APH10

NORDIC SOFTWARE SECURITY SUMMIT
NSSS.SE
THE SOFTWARE SECURITY REGULATION REVOLUTION

Secure by Design

November 2026

Anthony Harrison - anthony@aph10.com

© 2026 APH10 Limited

Who am I?

40 years delivering mission critical solutions across multiple sectors

Founder and Director APH10

Co-founder SBOMEurope

Open Source Software

Mentor





CoderDojo

© 2026 APH10 Limited

You can’t manage risks if you don’t understand what you are managing

APH10

© 2026 APH10 Limited

The CRA is about risk management

© 2026 APH10 Limited

The CRA is about risk management

Address **cyber risk** in digital products to **protect users of the product**

© 2026 APH10 Limited

The CRA is about risk management



Threat Modelling

© 2026 APH10 Limited

The CRA is about risk management



Threat Modelling



Security Controls

© 2026 APH10 Limited

Product Usage Risk Management

Foundation of Secure By Design

- Security requirements and architecture decisions
- Prioritisation of controls and assurance activities
- Acceptable trade-offs inherent to the intended use and deployment context
- Ongoing operational posture

Iterative process revisited at key stages of product lifecycle

© 2026 APH10 Limited

Secure by Design

- Secure by Design represents a fundamental shift in product security, embedding protective measures from conception rather than retrofitting them post-development.

- Secure by Default ensures products ship with the most secure configuration reasonably possible, reducing reliance on user expertise and limiting the potential for misconfiguration.

© 2026 APH10 Limited

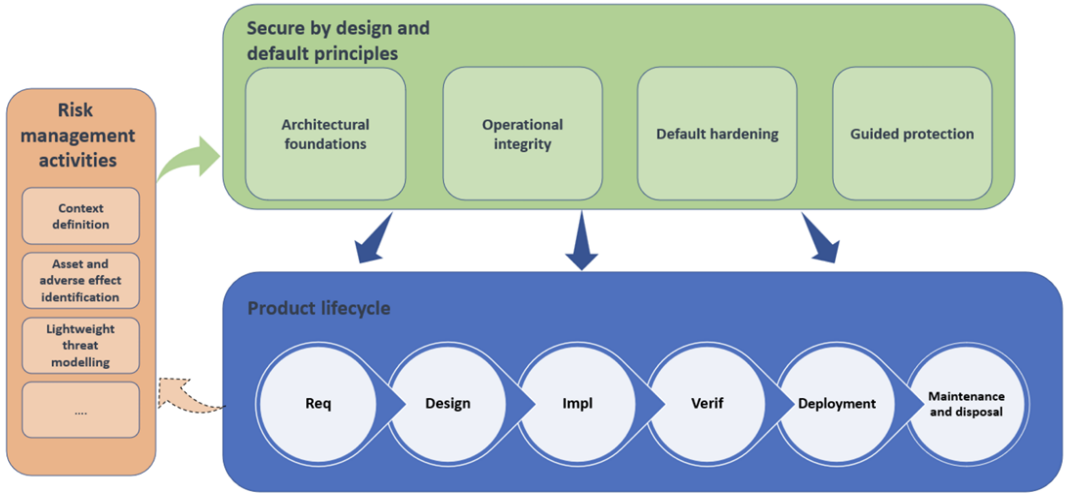
Secure by Design

- Secure by Design represents a fundamental shift in product security, embedding protective measures **from conception** rather than retrofitting them post-development.

- Secure by Default ensures products ship with the most secure configuration reasonably possible, reducing reliance on user expertise and limiting the potential for misconfiguration.

