DLB.ONE

TRAINING MATERIALS FOR IT / OT / EMBEDDED ENGINEERS

CYBERSECURITY 2026 / 2027

NIS2, the Cyber Resilience Act
and the security of test systems

5 modules • 6 hours • engineering practice

Source: www.dlb.one

Edition: 20 November 2026 | Training handbook

PARTICIPANT GUIDE

How to use this training

This is a complete resource for conducting a one-day workshop. It connects legal requirements with engineering decisions relating to electronics, firmware, network infrastructure and test systems. In every module, we distinguish **a legal requirement from an example of a technical implementation**.

> **Scope and limitations.** The legal position used in this document is as of **20 November 2026**. This is educational material, not individual legal advice or a certificate of compliance. Before implementing the recommendations, determine the organisation's status and the product classification, and verify current legislation and harmonised standards.

Training objectives

- Recognise when NIS2/the Polish National Cybersecurity System Act (KSC) applies to an **organisation**, and when the CRA applies to a **product with digital elements**.
- Perform a risk assessment for a test device, an OT network and a firmware update process.
- Establish an evidence trail: architecture, requirements, tests, versions, SBOM and risk decisions.
- Design monitoring, event classification, incident response and operational recovery.
- Organise CRA conformity assessment and customer verification without confusing them with an information security management system audit.

One-day agenda

Module	Scope	Duration
I	Legal framework, scope, roles and Technical File	75 min
II	Security by Design in hardware and software	85 min
III	Supply chain, SBOM/HBOM and open source	70 min
IV	Detection, incidents and continuity	70 min
V	CE, technical verification and audit	60 min

Total: **360 minutes of active work**, excluding breaks. Suggested breaks: two 15-minute breaks and a 30-minute lunch break.

Case study running throughout the training

ST-24 — test station (fictional example). An ATmega4809 controller collects sensor signals; an ESP32-C6 communicates with the station computer and server; and a Linux/Windows computer displays results. Measurement records and part identifiers are transmitted. Service technicians update firmware, connect through USB/UPDI and sometimes use remote access.

> **Attack scenario.** An attacker obtains access to a service account, installs modified firmware or changes reported results. The machine continues operating physically, but test history can no longer be trusted. Potential consequences include loss of traceability, incorrect quality decisions, downtime and a customer dispute.

ST-24 architecture overview

Zone	Components	Critical control point
OT / device	Sensors → ATmega4809 → ESP32-C6	Firmware integrity; service interface
Test station	Production PC; test application	Privileges, updates and operator identification
Network / DMZ	OT VLAN; gateway/API	Traffic rules, authentication and TLS
IT / server	API, MariaDB, logs and backup	Result integrity, retention and recovery

This is a teaching example. An actual configuration requires inventory and security and functional testing.

MODULE I — 75 MINUTES

A new legal framework for engineers

NIS2/KSC, CRA, roles, product scope, risk management and evidence of compliance.

> **Learning outcome.** Participants can distinguish obligations at the organisation level from obligations at the product level and prepare a documented decision on the regulations' applicability.

1. NIS2 and CRA regulate different things

NIS2 is **Directive (EU) 2022/2555**, implemented in Poland primarily through the Act on the National Cybersecurity System (KSC). It establishes a cybersecurity management framework for specified entities, management responsibilities, incident management, supply-chain measures and business continuity. The CRA is **Regulation (EU) 2024/2847** on products with digital elements placed or made available on the EU market in the course of a commercial activity. The CRA links security to the product lifecycle, vulnerability handling and conformity assessment. [R1–R3]

Question	NIS2 / Polish KSC	CRA
What is covered?	An essential or important entity and its systems	A product with digital elements and its manufacturer
Main requirement	ISMS, organisational and technical measures, incident handling	Security by design, vulnerability handling, documentation and conformity
Example evidence	Policy, risk assessment, training and incident records	Technical File, SBOM, test reports and firmware versions
Common mistake	“Every company must implement NIS2”	“Every device with an MCU needs a notified body audit”

> **Key conclusion.** A company may be subject to the KSC and also manufacture products covered by the CRA, but the obligations are not the same. Conversely, an organisation outside the KSC's scope may manufacture products subject to the CRA.

2. The 2026–2027 timeline

Date	Event and its significance
3 April 2026	Polish KSC amendment implementing NIS2 entered into force.
3 October 2026	Deadline for self-registration in the KSC register for entities subject to that requirement.
11 September 2026	CRA reporting duties began for certain actively exploited vulnerabilities and severe product security incidents.
3 April 2027	Main deadline for compliance by covered entities that met the criteria on 3 April 2026.
11 December 2027	Most substantive CRA product requirements begin to apply.

Date	Event and its significance
3 April 2028	First audit deadline for relevant essential entities under the conditions specified in the KSC.

Sources: [R1], [R2], [R4], [R5]. Not all entities follow the same timetable. Specific scope and deadlines must be assessed case by case.

