



PRACTICAL GUIDE

From Software Release to CRA Evidence

A practical guide for software manufacturers building durable, release-bound technical evidence for the Cyber Resilience Act.



CONTEXT

CRA is already operational - and evidence work should start now.

Article 14 reporting obligations are already in force. The broader CRA requirements apply from 11 December 2027. The useful question for software teams is not only what the law requires, but what technical facts and decisions they can still prove later.



Why evidence matters

The CRA requires manufacturers to operate product security across design, development, vulnerability handling, maintenance and the support period. For software teams, the practical problem is whether they can still show what was released, what was known, what was decided and what was later verified.

A new scan performed years later answers a different question from the evidence that actually existed for a historical release.



A Release context

Bind evidence to the exact product, release, commit and build artifact.



B Operational history

Keep provider facts, human decisions, remediation and verification separate.



C Durable proof

Retain evidence packages, fingerprints, signatures and verification history.



START WITH CONTEXT

Is your software in scope - and what kind of product is it?

CRA scope is not just a software inventory exercise. Manufacturers need an explainable position on product scope, economic-operator role, product classification and intended use.

1

Product with digital elements?

Software/hardware and relevant remote data processing.



2

Commercially made available on the EU market?

Internal-only and some non-commercial scenarios differ.



3

Any exclusion or sector-specific regime?

Check product-specific exclusions before proceeding.



4

Which economic-operator role?

Manufacturer, importer, distributor - or review required.



5

Which product classification?

Regular, Important Class I, Important Class II or Critical.

**Decision-support, not automatic legal classification**

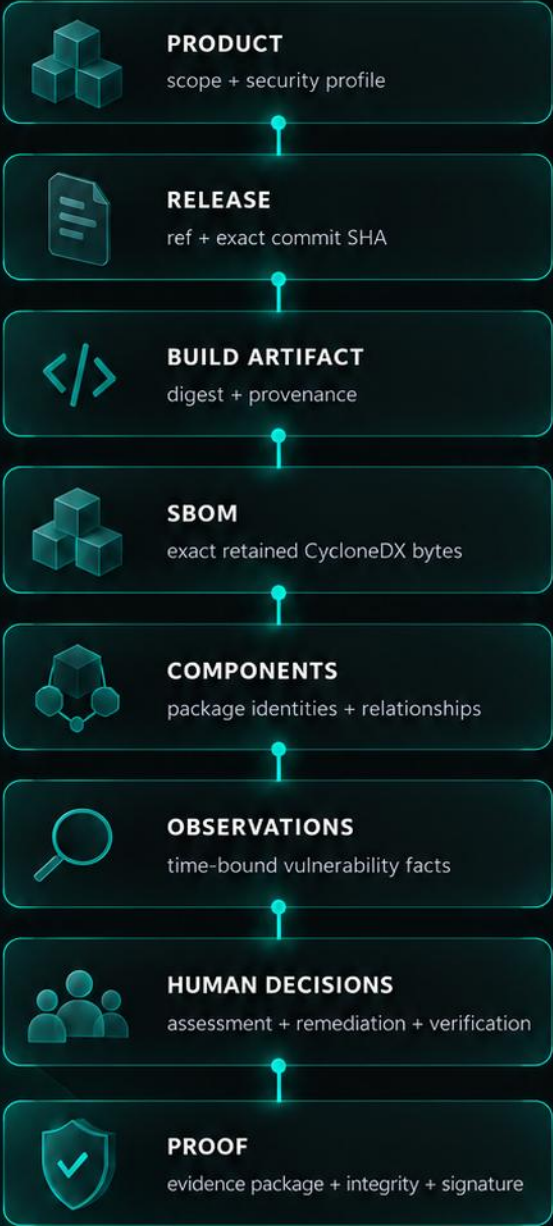
Use structured questions to surface likely scope and classification, but retain an authorised human rationale for confirmed decisions. Intended purpose, reasonably foreseeable use, operating environment and assets should remain versioned product context.



RELEASE EVIDENCE

What should a software team be able to reconstruct?

A useful evidence model follows the product and release - not a generic compliance checklist.



01

What exactly was released?

Repository, commit, artifact identity and SBOM.



02

What was known at the time?

Retained provider observations in that release context.



03

What did people decide?

Assessment, rationale, remediation, verification and approval.



04

What can be proven later?

Exact retained package, fingerprint, signature and verification history.



DESIGN PRINCIPLE

Evidence should remain understandable even after the current product state has changed.