© 2026 APH10 Limited

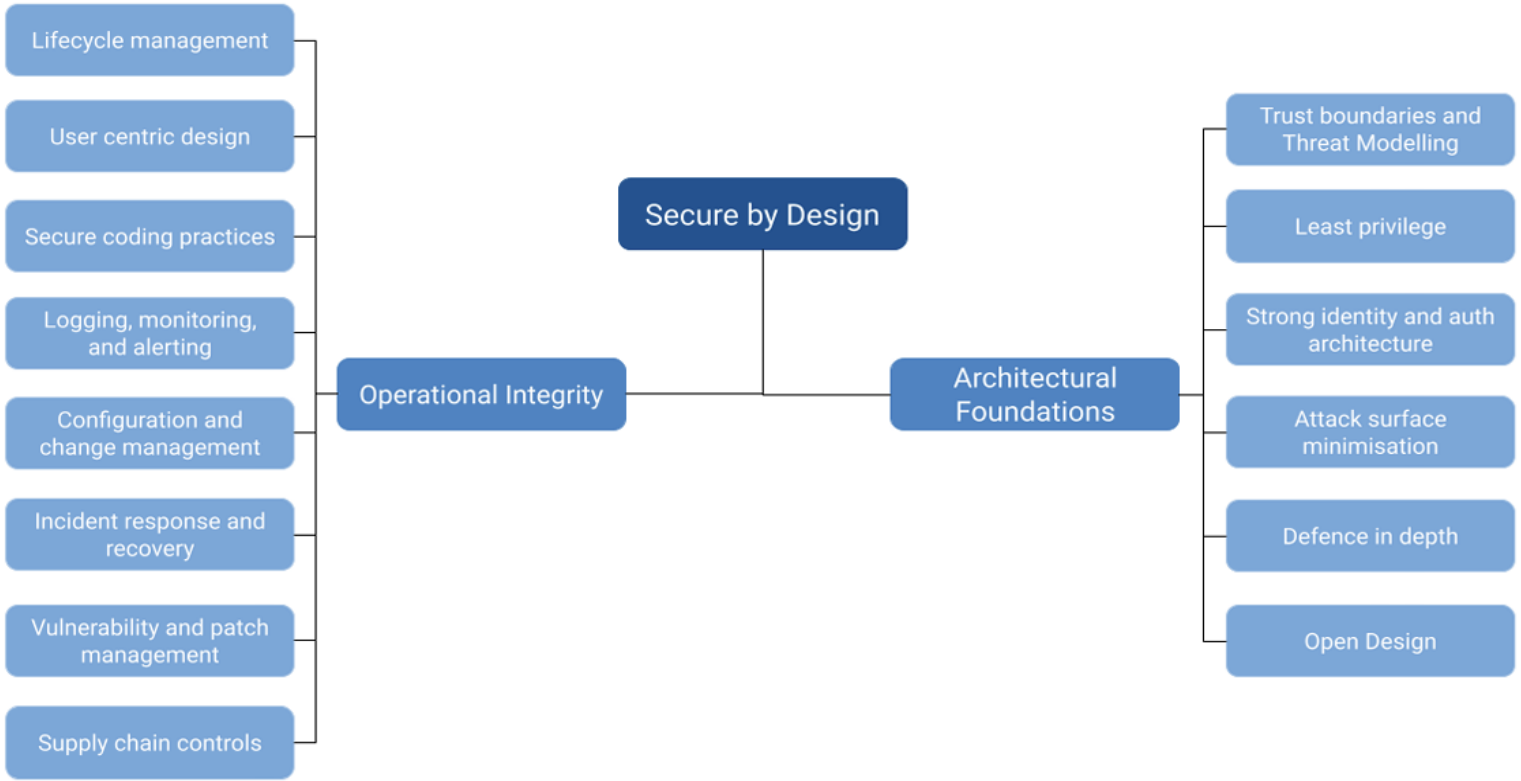




https://www.enisa.europa.eu/publications/enisa-technical-advisory-for-software-bill-of-materials

