LEVERAGING OSCAL TO SUPPORT CYBERSECURITY LIFECYCLE MANAGEMENT

Dr. Sara N. Matheu

Dr. Antonio Skarmeta

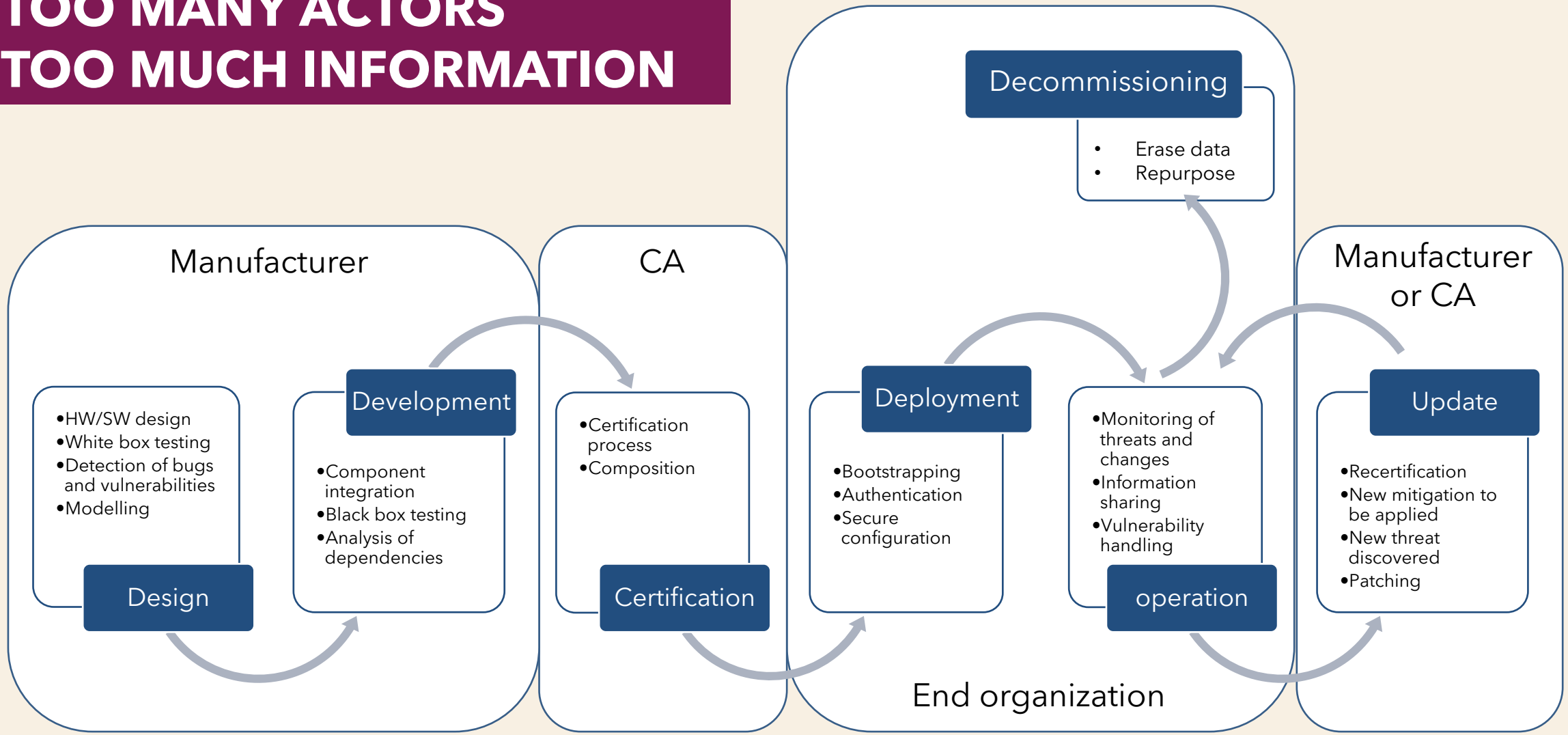
University of Murcia, Spain

AGENDA

- The product Lifecycle
- European projects: COBALT and DOSS
- Using OSCAL as a Device Security Passport
- Synergies with EU regulation
- Recap