3. When is a company subject to the Polish KSC?

Carry out a self-assessment based on actual activities, sector/subsector, size criteria, exceptions and special provisions. Essential and important entities may fall within the Act through different mechanisms, including ex officio registration. Checking only the Polish PKD activity code or number of employees is insufficient. [R2, R4]

- Identify the service actually delivered and the organisational unit responsible for it.
- Review Annexes 1 and 2 and Article 5 of the KSC, including thresholds and exceptions.
- Record the decision: **in scope / out of scope / requires legal confirmation**.
- For a covered entity, determine registration, ISMS, incident procedures, accountability, audits and reporting obligations.

4. Is our test station a “product with digital elements”?

First establish what the product is: a standalone controller, gateway module, the complete station, or software supplied to a customer. Then determine whether it is made available on the EU market as part of a commercial activity, its intended use, possible exclusions or specific legislation, and who acts as the manufacturer. Evaluate the particular configuration and distribution model. [R3]

Example	Preliminary assessment — not a substitute for legal analysis
Station used only for internal trials; not made available on the market	No typical market-availability event; assess other laws, customer requirements and risks.
Controller with firmware supplied to a customer under the manufacturer's brand	Potentially within CRA scope; verify product characteristics, intended uses and exclusions.
Control software sold or supplied as part of a product	May be covered as a product component or a product in its own right; record dependencies and responsibilities.
Supplier component integrated into a station sold under your own brand	Responsibility for the final product does not disappear; components and configuration must be verified.

> **CE caution.** The CRA complements product conformity requirements. It does **not** mean that every product requires mandatory certification by an external body. The assessment route depends on product classification, standards and CRA conditions. CE marking does not replace technical documentation.

5. Manufacturer, integrator and supplier responsibilities

Assign roles for every product and interface. Among other things, the manufacturer is responsible for product risk assessment, design conformity, user information, the applicable support period, vulnerability management and documentation. An integrator should collect evidence for sourced components and perform integration tests. A component's certificate does not automatically discharge responsibility for the complete system. [R3]

- Include update requirements, vulnerability notification times and an SBOM format in supplier agreements.
- Assign ownership of firmware-signing keys, the release publication process and risk acceptance.
- Record HW, bootloader, firmware, library, test equipment and PC software versions as one release baseline.

6. Risk is an engineering decision

A risk assessment is not merely a list of CVEs. It should account for product characteristics, intended use, physical access, ease of attack, and consequences for confidentiality, integrity, availability, functional safety and the industrial process. In some applications, a false test result can be more damaging than temporary server unavailability.

Example **workshop model (not a statutory method)**: likelihood **P = 1–5**, impact **I = 1–5**, and score **R = P × I**. Priorities: 1–5 low, 6–12 medium, 15–25 high. After applying controls, reassess residual risk and identify the decision owner and evidence of effectiveness.

ID	Threat	P × I	Control	Evidence
R-01	ATmega4809 firmware replacement	4 × 5 = 20	Control the origin of the package and programming process	Release manifest; service report
R-02	Falsified test result from PC	3 × 5 = 15	Authorised API; unique ID and record integrity	Negative test; correlated logs
R-03	Measurement API downtime	4 × 3 = 12	Local queue, retries and deduplication	Network outage and recovery test

Workshop I-1. Classify ST-24

Task. Consider three variants: **A** — internal-only station; **B** — station sold to a customer under the manufacturer's brand; **C** — remote maintenance service for critical infrastructure. For each, document possible KSC and CRA applicability, missing facts and the decision owner.

Output: A three-variant matrix × [KSC / CRA / CE / owner / sources].

Facilitator guidance / answer: Do not automatically tick “YES” solely because an ESP32 is present or a service contract exists. Distinguish the company's status from the product, considering EU market availability, the contractor's role and service type.

MODULE II — 85 MINUTES

Security by Design in hardware and software

From threat models to access controls, secure firmware, segmentation and tests.

> **Learning outcome.** Participants can identify the ST-24 attack surface and select proportionate security controls together with evidence that they work.

1. Security by Design is not the final pre-release test

Security starts with requirements — before PCB layout and coding. During concept design, describe trust boundaries, critical data, interfaces, roles, updates, the support period, suppliers and decommissioning rules. During system design, choose an architecture that limits access. During acceptance, test the actual protections. Controls must be proportionate to risk; a long checklist alone does not prove compliance. [R3]

Stage	Decision and deliverable	ST-24 example
Requirements	Security objective and acceptance criterion	A test result must have a trusted source and ID.
Design	Threat model and trust boundaries	OT VLAN; separated admin and API ports
Implementation	HW/FW/server controls	Signed releases; request authorisation
Verification	Positive and negative testing	Reject invalid signatures and tokens.
Operation	Logging, fixes, accountability	Issue register and support plan

2. Inventory entry points and trust boundaries

Mark ESP32-C6 to server communication, UART/RS485, UPDI service port, station PC, USB, Wi-Fi, VPN and physical access on the architecture diagram. For every boundary, specify identity, authentication, protocol, permitted traffic direction and failure scenarios.