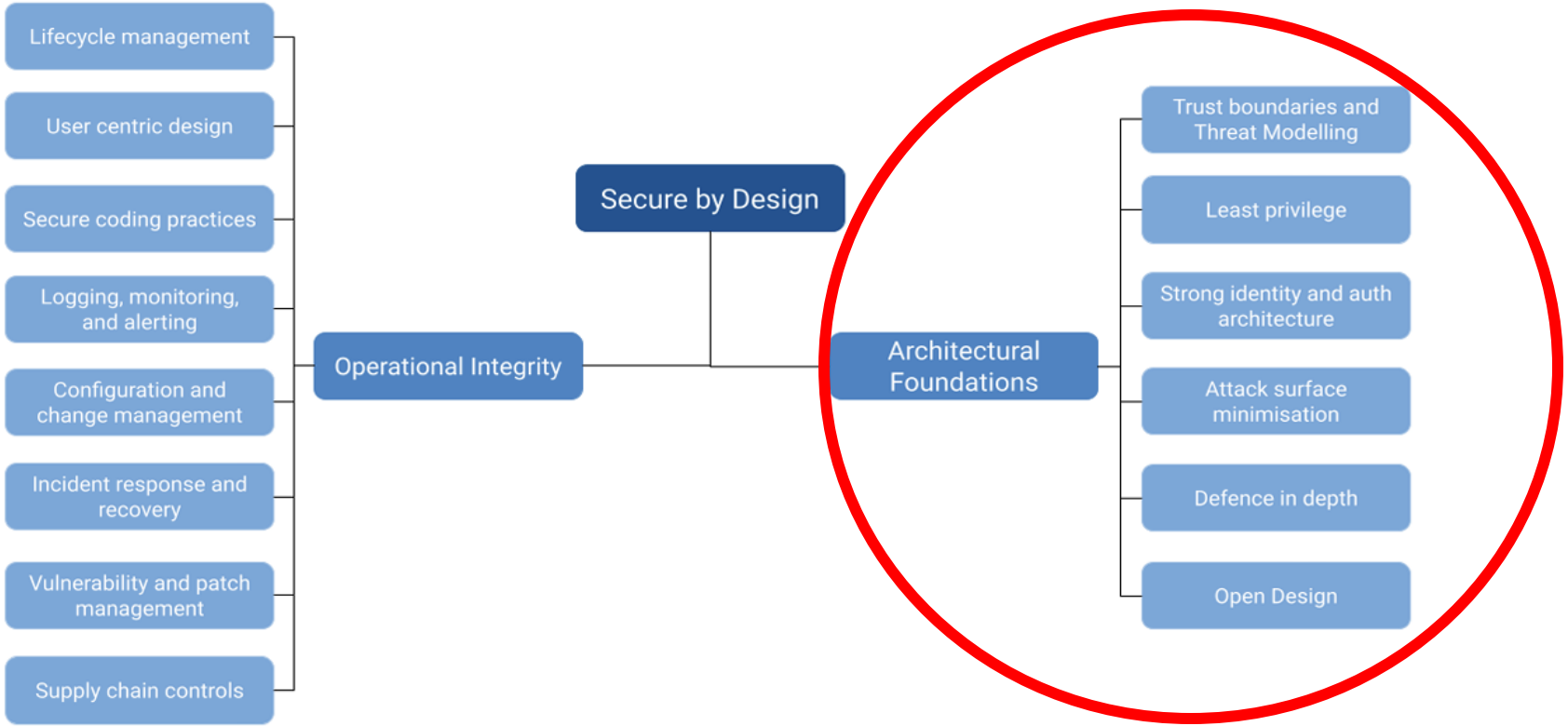
© 2026 APH10 Limited

ENISA Secure by Design Principles



© 2026 APH10 Limited

ENISA Secure by Design Principles



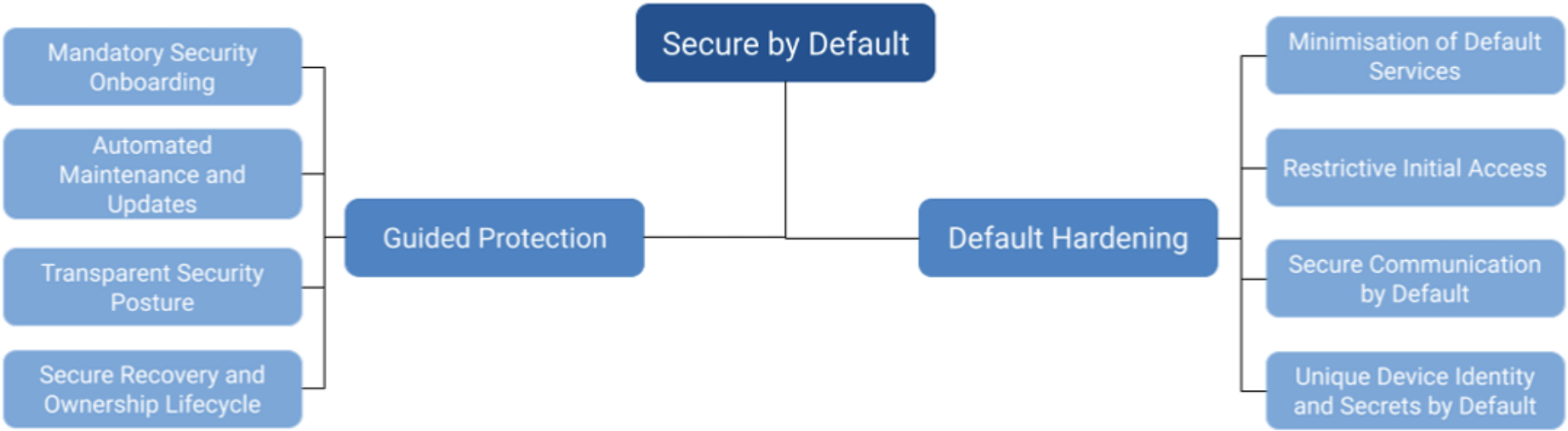
© 2026 APH10 Limited

Understanding ENISA Secure by Design (SbD)

- Secure Defaults
  - Is the component configured to be secure out-of-the-box?
- Least Privilege
  - Does the component request only the permissions it strictly needs?
- Defence in Depth
  - Does it have redundant security controls?
- Visibility and Transparency
  - Is the component's internal logic and data flow auditable?
- Resilience
  - How does the component handle unexpected inputs or failures?

