FRANK LEITNER CYBERSECURITY
SECURE BY DESIGN, TRUSTED IN PRACTICE

EU Cyber Resilience Act: What Product Teams Should Do Now

The EU Cyber Resilience Act enforcement begins December 2027, less than two years away. For manufacturers of software products, the preparation window is closing rapidly. Yet many product organizations are still in the early stages of understanding what compliance actually requires.

The CRA fundamentally changes how software-driven products are brought to market in the EU. It mandates secure-by-design practices, vulnerability handling processes, and demonstrable security throughout the product lifecycle. For product teams, this is not a documentation exercise. It is a multi-stream engineering program: building management systems, and embedded control products, these aren't minor adjustments, they require substantial technical implementation.

This article outlines a realistic 18-month preparation roadmap based on my experience helping product teams prepare for regulatory requirements. Even this timeline represents minimum viable preparation. Organizations with complex product portfolios, legacy systems, or immature security processes should expect to need significantly more time.

CRA doesn't ask you to improve your security. It asks you to prove it.

Timeline Overview

Here's what realistic CRA preparation looks like, starting today. The sections below explain what each phase actually involves. Starting in March 2026, this roadmap targets completion by September 2027, providing a three-month buffer before December 2027 enforcement. The timeline looks like this:

- **Months 1-2:** focus on scope assessment with the critical product inventory.
- **Months 2-4:** handle gap analysis requiring cross-functional assessment.
- **Months 3-8:** establish the vulnerability process including legal review and infrastructure.
- **Months 4-10:** implement SBOM generation with automation tooling.
- **Months 5-12:** enhance the SDLC through tool integration and training.
- **Months 10-14:** create the documentation framework and templates.
- **Months 14-16:** prepare for conformity assessment through internal audit.
- **Months 16-18**: provide final preparation and buffer for unexpected issues.

Many activities run in parallel. SBOM implementation overlaps with vulnerability process establishment. SDLC enhancement occurs alongside most other activities. However, dependencies exist: you can't document processes that don't exist yet, and automation requires process clarity.

Starting today puts you on the critical path. Starting later removes your margin for error.

Understanding Your Product's CRA Scope (Month 1-2)

Before investing effort in compliance activities, you need clarity on which products actually fall under CRA scope. This seems straightforward but often isn't.

The CRA applies to products with digital elements that connect to networks or other devices. For industrial product portfolios, this typically includes industrial automation controllers, building automation systems, embedded software in industrial equipment, IoT gateways and edge devices, and products with remote access or update capabilities.

However, gray areas exist. Pure cloud services have different treatment than products with digital elements. Component manufacturers face different obligations than system integrators. Products solely for internal use within a company may be out of scope. Critical products listed in CRA Annex III face additional third-party conformity assessment requirements.

For organizations with ten or more product lines, this assessment alone can take six to eight weeks. You need to examine each product's architecture, connectivity, update mechanisms, and intended use. Don't rush this step: misclassifying products wastes resources through unnecessary compliance work or dangerous non-compliance (and rushed last minute compensation attempts).

Create an inventory of your product portfolio with preliminary scope assessment and risk prioritization. Document your reasoning for scope decisions, particularly for borderline cases. This documentation becomes important if you later face questions from market surveillance authorities.

Misclassifying products wastes resources or creates dangerous non-compliance.

Conduct a CRA-Focused Gap Analysis (Month 2-4)

Once you know which products require CRA compliance, assess how far your current practices fall from the requirements. The CRA's essential cybersecurity requirements in Annex I cover

- secure development processes
- vulnerability handling
- secure default configurations
- protection against unauthorized access
- secure update mechanisms
- logging and monitoring capabilities.

A thorough gap analysis examines your secure development lifecycle maturity, vulnerability management and disclosure processes, supply chain security and software bill of materials capabilities, and security documentation and conformity assessment readiness.

If you've already implemented IEC 62443 for industrial security, you have a head start. The standards overlap significantly in areas like security requirements specification, threat modeling, and secure development practices. However, CRA adds specific obligations around vulnerability disclosure timelines, SBOM provision, and end-of-support declarations that may not be covered by existing IEC 62443 implementations.

This assessment takes time because it requires input from multiple teams. You'll need to interview development leads about current security practices, product managers about vulnerability handling, IT and DevOps about build systems and component tracking, and legal and compliance teams about documentation and disclosure processes.