Interface	Example attack	Minimum control
UPDI / service port	Firmware replacement during servicing	Physical access control; supervised procedure; image verification and programming log
ESP32-C6 → API	Device impersonation	TLS with certificate verification, device identity and authorisation
PC → database/API	Changes to tests, data or thresholds	Roles and change approval; API-only access
Web panel / remote service	Technician account takeover	MFA; individual accounts; VPN; grant/revoke access
OT network	Lateral movement from office IT	Firewall between zones, allow-list rules and exception reviews

3. Secure firmware: MCUs with different capabilities

ESP32-C6: Verify the availability and configuration of the platform's security mechanisms, including Secure Boot, flash encryption, debug interface protection and downgrade prevention. These require deliberate key provisioning, a service policy and a secure recovery route. Enabling one option is not proof that the entire update process is secure.

ATmega4809: Standard programming through UPDI is not equivalent to an authenticated update. Where the hardware does not provide native signature verification at boot, protect the process externally: a controlled service tool, signed package manifest, operator identification, post-flash comparison of versions/hashes and protection from unauthorised physical access. High-risk uses may require an architectural change or an additional security component.

> **CRC is not proof of authenticity.** A CRC detects accidental data corruption but does not stop deliberate firmware replacement. Cryptographic mechanisms and key protection are needed to verify origin.

Example firmware release policy

- Source and build configuration are linked to a commit/tag and a separate version number.
- The build pipeline produces a manifest containing device model, HW revision, build ID, SHA-256 hash, release signature, dependency list and test results.
- Service staff retrieve approved packages only; received artefacts are verified before use.
- Each programming event records device ID, operator, time, versions before/after and verification outcome.
- Rollback is controlled and permitted only to compatible, still-supported versions; recovery procedures must not bypass integrity checks.

4. Test network segmentation and access management

In ST-24, separate the devices/controllers zone, production PCs, an intermediate-services DMZ, and IT/administration. OT traffic should be narrowly defined. A VLAN helps organise the network, but without proper routing and firewall rules it is not a complete security boundary.

Communication path	Example policy	Rationale
ESP32-C6 → API	Only HTTPS/TCP 443 to one host	Limits unnecessary inbound/outbound traffic
OT PC → update repository	Only via a controlled gateway/proxy	Enables package validation and logging
IT → OT	Default deny; service via MFA-protected bastion	Auditable, time-limited access
OT → log server	Dedicated authenticated/integrity-protected channel	Supports incident investigation
OT ↔ Internet	Block except required destinations and services	Reduces the attack surface

5. Account and service access management

Every technician has an individual, least-privilege account. The station owner approves privileged access with a defined scope and expiry. Apply MFA where technically possible, especially to VPN, bastions, admin panels and repositories. If a device cannot support interactive MFA, enforce strong authentication at the access gateway and control the session.

- Avoid uncontrolled default or shared service accounts; use unique credentials and rotation.
- Separate privileges for operator / diagnostics / configuration / updates / administrator.
- Log remote sessions and changes to test parameters, calibration and accepted thresholds.
- Provide documented “break-glass” access, with activation logging and post-event review.

6. Choose negative tests to match requirements

A security test should specify initial conditions, steps, expected result, actual result, log and approver. Examples include: an invalid HTTPS certificate, an invalid package signature, replay of a measurement record, loss of network during data writes, unauthorised recipe changes and attempted installation of an older unsupported version.

Test ID	Attempt	Expected response
SEC-01	Substitute an unknown API certificate	Reject connection; do not send data
SEC-02	Change one byte in a signed image	Reject release/update
SEC-03	Operator manually changes a test threshold	Deny, log attempt, leave setting unchanged
SEC-04	Submit the same measurement ID twice	Store once; retain deduplication trace

Workshop II-1. ST-24 threat model

Task. Draw the IT–OT boundaries and mark eight entry points. For the three most important threats, select a control, test and owner.

Output: Architecture diagram and table [entry point / attack / consequence / control / test / owner].

Facilitator guidance / answer: Silent quality-data manipulation may be the highest risk. Do not accept “antivirus on the PC” alone as protection for firmware or results integrity. Pay attention to authority to change recipes and firmware versions.

Workshop II-2. Service operation and unsigned firmware

Task. A technician connects UPDI to an ATmega4809. Design a process that distinguishes successful installation of a known image from accepting an arbitrary HEX file.

Output: A five-step service instruction and audit record.

Facilitator guidance / answer: Establish the package trust anchor, independent verification of the signed manifest, model/revision compatibility, post-flash readback and hash/version checks, and archiving of the programming record. Account for hardware limitations.

MODULE III — 70 MINUTES

Supply chain and multi-supplier component integration

Supply chain risk management (SCRM), supplier verification, SBOM/HBOM, open source and automation of vulnerability management.

> **Learning outcome.** Participants can establish an up-to-date inventory of product components, detect a problem in a dependency, and decide whether to update a component or implement compensating measures.

1. Supplier risk is product risk

Assess not only purchased controllers and radio modules, but also SDKs, libraries, prebuilt firmware, build tools, update services and service subcontractors. The main questions are: who fixes vulnerabilities, how quickly are customers notified, which versions are supported, and can we test updates before deployment?