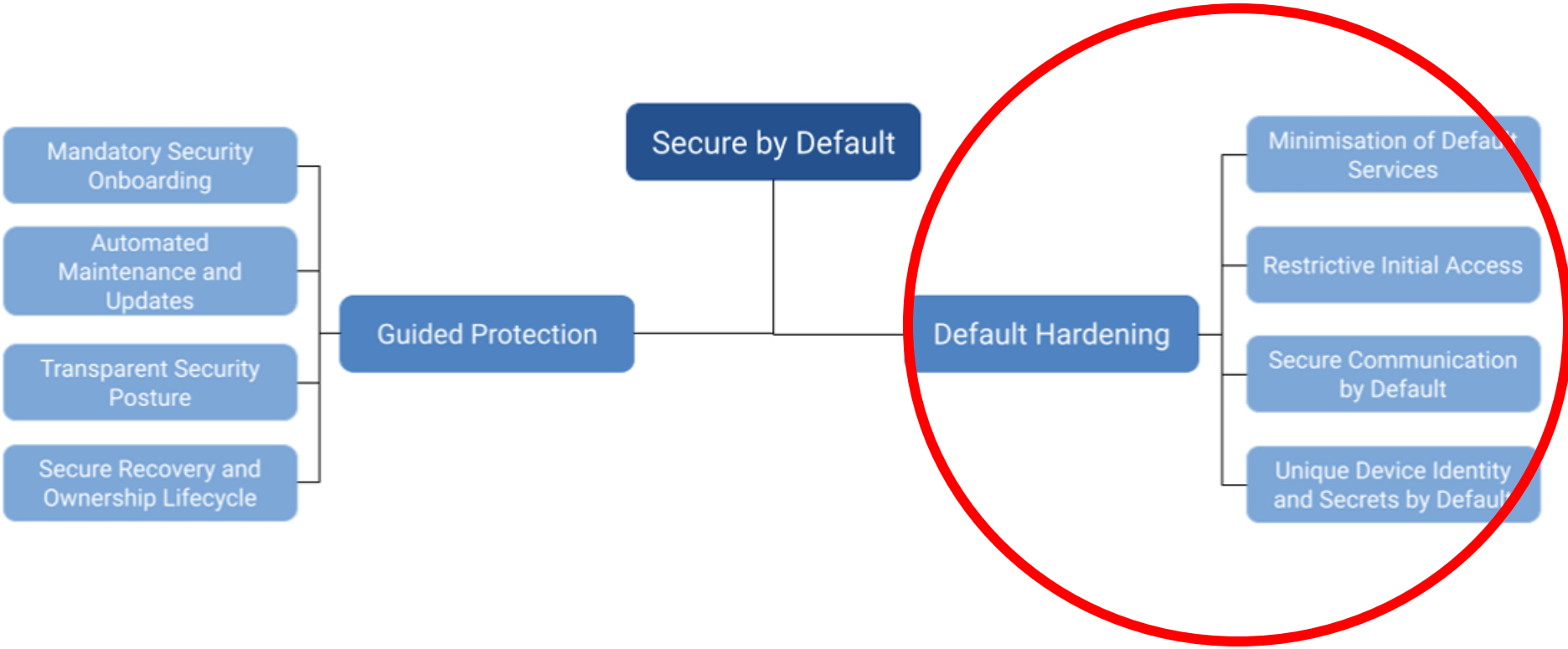
© 2026 APH10 Limited

ENISA Secure Defaults



© 2026 APH10 Limited

ENISA Secure Defaults



© 2026 APH10 Limited

Mapping ENISA controls to NIST 800-53

SbD Principle	Core Objective	Primary NIST 800-53 Families
Secure Defaults	Ensure out-of-the-box security	AC (Access Control), CM (Configuration Management)
Least Privilege	Minimise permissions	AC (Access Control)
Defence in Depth	Redundant security layers	SC (System and Communications Protection) SI (System and Info Integrity)
Identity and Authentication	Enforce roles and unique identities	IA (Identification and Authentication)
Attack Surface Minimisation	Only essential capabilities	CM (Configuration Management SA (System and Services Acquisition)

© 2026 APH10 Limited

The Challenge of Software Supply Chain Security

- Knowing a component exists (inventory) is no longer enough.
  - Need to know the security posture of that component.

© 2026 APH10 Limited

The Challenge of Software Supply Chain Security

- Knowing a component exists (inventory) is no longer enough.
  - Need to know the security posture of that component.

- How do we communicate that a component adheres to ENISA's Secure by Design (SbD) principles?

