- Clinical continuity
- Critical functions
- Trusted restoration

HOW THE CONTROL WORKS

ADVERSE CONDITION

Outage • overload • scanning • partial compromise

The system encounters disruption or hostile activity.

RESILIENT OPERATION

Contain • isolate • limit load • degrade safely

Critical functions and data receive continued protection.

RECOVERY

Restore a known-good configuration and state

An authenticated, authorized process returns trusted operation.

CONTROL 07

Resiliency and Recovery in practice

07

FOUR PRIMARY IMPLEMENTATION APPROACHES

1 Containment and isolation

Separate components and privileges so one compromise does not control the full system.

2 Defined degraded operation

Specify what each component can do safely when communications or dependent services are unavailable.

3 Availability protection

Use resource limits, rate limiting, redundancy, failover, watchdogs, or other risk-appropriate protections.

4 Trusted recovery

Retain or restore known-good configuration, software, and data through authenticated and authorized procedures.

MEDICAL-DEVICE EXAMPLES

- **Network loss**
  An infusion pump continues an already verified therapy safely and clearly reports the loss of connectivity.
- **Monitoring failover**
  A monitoring system uses failover so one unavailable server does not stop all bedside monitoring.
- **Resource protection**
  A cloud-connected device limits excessive requests so scanning cannot exhaust resources needed for clinical functions.
- **Trusted configuration**
  An authorized technician can restore trusted defaults through a controlled recovery procedure.

QUESTIONS RA AND QA CAN ASK

Which critical functions must continue during network loss, service failure, overload, or partial compromise?

What is the defined safe or degraded behavior for each component?

How does the team confirm that a recovered device is in a trusted state?

REMEMBER

Resiliency limits disruption during an event. Recovery restores trustworthy operation afterward.

08

SECURITY CONTROL CATEGORY

Updatability and Patchability

How will the manufacturer correct vulnerabilities throughout the supported lifecycle?

DEFINITION

Updatability and patchability allow a manufacturer to develop, verify, distribute, install, and confirm software or firmware changes securely and in time to address cybersecurity risk.

IN PLAIN LANGUAGE

A secure device must be serviceable. The update path is a protected route for repairing software after deployment while preserving trusted operation.

WHY IT IS IMPORTANT

- Previously unknown vulnerabilities can emerge after release, and threats can change during a long service life.
- A patch provides protection when the manufacturer can build, test, distribute, and install it in time.
- A failed, interrupted, counterfeit, or unauthorized update can make a device unavailable or change its behavior.

WHAT THIS CATEGORY PROTECTS

Fielded devices | Update trust chain | Supported lifecycle

HOW THE CONTROL WORKS

NEED
Vulnerability identified
The manufacturer determines affected products and versions.

PREPARE
Build • test • sign • distribute
Preserved tools and controlled processes support a timely patch.

APPLY
Verify • install • recover from interruption
The device accepts an approved package and preserves safe operation.

CONFIRM
Track fielded versions and deployment
The manufacturer can identify devices that remain exposed.

CONTROL 08

Updatability and Patchability in practice

08

FOUR PRIMARY IMPLEMENTATION APPROACHES

1 Lifecycle readiness

Reserve the storage, processing capacity, build tools, test environments, third-party licenses, and support processes needed for future updates.

2 Secure update chain

Authenticate and authorize update actions, protect distribution, verify the package and metadata, and accept only approved updates.

3 Reliable installation

Define behavior for interrupted transfer, failed verification, power loss, rollback, and recovery.

4 Timely deployment and visibility

Support urgent security releases, track fielded versions, measure deployment, and communicate user actions.

MEDICAL-DEVICE EXAMPLES

- **Connected pump**
  The pump verifies a signed package, preserves the trusted version until installation succeeds, and recovers safely after interruption.
- **Mobile medical app**
  The application uses the platform signed-update channel while the manufacturer tracks supported app and operating-system versions.
- **Cloud medical-device function**
  A controlled pipeline uses staged release, health monitoring, and tested rollback for urgent security fixes.
- **Clinic programmer**
  The programmer receives and verifies an update before it can deliver the approved package to an implanted device.

QUESTIONS RA AND QA CAN ASK

Can the team reproduce, test, sign, and release an update near the end of the supported life?

How does the device respond to an invalid, interrupted, failed, or older update?

How will the manufacturer know which fielded versions remain vulnerable and whether users completed the update?

REMEMBER

Patchability is an end-to-end lifecycle capability that connects engineering, quality, deployment, and field support.

PUTTING IT TOGETHER

Eight categories. One security architecture.

A medical-device system needs controls that work together across people, devices, software, services, data flows, and the supported lifecycle.

01 Authentication
Who or what is this, and where did this information come from?

02 Authorization
What is this authenticated entity allowed to do?

03 Cryptography
How does the system establish and protect digital trust?

04 Code, Data, and Execution Integrity
Is the system using approved code, valid data, and an expected running state?

05 Confidentiality
Who or what is allowed to see this information?

06 Event Detection and Logging
How will the team recognize and investigate suspicious activity?

07 Resiliency and Recovery
How will critical functions continue and trusted operation be restored?

08 Updatability and Patchability
How will the manufacturer correct vulnerabilities throughout the supported lifecycle?

A PRACTICAL REVIEW SEQUENCE

1. Start with the system architecture, threat model, and cybersecurity risk assessment.
2. Select and implement controls based on the device and its risks.
3. Convert controls into clear design requirements and measurable acceptance criteria.
4. Trace risks to controls, requirements, tests, and results.
5. Test each control in its intended security context and document the evidence.
6. Reassess effectiveness as the device, environment, threats, and fielded versions change.

PRIMARY SOURCE

U.S. Food and Drug Administration. *Cybersecurity in Medical Devices: Quality Management System Considerations and Content of Premarket Submissions.* February 3, 2026. Section V.B.1 and Appendix 1.

FDA guidance contains nonbinding recommendations. Applicable statutory and regulatory obligations remain separate. Use device-specific risk analysis and qualified professional judgment.

Read the FDA guidance

TALK WITH CYBERMED

Schedule a call with CyberMed

Scan the QR code or visit:

cybermed.ai/jb30

CONNECT WITH JOSE

linkedin.com/in/jose-bohorquez
info@cybermed.ai