Area	Question for the supplier	Required evidence
Support	Until when will fixes be released?	Support policy; end-of-support date
Vulnerabilities	How are security advisories published?	PSIRT contact / portal / SLA
Updates	Are update packages signed?	Verification specification and demonstration
Components	Is a version-specific SBOM available?	CycloneDX or SPDX; version and date
Access	Are there default passwords or debug interfaces?	Interface list and factory settings
Changes	Can the BOM/SDK change without notice?	Product change notice (PCN), changelog and approval process

2. SBOM versus HBOM — why the distinction matters

A **Software Bill of Materials (SBOM)** identifies software components and their dependency relationships: for example, TLS libraries, framework versions, server packages and firmware. A **Hardware Bill of Materials (HBOM)** records physical modules and components: an ESP32-C6, ATmega4809, Wi-Fi module, power supply and PCB revisions. HBOM is a useful engineering artefact. The legal requirements and scope of the SBOM should be assessed against the CRA and technical documentation, rather than assuming that a complete list of every component must be publicly disclosed. [R3]

Type	Example item	Why it is needed
SBOM	ESP-IDF/Arduino core with exact versions	Check advisories, EOL and dependencies
SBOM	JSON / HTTP / TLS library	Assess exploit paths and fixes
SBOM	Linux/Windows packages and station application	Monitor operational software

Type	Example item	Why it is needed
HBOM	ESP32-C6 module, manufacturer, part number, revision	SoC/ROM issues and replacements
HBOM	ATmega4809, PCB revision and programmer	Flashing method, compatibility and service
HBOM	Switch, firewall, gateway and server	FW/OS versions and ownership

Minimum SBOM entry — workshop template

Field	Example value
Product / release	ST-24 / rel-1.4.2
Component name	Communication library
Version / identifier	2.7.1 / purl or other stable ID
Supplier / source	Vendor repository, URL and commit
Licence	Open-source licence compliance review
Vulnerability status	CVE / none known / assessment pending
Person and date	Product owner; scan date

3. DevSecOps automation for a small engineering team

You can start without complex infrastructure. At each release, run a pipeline from a clean commit: dependency analysis and SBOM generation, basic code and secret scanning, tests, archiving of results and generation of a signed artefact. Dependency changes require review of both licence compliance and vulnerabilities.

- **Repository:** protect branches, review changes, require unique accounts and MFA, and retain permission logs.
- **CI:** prevent secret exposure, separate build and release privileges, pin tool versions.
- **SCA/SBOM:** generate dependencies for the exact release; do not scan only README files.
- **SAST and testing:** use findings to start triage, not as proof that vulnerabilities do not exist.
- **Release:** retain pipeline outcome, manifest, checksums, signature and IDs of devices included in an update.

4. When a CVE is detected: VEX and triage procedure

A scanner finding that “this library has a CVE” does not necessarily mean the product is exploitable. Check the component version, whether the affected function exists and is reachable, configuration, possible attack path, exposure, impact and available fix. Record the decision and supporting reasoning in a register. The VEX concept can express **affected / not affected / under investigation / fixed**, with justification tied to a release.

Step	Action	Timing / criterion
1. Detection	Map the CVE to product(s) and release(s)	Prompt triage and record
2. Assessment	Confirm attack path and exposure	Product risk, not CVSS alone

Step	Action	Timing / criterion
3. Decision	Patch, mitigation, no impact or exception	Risk owner approval
4. Test	Functional and security regression	Evidence for actual HW revision
5. Distribution	Package, notice, instructions and rollback	Based on priority and support policy
6. Closure	Record product/release and rescan	Confirmed status

> **Important distinction.** Handling a component vulnerability and legally notifying an authority are different activities. CRA reporting applies to actively exploited vulnerabilities and severe incidents affecting the product's security; every CVE finding does not automatically satisfy those legal thresholds. [R5]

5. Managing component obsolescence and end of support

Maintain a compatibility map: PCB revision ↔ module ↔ bootloader version ↔ firmware ↔ PC configuration. For products expected to operate in factories for many years, plan for unavailable components, build-tool and cloud-provider EOL, MCU memory limits affecting updates, and restoration of a station after an incident.

Workshop III-1. CVE in an ESP32 network library

Task. Dependency scanning reports a serious vulnerability in a network library used by ST-24 firmware. The team does not know whether the vulnerable function is called. Prepare a plan from the first alert to closure.

Output: Vulnerability record listing product/release, component, exploit conditions, risk, actions and evidence.

Facilitator guidance / answer: Do not decide from the CVSS number alone. Check execution path and configuration; establish whether the device is online, whether a compensating control exists and who is responsible for notifying users.

Workshop III-2. Build an HBOM and an SBOM

Task. List five hardware and five software items for the station, including revision, version, manufacturer, support status and owner.

Output: Two inventories linked to one ST-24 release.

Facilitator guidance / answer: Clear mapping between parts and each product variant/unit is critical. If release contents change, updating the SBOM only once per year is insufficient.

MODULE IV — 70 MINUTES

Detection, incidents and business continuity

Telemetry, response procedures, technical logs, reporting obligations, BCP and recovery exercises.

> **Learning outcome.** Participants can design minimum viable monitoring, practise incident handling and containment, and safely restore production.

1. Event, vulnerability and incident are different concepts

An **event** is an observable change in state (for example, a failed login). A **vulnerability** is a weakness that could be exploited. An **incident** is an event capable of compromising system security or operation. Exact legal definitions and reporting categories vary by legislation. Users need a simple route to report unusual symptoms before the formal legal classification has been determined.