© 2026 APH10 Limited

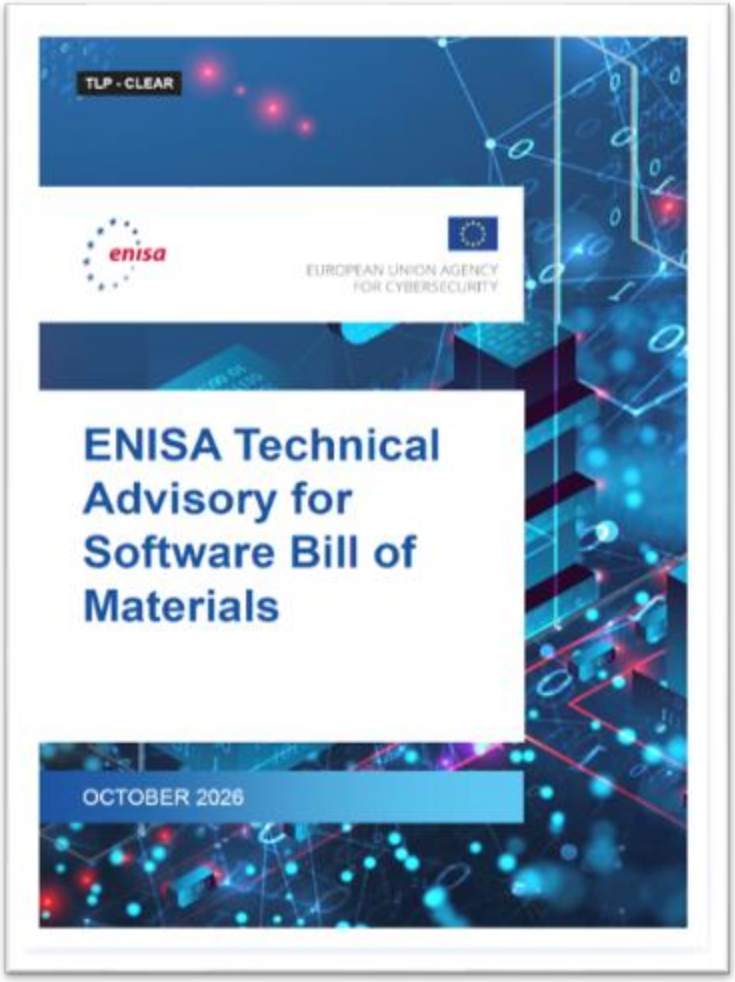
The Challenge of Software Supply Chain Security

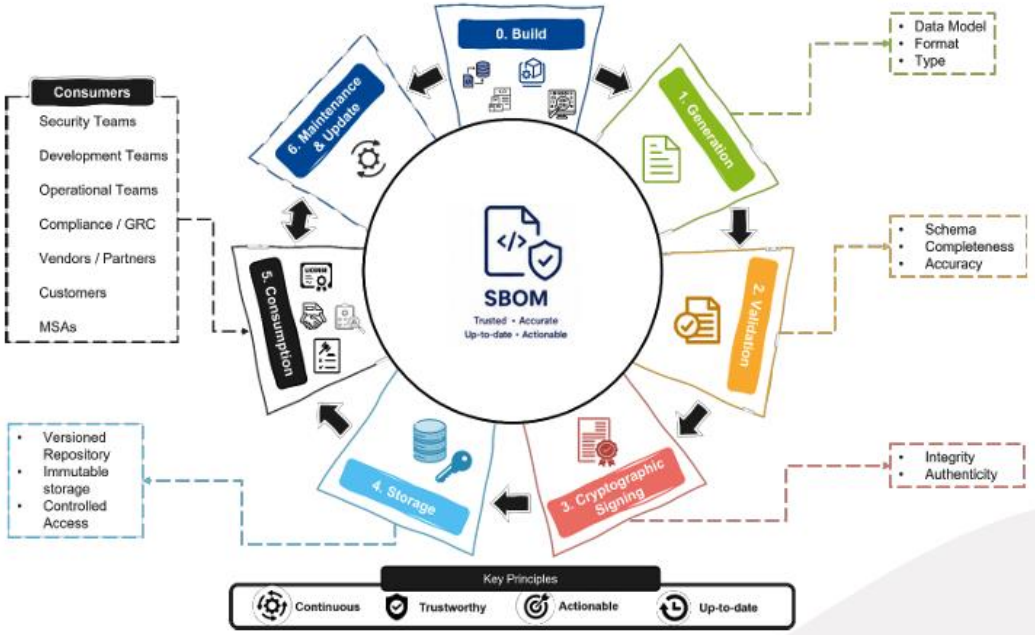
- Knowing a component exists (inventory) is no longer enough.
  - Need to know the security posture of that component.

- How do we communicate that a component adheres to ENISA's Secure by Design (SbD) principles?