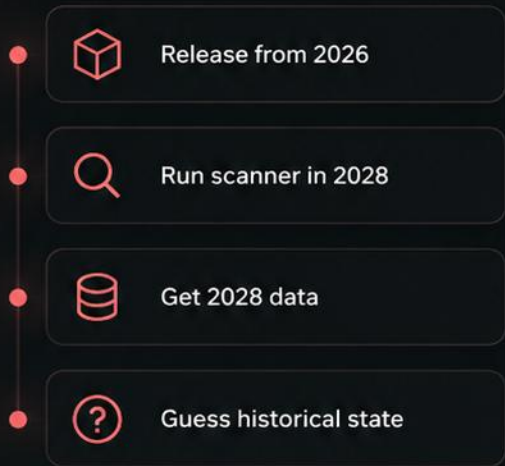
A CORE DISTINCTION

Preserve what existed. Do not recreate history later.

Historical evidence and a current scan are both useful - but they answer different questions.

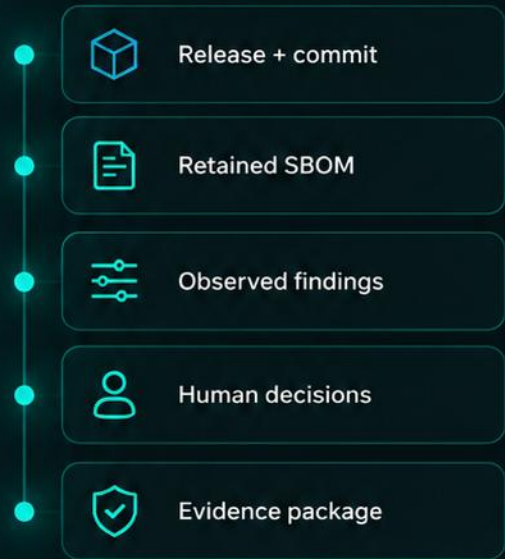
RECONSTRUCTION LATER

Run today's scanner against yesterday's release



RETAINED EVIDENCE

Keep the evidence that actually existed



At Release ≠ Current Security State

A release may gain new vulnerability observations, remediation evidence or reporting cases months later. Keep those current facts visible without rewriting what was known at release time.

FROM FINDINGS TO ACTION

Vulnerability operations and Article 14 are connected - but they are not the same workflow.

Provider data supplies technical facts. Human decisions determine affectedness, remediation and whether an Article 14 case should be opened.

VULNERABILITY OPERATIONS



ARTICLE 14 CASE



Important: in the Netherlands, Article 14 reporting has applied since 11 September 2026. An ordinary vulnerability/CVE is not automatically reportable; active exploitation or severe-incident criteria matter.

RELEASE EVIDENCE CHECKLIST

Can your team answer these questions today?

This is an evidence-readiness checklist, not a legal compliance score.

- ☐ **01** Can you identify the exact commit for a shipped release?
- ☐ **02** Can you identify the distributed build artifact and its digest?
- ☐ **03** Do you retain the exact SBOM used for that release?
- ☐ **04** Can you show which vulnerability observations existed for that release?
- ☐ **05** Can you show who assessed a finding, when and why?
- ☐ **06** Can you show the remediation plan and later verification evidence?
- ☐ **07** Can you distinguish At Release from Current Security State?
- ☐ **08** Is the product support-period decision documented and versioned?
- ☐ **09** Can you point to the cybersecurity risk assessment and technical documentation?
- ☐ **10** Can you retrieve an immutable evidence package and verify its integrity/signature later?



If several answers depend on manually reconstructing data across Git, CI/CD, security tools, tickets and documents, the underlying problem is evidence operations - not another vulnerability scanner.

NEXT STEP

Keep your security tools. Prove your releases.

CRA Operations is designed as a release-evidence layer above the engineering and security tools your team already trusts.



01

CONNECT

Bring in source, release, CI, SBOM and security evidence.

GitHub and GitLab supported.
GitHub Actions and GitLab CI evidence intake.



02

PRESERVE

Retain exact evidence, provenance and human decisions in release context.

No reconstructed history and no generic compliance score.



03

PROVE

Generate immutable evidence packages with integrity, signatures and verification history.

Technical proof stays separate from legal conclusions.



CRA OPERATIONS

Every release. Every vulnerability. Every decision. Proven.

Talk to SuperSimple about a pilot, or use this guide as a starting point for your release-evidence model.

cra@supersimple.nlcra.supersimple.nl

Sources

Regulation (EU) 2024/2847 (Cyber Resilience Act); Nationaal Cyber Security Centrum, "Melden onder de Cyber Resilience Act"; Ministerie van Economische Zaken, Gids Cyber Resilience Act v2.0.

This guide is practical product-security information only. It does not constitute legal advice, certification or a determination of CRA conformity. The regulation and competent authorities remain authoritative.