Symptom	Initial response	Possible consequence
Ten failed service logins	Alert, verify source, block/rate limit	Attempted account compromise
New FW version with no approved manifest	Stop station; preserve evidence	Firmware and result manipulation
Database unavailable	Buffer records and monitor queue	Data loss or delay
PC results do not match device records	Preserve records; investigate	Quality and integrity risk
Unusually high OT → Internet traffic	Assess process impact; isolate logically	Exfiltration / lateral movement

2. What should a log record contain?

Each important record should contain a UTC timestamp or unambiguous time zone, device/station identifier, event correlation ID, software version, operation type, result, source and severity. For authorisation events also log account/role and session ID. Synchronise clocks and control retention; do not store secrets or personal data without a purpose and legal basis.

Field	Example value	Why
`timestamp_utc`	`2026-11-20T09:41:21Z`	Ordering and correlation
`station_id`	`ST-24-07`	Device identification
`event_id`	`e7f4...`	Deduplication and tracking
`fw_version`	`esp32-1.4.2 / avr-1.1.8`	Product variant
`event_type`	`measurement_upload`	Detection rules
`result / status`	`rejected / invalid_signature`	Rejection reason
`actor / source`	`service-operator-02 / vpn`	Action auditing

3. Minimum monitoring architecture for ST-24

Devices produce only necessary telemetry. The station computer forwards logs to a central collector, while the gateway retains firewall/VPN logs. The server gathers API logs, authentication failures, release history and availability metrics. Monitor traffic volume, unexpected destinations, MCU restarts, failed updates and interruptions in measurement transfer. An alert must reach a named person and have a handling procedure.

> **Example: results in `access.log`.** If measurements are sent through HTTP GET requests and appear in `access.log`, consider authenticated HTTPS POST with server-side API validation, an idempotency identifier and local buffering. A web server log should not be the sole source of production data; do not expose tokens or sensitive data in URLs. This is a design recommendation, not a verbatim CRA provision.

Simple alerting rules — examples

- **Firmware changed:** Device-reported version or programmer record differs from approved baseline.
- **Multiple authentication failures:** Increase in failed logins from an account or IP address.
- **Missing heartbeat:** Critical station does not communicate within a defined interval.
- **Data integrity:** Impossible sequence, unknown measurement ID, mismatched calibration or duplicate record.
- **Unexpected egress:** OT network connects to an unregistered Internet destination.

4. Incident response procedure — operational model

Phase	Actions	Artefact
1. Report	Record detection time, source and impact	Incident record
2. Triage	Scope, severity, possible safety/quality impact	Classification and priority
3. Contain	Disconnect session, block account/traffic, stop release	Decision and action log
4. Analyse	Preserve logs/images; determine versions and timeline	Timeline and hypothesis
5. Remediate	Reset credentials, replace FW, adjust rules, patch	Change record and test evidence
6. Recover	Resume after process owner approval and control tests	Recovery protocol
7. Learn	Root cause, new controls, accountable owner	Report and review deadline

5. RACI roles and the decision to stop the line

Activity	OT / FW engineer	IT / SOC	Production manager	Management / compliance
Technical triage	R	R	C	I
Network/account isolation	C	R	A	I
Stop/resume testing	R	C	A	I

Activity	OT / FW engineer	IT / SOC	Production manager	Management / compliance
Regulatory reporting classification	C	R	C	A
Customer contact	C	C	R	A
Post-incident review	R	R	C	A

Legend: R = Responsible (performs), A = Accountable (approves), C = Consulted, I = Informed. This allocation is illustrative and must be adapted to the organisation.

6. NIS2 and CRA: incident reporting frameworks

For an entity covered by the Polish KSC, obligations, recipients and procedures arise from Polish national law and implementing provisions. NIS2 envisages, among other stages, an early warning within **24 hours**, an incident notification within **72 hours** and generally a final report within **one month**. In Poland, operational decisions must be based on the current KSC and the applicable definition of the incident. [R1, R2]

Under the CRA, from **11 September 2026** manufacturers must report specified actively exploited vulnerabilities and severe incidents affecting product security. The early warning is due within **24 hours of becoming aware**, and the notification within **72 hours**. The final report for an actively exploited vulnerability is generally due within **14 days after a corrective measure becomes available**; for a severe incident, within **one month after the 72-hour notification**. The legislation and reporting platform define qualifying events and procedures. [R3, R5]

> **Not every server error is reportable.** Database downtime is not automatically a CRA-reportable incident. Conversely, active exploitation of a product vulnerability may trigger a reporting duty while production is still running. Appoint the person who decides on classification and establish a documented procedure.

7. Business continuity and disaster recovery (BCP/DRP)

A **Business Continuity Plan (BCP)** describes how critical functions are maintained during disruption; a **Disaster Recovery Plan (DRP)** describes restoration of resources. For test stations, distinguish safe shutdown from continued production. Unverified results must not be accepted merely to meet productivity KPIs.