Moving a SBOM from "Component Inventory" to "Security Attribute Inventory"

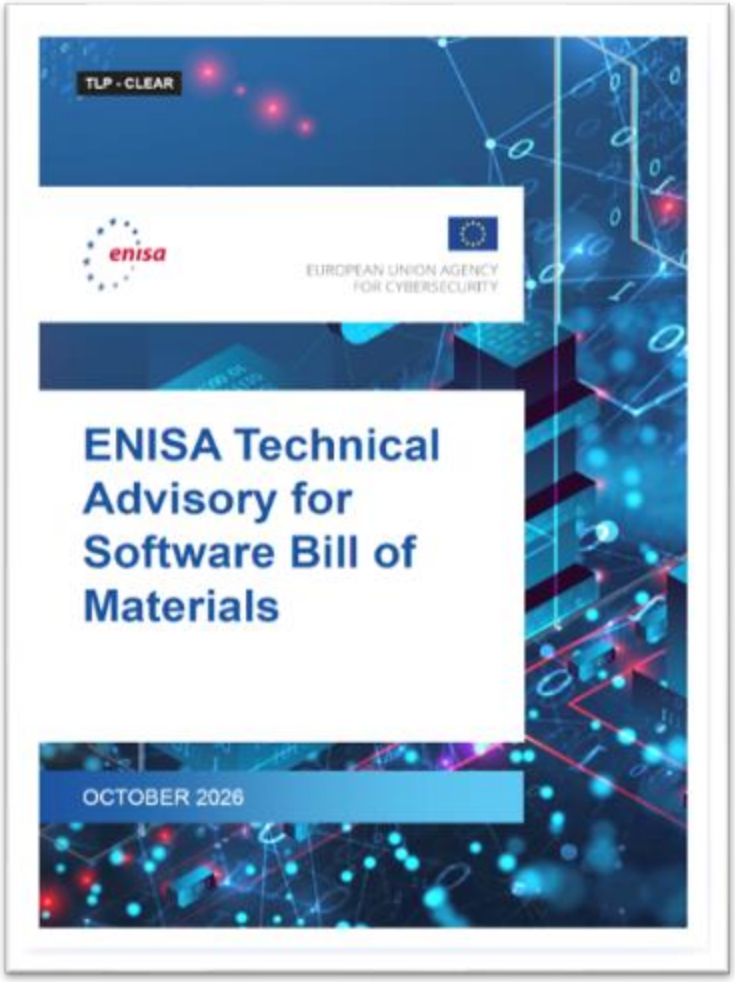
© 2026 APH10 Limited

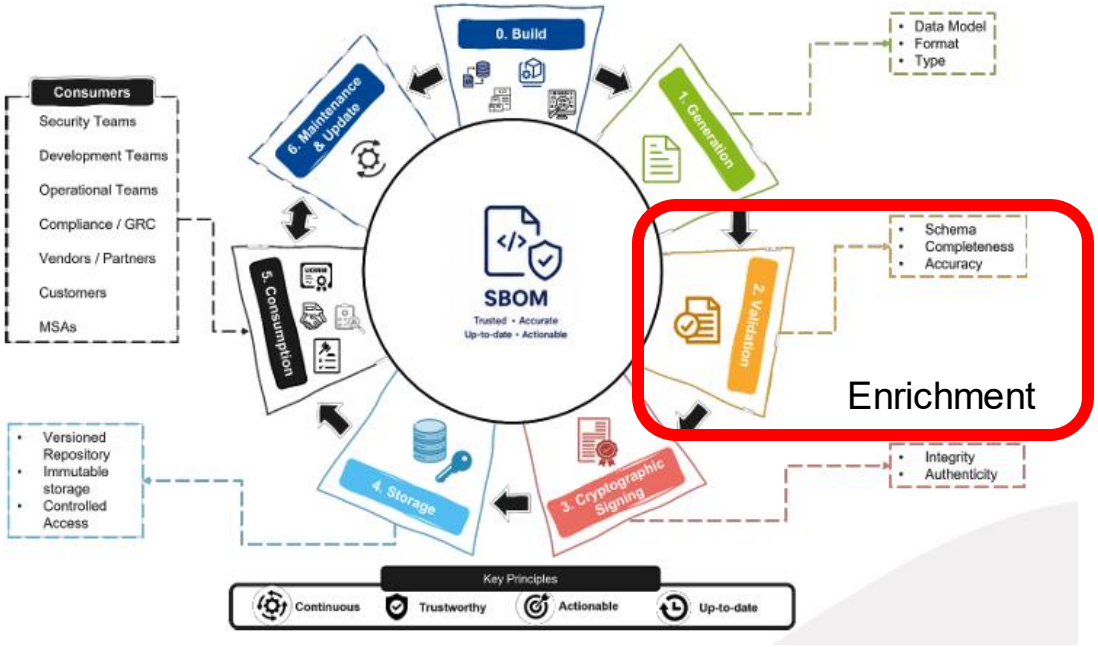




https://www.enisa.europa.eu/publications/enisa-secure-by-design-and-default-playbook