Map your current processes against CRA Annex I requirements systematically. For each requirement, document current state, gap severity, implementation effort, and dependencies. Prioritize gaps by their impact on compliance and the effort required to close them. Some gaps are quick wins that build momentum. Others require months of infrastructure work and should start immediately.

> *Some gaps are quick wins. Others require months of infrastructure work and should start immediately.*

Establish Your Vulnerability Handling Process (Month 3-8)

For many industrial product organizations, vulnerability handling represents the largest gap. Historically, industrial products operated in isolated networks with infrequent updates. The CRA requires a completely different approach, it is a regulated obligation with defined expectations around responsiveness and communication and no longer coluntary best practice.

You need a complete vulnerability handling process that includes intake mechanisms for vulnerability reports from external researchers, internal security teams, and automated scanning tools. Triage procedures must assess severity and exploitability, typically using CVSS scoring. You need coordinated disclosure timelines that meet CRA's requirements, including acknowledging reports within a reasonable timeframe and providing remediation within appropriate periods based on severity.

The process must include patch management and update mechanisms that can deliver security fixes to deployed products, customer communication channels for security advisories and notifications, and a CSIRT or designated security contact point visible to researchers and customers.

Many organizations consider adopting CSAF (Common Security Advisory Framework) or VEX (Vulnerability Exploitability eXchange) formats early. These standardized formats facilitate automated vulnerability information exchange and are increasingly expected by enterprise customers.

This implementation requires significant infrastructure. You need ticketing systems to track vulnerability reports, secure communication channels for coordinated disclosure, advisory publication platforms (often integrated with product documentation sites), and potentially bug bounty or vulnerability rewards program infrastructure.

Legal review is essential. Your vulnerability disclosure policy must balance transparency with liability concerns. Your legal team needs to review disclosure timelines, communication templates, and coordination procedures with affected third parties.

Draft your vulnerability disclosure policy and publish it prominently. Assign clear roles for vulnerability intake, triage, remediation, and disclosure. Establish intake mechanisms such as security email addresses and web forms. Conduct dry-run scenarios with intentionally introduced test vulnerabilities to validate your process end-to-end before you need it for real incidents.

The five-month timeline for this work is realistic only if you have dedicated resources. Part-time effort easily extends this to six or eight months.

Reality: Without intake, triage, remediation, and disclosure infrastructure, you don’t have compliance

Get Your SBOM House in Order (Month 4-10)

Software bill of materials generation is consistently underestimated by organizations new to the practice. The CRA requires providing SBOMs for products with digital elements, enabling customers to understand supply chain risks and component vulnerabilities.

Legacy products often lack complete component visibility. Build systems may not systematically track all dependencies. Third-party commercial components may come without their own SBOMs, requiring negotiation with suppliers. Manual SBOM generation might work for a handful of products but doesn't scale to continuous delivery of dozens of product variants.

Take a phased approach. Start with a pilot using one representative product. Manually identify all software components including open source libraries, commercial third-party components, internally developed modules, and system dependencies. Choose an SBOM format, SPDX and CycloneDX are the leading standards, each with different tooling ecosystems and strengths. Generate your first SBOM manually to understand the challenges, gaps in your component knowledge, and tooling requirements.

This manual pilot typically takes four to six weeks but provides invaluable learning. You'll discover undocumented dependencies, components without clear version tracking, and third-party elements that require supplier engagement.

Following the pilot, establish tooling and automation. Research and select SBOM generation tools appropriate for your technology stack. Procure and configure these tools, which may require budget approval cycles. Integrate SBOM generation into your CI/CD pipelines so every build automatically produces an SBOM. Implement validation and quality checks to catch missing or incorrect component information.

This automation phase takes three to four months for most organizations. Build system integration requires dedicated development time. Different products may need different tooling approaches. Testing and validation across your product portfolio reveals edge cases and exceptions.

Generate your first SBOM manually, then build an automation roadmap based on lessons learned. Don't assume vendor promises about automated SBOM generation will work without significant configuration and validation effort.

Reality: SBOM automation is build system work, not documentation work.

Enhance Your Secure Development Lifecycle (Month 5-12)

The CRA requires demonstrable secure-by-design practices throughout the product lifecycle. This goes beyond security as a final testing gate to security integrated into every development phase.