Element	ST-24 example
RTO — recovery time objective	4 hours for server, based on impact assessment (illustrative)
RPO — recovery point objective	Target: zero lost approved results, using buffering/deduplication
Local buffering	Durable queue with capacity limit and backlog counter
Recovery	Versioned PC image, configuration restore and post-deployment tests
Recovery drill	Disconnect API for 30 minutes, restore and reconcile counts
Communication	Define who stops tests and when quality/customers are informed

Workshop IV-1. Incident: unknown firmware version

Task. At 09:15 an operator sees an unfamiliar ST-24 firmware version. Two measurement records differ from copies on the device. The team suspects service-account misuse. Run a 15-minute simulation and distinguish technical, production and regulatory decisions.

Output: Incident record, first-hour timeline, RACI, evidence list, decision on stopping production and notification.

Facilitator guidance / answer: Prioritise human safety and product trustworthiness. Do not erase logs or overwrite disks without an evidence-preservation plan. Review VPN access and flashing history; compare the approved image and independent test results.

Workshop IV-2. MariaDB failure while a station is operating

Task. Remove database access. Decide what happens to records, how backlog is reported and how transmission resumes without duplicate measurements.

Output: Recovery test sequence and RTO/RPO indicators.

Facilitator guidance / answer: Durable queue, unique measurement IDs, write acknowledgement, retry/backoff, storage limits, fill-level alert and agreed conditions for stopping production.

MODULE V — 60 MINUTES

CE marking, technical verification and audit

From requirements mapping to the Technical File, resilience tests, declarations of conformity and product oversight.

> **Learning outcome.** Participants can prepare a conformity-assessment plan and product evidence set, and distinguish penetration testing from a formal conformity assessment procedure.

1. How does the CRA relate to CE marking?

For a product within the CRA's scope, the manufacturer determines the product classification, identifies applicable essential requirements, prepares a risk assessment, implements controls, carries out the required conformity assessment and technical documentation, then draws up the relevant EU declaration of conformity and applies the CE mark. The CRA coexists with other EU legislation requiring CE marking, depending on the product and its functions, but the scopes and tests are not identical. [R3]

Possible CRA routes include **internal production control (Module A)** and, in circumstances specified by law, procedures involving a notified body. The correct route depends on the classification and CRA conditions. The phrase “CE certification” is often shorthand: the CE mark is the result of a conformity assessment, not a single certificate issued to every product by an outside authority.

Step	Review question	Evidence
1. Scope	Is this a product within the CRA? Any exemptions?	Documented classification decision
2. Requirements	Which Annex I requirements apply?	Requirements and deviation matrix
3. Risk	Which threat scenarios were identified?	Risk assessment and decisions
4. Design/tests	Do the measures actually work?	Verification plan and report
5. Assessment route	Is Module A permitted?	Rationale and procedure outcome
6. Documents	Are the Technical File and instructions complete?	Controlled document set
7. Release	Are the EU declaration and marking correct?	Declaration, release and product ID
8. Oversight	How are vulnerabilities and updates handled?	PSIRT, support policy and event log

2. Technical File under the CRA — a practical structure

Article 31 and Annex VII of the CRA specify technical documentation content as applicable to the product. The following is a workshop-friendly engineering structure; it is **not** a verbatim transcription of the Annex. [R3]

Section	Minimum proposed content for ST-24
00. Identification	Model, manufacturer, intended purpose, HW/FW/SW versions, pictures/drawings and variants
01. Architecture	Block diagrams, PCB, firmware, connections, trust boundaries and interfaces

Section	Minimum proposed content for ST-24
02. Requirements	Annex I mapping, other applicable regulations, scope and exemptions
03. Risk assessment	Threat scenarios, criteria, decisions and residual risk
04. Controls	Access, updates, identity, logs, physical protection and compensating measures
05. Components	SBOM, suppliers, versions, HBOM, EOL and vulnerability monitoring
06. Tests	Test plan, results, defects, regression, evidence and tested versions
07. Vulnerability process	Reporting channel, triage, fix publication and support policy
08. Instructions	User information, secure configuration and lifecycle guidance
09. Conformity	Assessment route, applied standards, EU declaration and change register

3. Traceability: requirement → implementation → test → evidence

An auditor or conformity reviewer should be able to move from a requirement to a specific implementation and test record. Replace statements such as “we use best practices” with IDs, dates, version references, outcomes and artefact locations.

ID	Requirement / objective	Implementation	Test and evidence
SEC-R01	Only approved firmware releases	Signed manifest; controlled servicing	SEC-02; release report R-14
SEC-R02	Protect transmitted data	TLS + device authorisation	SEC-01; connection log and config
SEC-R03	Only authorised users change thresholds	PC roles; configuration change log	SEC-03; access report
SEC-R04	Secure data recovery	Buffer and deduplication	SEC-04; failover test report

4. What should resilience testing actually cover?

Select test methods based on the threat model. Examples include static application security testing (SAST), software composition analysis (SCA), API protocol tests, authentication tests, UART/HTTP parser fuzzing, attempts to modify firmware, link failure testing, review of factory settings, and inventory of network ports and accounts. For OT, run potentially disruptive tests on a controlled testbed, without endangering physical safety or production continuity.