© 2026 APH10 Limited





https://www.enisa.europa.eu/publications/enisa-secure-by-design-and-default-playbook

© 2026 APH10 Limited

The Proposed Approach: "Property-Based Attribution"

- Use user defined properties field to map specific ENISA principles to specific components.
  - Available in CycloneDX (all versions) and SPDX (from version 3.0)

- The Strategy
  - Identify the Principle: (e.g. Secure Defaults).
  - Define the Metric: (e.g. "Is MFA enforced by default?").
  - Assign the Property: Attach a boolean or descriptive value to the component(s) in the SBOM

© 2026 APH10 Limited

The Proposed Approach: "Property-Based Attribution"

- Example Property Mapping:
  - enisa-sbd:secure-defaults: true
  - enisa-sbd:least-privilege: restricted-access-only
  - enisa-sbd:defence-in-depth: enabled
- Create ENISA-sbd taxonomy based on ENISA SbD Playbook

© 2026 APH10 Limited

The Proposed Approach: "Property-Based Attribution"

```
{
"type": "component",
"name": "Auth-Library-v2.1",
"version": "2.1.0",
"description": "Handles user authentication and session
management.",
"properties": [
{ "name": "enisa-sbd:compliance", "value": "verified" },
{ "name": "enisa-sbd:secure-defaults", "value": "true" },
{ "name": "enisa-sbd:least-privilege", "value": "enforced" }
]
}
```

APH10

© 2026 APH10 Limited

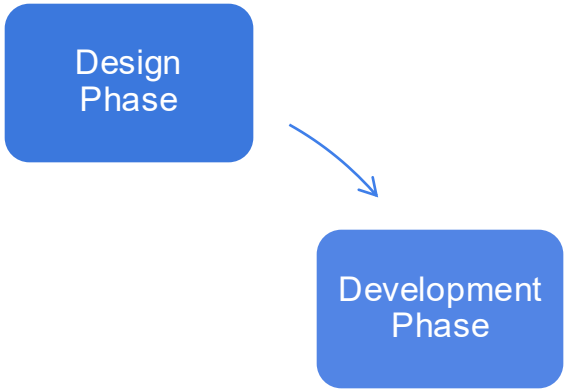
From code to compliance

Design Phase

- Threat modelling identifies the controls to mitigate the risk
- Architects identify the required ENISA SbD principles for a new feature.
- Design SBOM identifies features with SbD mapping

© 2026 APH10 Limited

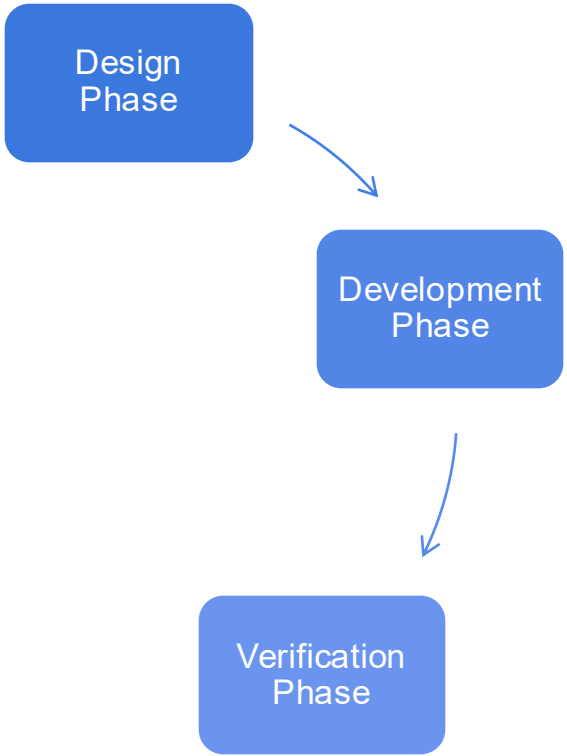
From code to compliance



- Developers implement the feature (e.g. enforcing MFA).
- Source SBOM linked to Design SBOM
- Developers identify components implementing features