Key SDLC enhancements include:

- security requirements in product specifications from the earliest planning stages
- threat modeling integrated into design phase for new features and products
- automated security testing in CI/CD pipelines through static analysis, dynamic analysis, and dependency scanning
- security-focused code review practices and training
- security testing and validation before release with documented results.

Implementing these enhancements requires bridging security and development teams. Embed security requirements directly into existing sprint and development workflows rather than creating parallel security processes that teams will bypass. Conduct regular threat modeling sessions with architects and senior developers to build security thinking into design culture. Avoid positioning security as gatekeeping, frame it as risk-informed decision making that keeps products viable in regulated markets.

This cultural and process shift takes six months minimum. You need to select and implement security tooling, which requires evaluation, procurement, and integration. Teams need training on threat modeling, secure coding practices, and security testing interpretation. Initial implementations will be clumsy and slow: processes need refinement based on real experience. Tool configurations require tuning to reduce false positives while catching real issues.

Start by assessing current SDLC maturity against secure development frameworks like IEC 62443-4-1 or NIST SSDF. Select two or three priority enhancements with the highest compliance impact and feasibility. Pilot these enhancements with one team before attempting broader rollout. Document what works and what doesn't, then refine your approach before scaling.

Trying to implement everything simultaneously across all teams will fail. Phased, iterative improvement succeeds where big-bang transformations collapse.

Reality: Security integrated into every sprint beats a security gate at the end, every time.

Prepare Your Documentation Framework (Month 10-14)

The CRA requires demonstrating compliance through technical documentation. You can't retrofit this documentation at the end, it must reflect actual security activities throughout development.

Start documenting systematically now. Security risk assessments and threat models for each product should be documented with identified threats, risk ratings, and mitigation decisions. Secure SDLC evidence includes tools used, process descriptions, and training records showing teams are actually following secure practices. Vulnerability handling records should log every report, assessment, and resolution from the moment you establish the process. Third-party component analysis needs documented risk decisions about component selection, version choices, and known vulnerability acceptance. Security testing results from each release should be retained as evidence of verification activities.

Don't wait until conformity assessment to discover you lack evidence of activities you actually performed but didn't document.

Create reusable templates that scale across products. A threat model template ensures consistent analysis across different products. A security test report template standardizes evidence collection. A risk assessment template facilitates comparable decision documentation.

Establish clear documentation ownership and review cycles. Each product team should know who maintains security documentation, where it's stored, when it's reviewed and updated, and how it integrates with conformity assessment needs.

This documentation infrastructure takes three to four months to establish properly. Templates need creation and refinement, teams need training on their use, storage and retrieval systems need implementation, and initial documentation for existing products needs creation.

Reality: Six months of undocumented development creates a six-month compliance gap.

Conformity Assessment Preparation (Month 14-16)

With technical implementation largely complete, prepare for formal conformity assessment. Most products will follow self-assessment under Module A of the CRA. Critical products listed in Annex III require third-party assessment by notified bodies.

Conduct an internal readiness review. Systematically verify gap remediation for each item identified in your initial gap analysis. Check documentation completeness against CRA requirements. Conduct an internal audit or dry-run assessment simulating the rigor of external review.

This review invariably uncovers remaining issues. A process documented on paper may not be consistently followed in practice. Evidence may exist but not be easily retrievable. Technical implementations may not fully meet requirements upon close inspection.

If your product requires third-party assessment, begin notified body engagement early. Research and select a notified body with expertise in your product domain. Understand their specific requirements, timelines, and costs. Submit preliminary documentation to identify gaps before formal assessment.

Notified bodies are experiencing high demand as CRA enforcement approaches. Capacity constraints alone may become a bottleneck independent of your technical readiness. Waiting until month sixteen to initiate contact may mean you can't secure assessment capacity before the deadline.

Complete your internal compliance review, address any remaining gaps, and initiate notified body discussions if applicable. Budget both time and money for this phase, external assessments are neither quick nor inexpensive.

Notified bodies won't wait for you. Neither will the deadline.

Final Preparation and Buffer (Month 16-18)

The final months before your target completion date focus on conformity documentation finalization and preparing for market requirements.