Penetration test versus Red Team

A **penetration test** assesses a defined scope of systems and vulnerabilities within an agreed testing window. A **Red Team exercise** generally tests an organisation's ability to detect and contain a realistic attack scenario spanning people, processes and technology. Neither automatically replaces a formal product conformity assessment. Link findings back to requirements and the risk assessment.

5. Security testing report — template

Report section	Contents
Objective and scope	What was tested; exclusions and why; environment/testbed
Identification	Product, HW revision, FW/SW versions, model, build hash and date
Method	Acceptance criteria, reference standard and tools
Results	PASS/FAIL tests, screenshots/logs/PCAP, risk and reproduction
Actions	Fix or compensating measure, owner and deadline
Retest	Date, configuration, outcome and approval
Limitations	What remains untested and residual risk

6. What is reviewed after the product is placed on the market?

The CRA takes a lifecycle approach: processing security reports, monitoring known vulnerabilities, providing appropriate updates and informing users. The support period must be determined based on expected use and CRA rules; as a general rule it is at least **five years**, unless the product is expected to be used for a shorter period. [R3]

- Define and communicate the support period; assign responsibility for vulnerability monitoring.
- Maintain product version records and deliver patches to the installed device population.
- Reassess the effect of hardware, architectural or intended-use changes on documentation and conformity.
- Do not end oversight when CE marking is applied; maintain vulnerability and incident handling for as long as required.

7. Consequences of non-compliance and audit readiness

The regulations provide for administrative market-surveillance measures and penalties. The legal basis and severity of a penalty must be assessed for the specific infringement and competent authority; the statutory maximum should not automatically be treated as the most likely outcome. Other practical risks include delayed delivery, withdrawal, rework costs, lost contracts and customer claims.

> **An audit attempts to reconstruct a decision.** Good documentation allows an independent engineer to establish exactly which variant was tested, why a security control was selected, which test proves it works, and what risk remained when the product was released.

Workshop V-1. Technical File completeness review

Task. The group receives ST-24 release 1.4.2, a penetration testing report, a list of libraries without versions, an electrical schematic and no programming history. Identify gaps, prioritise them and design a release gate.

Output: A table [artefact / status / risk of omission / owner / due date / decision].

Facilitator guidance / answer: Distinguish missing documentation from a missing actual control. Complete release-specific SBOM versions, risk assessment and tests. A penetration test does not automatically address all of Annex I.

ANNEX A — WORKING TEMPLATES AND CHECKLISTS

Templates for printing and use by the project team or during the workshop.

> **Learning outcome.** Participants can turn training insights into actual procedures, registers and engineering deliverables.

A1. KSC / CRA applicability checklist

Question	Response / evidence
Does the company provide services in a sector listed in the KSC annexes?	YES / NO / TO ASSESS; business scope
Do KSC size criteria and exemptions apply?	YES / NO / TO ASSESS; legal decision
Is KSC registration and use of S46 required?	Owner, deadline and proof
Is the product hardware/software with digital elements?	Functions and product boundary
Is it made available on the EU market as part of a commercial activity?	Sales model, offer/contract
Are exclusions or special product acts applicable?	Legal basis and rationale
Who is the CRA manufacturer?	Brand owner, integrator, importer
What is the CRA classification and conformity route?	Decision, source and approver

Decision: Scope _____ / person _____ / date _____ / supporting document _____

A2. Risk register — blank form

ID / asset	Attack and effect	P × I	Control / evidence	Residual risk
R-___ / ___	_____	___	_____	___
R-___ / ___	_____	___	_____	___
R-___ / ___	_____	___	_____	___
R-___ / ___	_____	___	_____	___

For every row also record the owner, review date, action plan, acceptance decision and related test. The P × I scale is only a workshop tool.

A3. Product and firmware release checklist

- ☐ The product has a unique identifier, version and hardware revision.
- ☐ Source, dependencies and build process are pinned to an approved release.
- ☐ An up-to-date risk assessment covers functions, configuration and environment.

- ☐ Service interfaces, default accounts and network ports have been reviewed.
- ☐ Secrets and credentials are not stored in plaintext in the source repository.
- ☐ Update packages have a controlled origin and are verified before use.
- ☐ MCU-specific limitations of built-in protection have been assessed.
- ☐ Regression, authorisation, network-failure and malformed-data tests were performed.
- ☐ Logs, test reports and SBOM are linked to the exact release.
- ☐ A secure rollback/device recovery process is available.
- ☐ Service instructions, the support scope and security contact were approved.

A4. Internal incident notification template

Field	To complete
Identifier and reporter	IR-____; account / telephone
Timeline	Detected: ____; response started: ____
Impact	Systems / devices / production / human safety / data
Symptoms and evidence	Addresses, event IDs, versions, logs, PCAP, hostname
Immediate decisions	Block account/segment; stop production; back up evidence
Owner and communication	Technical: ____; legal/compliance: ____; customer: ____
Reportability assessment	NIS2/KSC: ____; CRA: ____; other: ____; rationale
Closure	Retest, permission to resume, root cause, preventive actions

A5. Technical File checklist — working version

- ☐ Product description, intended purpose, manufacturer, models and release identification.
- ☐ Drawings, photographs, system diagram and interface description.
- ☐ Software versions affecting conformity and component inventory.
- ☐ Cybersecurity risk assessment and reasons for applying requirements.
- ☐ Mapping of CRA essential requirements (Annex I) to implementations.
- ☐ Description of design, production, updates and vulnerability handling.
- ☐ Test/verification results and applied standards or specifications.
- ☐ User instructions and secure configuration information.
- ☐ Conformity assessment procedure documentation and relevant EU declaration.
- ☐ Change history and documentation maintenance responsibilities.