© 2026 APH10 Limited

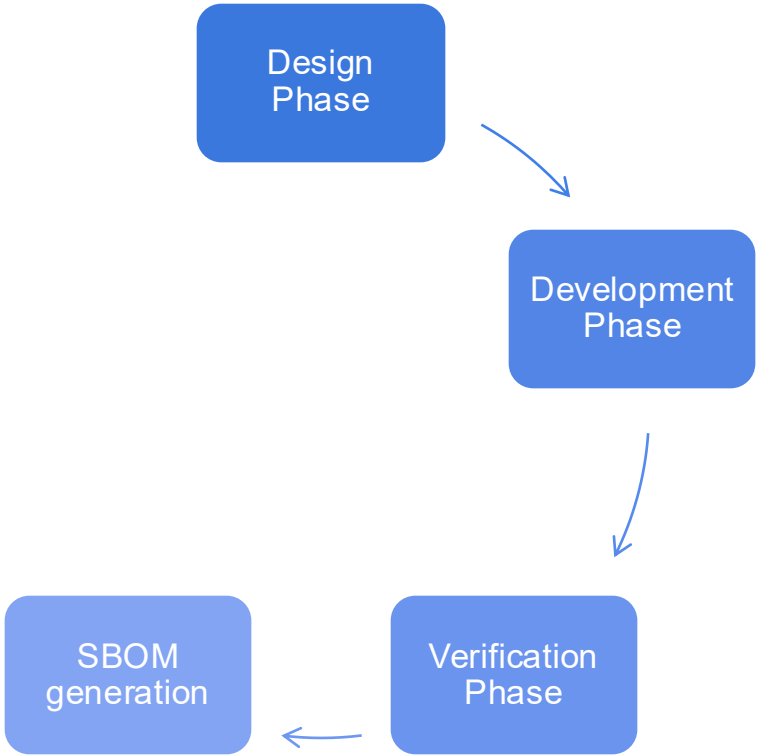
From code to compliance



- Security Team verifies the implementation against threat model

© 2026 APH10 Limited

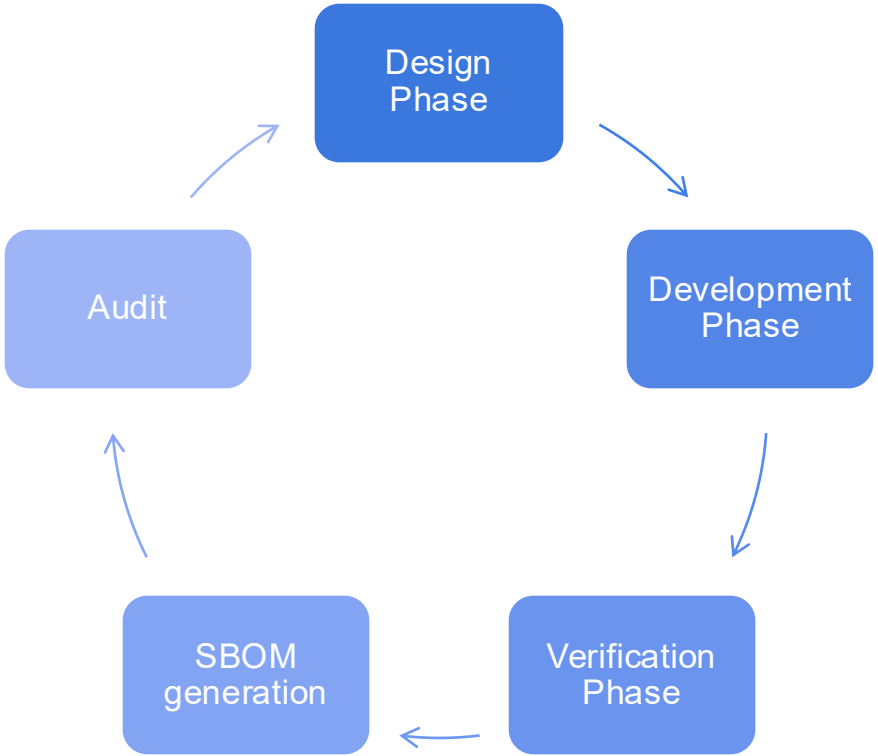
From code to compliance



- The CI/CD pipeline injects the verified properties into the SBOM file.
- Build SBOM linked from Source SBOM

© 2026 APH10 Limited

From code to compliance



- Compliance dashboard flags any component missing a required mapping.

© 2026 APH10 Limited

Summary

- By mapping ENISA principles to security controls (e.g. NIST 800-53) within an SBOM, we can create a Traceable Security Flow
- Next Steps
  - Standardise the Taxonomy
    - Define exactly what "least privilege" looks like
  - Schema Definition
    - Establish the list of required enisa-sbd and nist-800-53 properties
  - Pilot
    - Integrate these properties into a product SBOM
  - Investigate linkage with Threat Modelling (CycloneDX 2.0)

© 2026 APH10 Limited

Contact Details - Anthony Harrison

LinkedIn

GitHub

https://www.linkedin.com/company/aph10/
https://www.linkedin.com/in/anthonypharrison/



Email: anthony@aph10.com

https://www.aph10.com
https://www.sbomeurope.eu
https://www.sbomfocus.eu

APH10

© 2026 APH10 Limited