Prepare your EU Declaration of Conformity, the formal statement that your product meets CRA requirements. Ensure CE marking readiness with proper labeling and documentation. Plan customer communications about your CRA compliance, security update commitments, and vulnerability disclosure process. Complete final documentation review, ensuring everything is current, accurate, and accessible.

Critically, maintain buffer time. Unexpected issues always emerge in final stages. A supplier might not provide needed component documentation. A notified body might request additional evidence. An internal audit might reveal a process gap. Market surveillance authorities might publish new guidance requiring adjustments.

Organizations that plan to finish exactly at the deadline will miss it. Those who plan to finish months early might actually finish on time.

> *Plan to finish months early. You might actually finish on time.*

Why This Timeline Is Still Aggressive

This eighteen-month timeline assumes dedicated resources. If your security team handles CRA preparation alongside other responsibilities, or if development teams contribute only part-time, expect significant timeline extension. Full-time equivalent resource allocation matters enormously.

The timeline assumes moderate complexity. Organizations with dozens of legacy products, distributed development teams across multiple sites, or very immature security processes need substantially more time. Products with complex supply chains or novel technologies face additional challenges.

Parallel activities have dependencies that constrain true parallelization. SBOM automation requires SDLC process clarity about build systems and component management. Documentation requires completed implementations to document. Conformity assessment requires everything else to be finished.

Many organizations will need twenty-four months or more for comprehensive preparation. If you're reading this in 2026 and haven't begun the process, you're already on the critical path with no room for delays.

> *Reality: Part-time CRA preparation delivers part-time results, with a fixed deadline.*

Common Pitfalls to Avoid

Don't treat CRA purely as a legal or compliance exercise, and leadership cannot outsource this to legal departments or certification bodies. It requires significant technical implementation that legal teams can't accomplish alone. Engineering, product management, and security teams must drive the technical work.

Don't underestimate SBOM automation effort. Manual approaches might work for a few products but don't scale to continuous delivery of multiple product variants. Plan for substantial build system integration work.

Identify what can proceed in parallel versus what has hard dependencies. Sequential execution of every step extends your timeline beyond the available window. Some tasks must finish before others begin, but many can overlap with proper planning.

If you've already implemented IEC 62443, leverage that work. The standards overlap significantly in secure development, threat modeling, and security architecture. Don't duplicate effort, but also recognize CRA adds specific requirements around vulnerability disclosure and SBOM that may not be covered.

Don't wait for final clarity on every regulatory detail. Core obligations are clear enough to start implementation. Using uncertainty as an excuse for delay means you'll still lack clarity but now also lack time.

Finally, ensure sufficient resource allocation. This work requires dedicated time from development, security, and product management teams. Adding CRA preparation on top of existing full workloads guarantees both missed deadlines and poor quality outcomes.

Reality: CRA is a technical program. Not a compliance checkbox.

Moving Forward

CRA readiness is achievable with structured, realistic preparation - but it requires starting immediately. This eighteen-month timeline represents minimum viable preparation for organizations beginning today. Don't underestimate the effort required.

Organizations starting now can meet the December 2027 deadline, but delays compound rapidly. A two-month slip in gap analysis delays everything that follows. Infrastructure procurement delays ripple through dependent activities. Resource constraints stretch every phase beyond initial estimates.

If you're unsure whether your current roadmap realistically meets December 2027, a focused gap assessment now is significantly cheaper than late remediation in 2027. Reach out for a realistic assessment of your specific timeline needs.

The teams that will succeed are those that treat CRA preparation as a significant technical program requiring dedicated resources, executive support, and realistic planning. Those that treat it as a compliance checkbox to address later will find themselves scrambling in late 2027 with insufficient time to implement required technical controls.

> *The December 2027 deadline is fixed, your preparation timeline should reflect that reality.*

About the author

Frank Leitner is a product security specialist focused on Secure-by-Design development practices, structured threat modeling, and the implementation of ISA/IEC 62443 in software-driven and connected products.

He supports organizations in embedding security into their development lifecycle and translating regulatory expectations, such as those introduced by the EU Cyber Resilience Act, into structured engineering processes, traceable requirements, and verifiable technical capabilities.

His work centers on aligning security architecture, product management, and engineering disciplines to enable sustainable, compliance-ready product development.

Website: https://www.frank-leitner.com