This checklist supports documentation preparation. Actual requirements must be checked against **CRA Article 31 and Annex VII**. Not every item will take the same form for every product.

A6. Minimum implementation plan — first 90 days

Period	Task	Owner / output
Days 1–15	KSC and CRA classification; owners; inventory of systems/products	Management + compliance + engineers; decisions and register
Days 16–30	ST-24 threat model, risk register, monitoring scope, network map	OT/IT lead; risk assessment and diagram
Days 31–45	Release repository, manifest, SBOM/HBOM, dependency scan	Embedded + DevOps; CI artefacts
Days 46–60	Access and update procedures, logging, alerts, incident drills	IT/OT + production; test report
Days 61–75	Technical File, traceability, negative tests, support plan	Product owner; version-controlled documentation
Days 76–90	Gap review, budget plan, risk acceptance and mock audit	Management; remediation plan and ownership

A7. Knowledge check — 12 questions

1. Does the mere presence of a microcontroller automatically bring a product within the CRA?
2. Why must KSC and CRA applicability be evaluated separately?
3. Which three pieces of evidence can link firmware to a particular physical device?
4. Why can a CRC not protect against deliberate HEX file substitution?
5. Is a VLAN without ACL/firewall rules sufficient to separate a critical zone?
6. Beyond CVSS, what must be checked when a library vulnerability is found?
7. What is an SBOM used for, and what is an HBOM used for?
8. What conditions trigger CRA reporting after 11 September 2026?
9. How much time does the CRA allow for early warning after becoming aware of a reportable issue?
10. What is the difference between RTO and RPO?
11. Does a penetration test replace the Technical File and CRA conformity assessment?
12. What linkage should exist between a requirement and its test?

A8. Short answer key for the facilitator

1. **No.** Product definition, market availability, exemptions and the manufacturer's role are among the relevant factors.
2. KSC covers certain organisations and services; CRA covers certain products with digital elements.
3. For example: release manifest, device ID and programming/verification record.
4. A CRC is not cryptographic evidence of authenticity or integrity against an active attacker.
5. **No.** Controlled traffic paths, firewall rules and monitoring are required.
6. Component version, exposure, reachability, configuration, product impact and mitigations.

7. SBOM maps software components; HBOM maps physical components and variants.
8. The event must meet the criteria for an actively exploited vulnerability or severe incident affecting product security.
9. Generally within **24 hours of becoming aware**, without undue delay.
10. RTO is the target recovery time; RPO is the acceptable point of data loss.
11. **No.** A penetration test is one possible verification artefact, not the conformity assessment process.
12. A traceability matrix linking requirement ID, implementation, test method/result, version and evidence artefact.

BIBLIOGRAPHY AND LEGAL REFERENCES

Sources and reference documents

Sources were checked as of **20 November 2026**. For later delivery of this training, verify updated legislation, standards and official interpretations again.

[R1] Directive (EU) 2022/2555 — NIS2.
https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32022L2555

[R2] Polish Ministry of Digital Affairs — KSC amendment: obligations and deadlines.
https://www.gov.pl/web/cyfryzacja/nowelizacja-ustawy-o-krajowym-systemie-cyberbezpieczenstwa-ksc---obowiazki-podmiotow-kluczowych-i-waznych

[R3] Regulation (EU) 2024/2847 — Cyber Resilience Act, especially Articles 13, 14, 31–32 and Annexes I, II, VII and VIII.
https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32024R2847

[R4] Polish Ministry of Digital Affairs — key KSC dates.
https://www.gov.pl/web/cyfryzacja/nowelizacja-ustawy-o-krajowym-systemie-cyberbezpieczenstwa-ksc--najwazniejsze-terminy

[R5] European Commission — CRA reporting obligations from 11 September 2026.
https://digital-strategy.ec.europa.eu/en/policies/cra-reporting

[R6] ENISA — Single Reporting Platform, FAQ and notification deadlines.
https://www.enisa.europa.eu/topics/product-security/vulnerability-services/eu-incident-response-and-cyber-crisis-management/single-reporting-platform-srp/frequently-asked-questions

[R7] European Commission — CRA overview and conformity assessment modules.
https://digital-strategy.ec.europa.eu/en/policies/cra-summary

[R8] User-provided training outline: NIS2/CRA workshop programme, supplied on 20 November 2026; five modules and proposed agenda.

> **What is legally required and what is good practice?** NIS2/KSC and CRA set obligations and framework requirements. The phrases “12 areas / 40 requirements”, P × I scales, RACI, a particular VLAN topology, selected SBOM/HBOM fields, KPIs and the 90-day timetable are presented as training tools or recommended methods, **not as statutory checklists**.

END OF TRAINING MATERIALS · EDITION OF 20 NOVEMBER 2026