THE PRODUCT LIFECYCLE

TOO MANY ACTORS
TOO MUCH INFORMATION



EUROPEAN PROJECTS: COBALT AND DOSS

Leveraging OSCAL for security lifecycle management

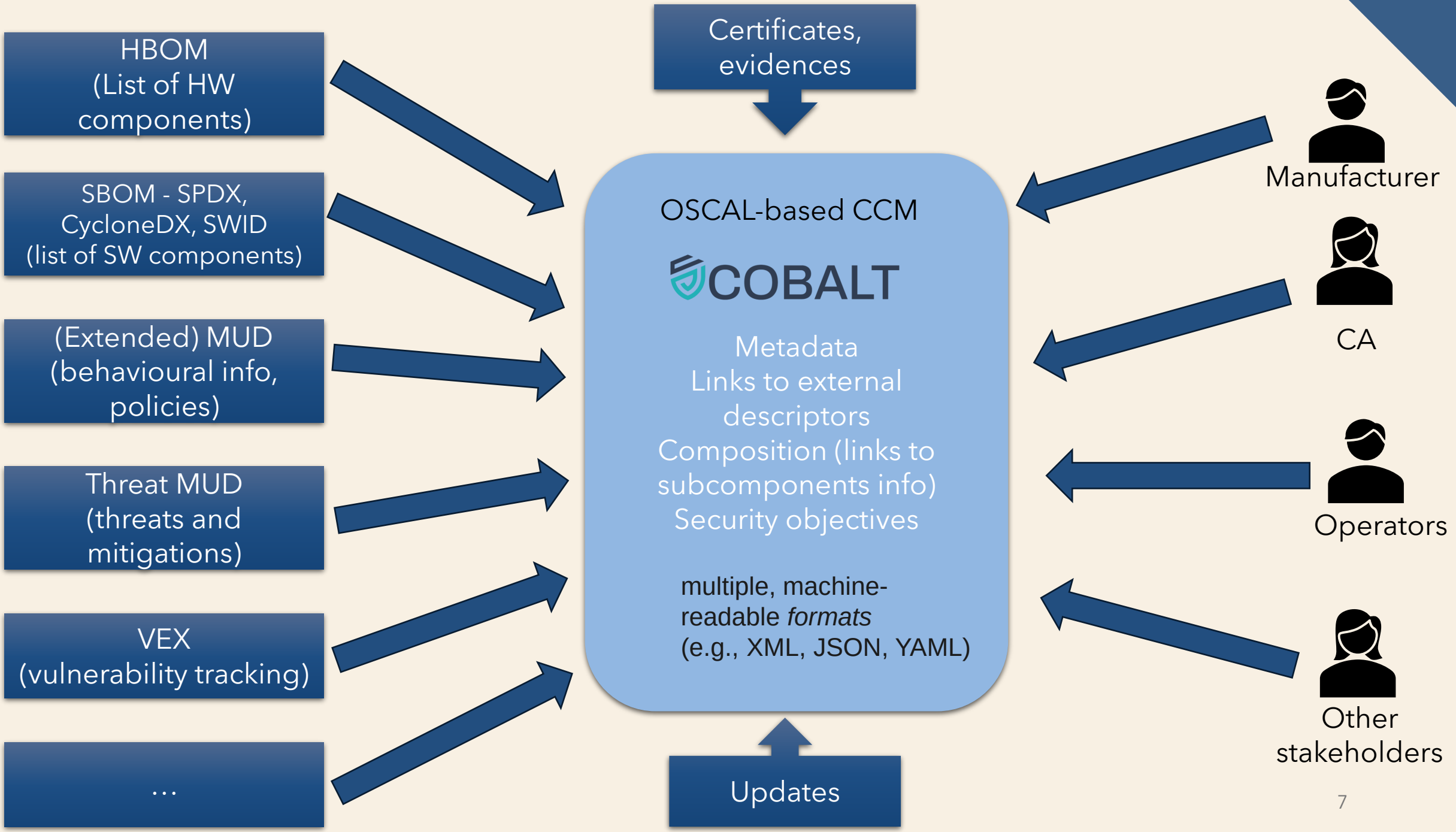
A DEVICE SECURITY PASSPORT

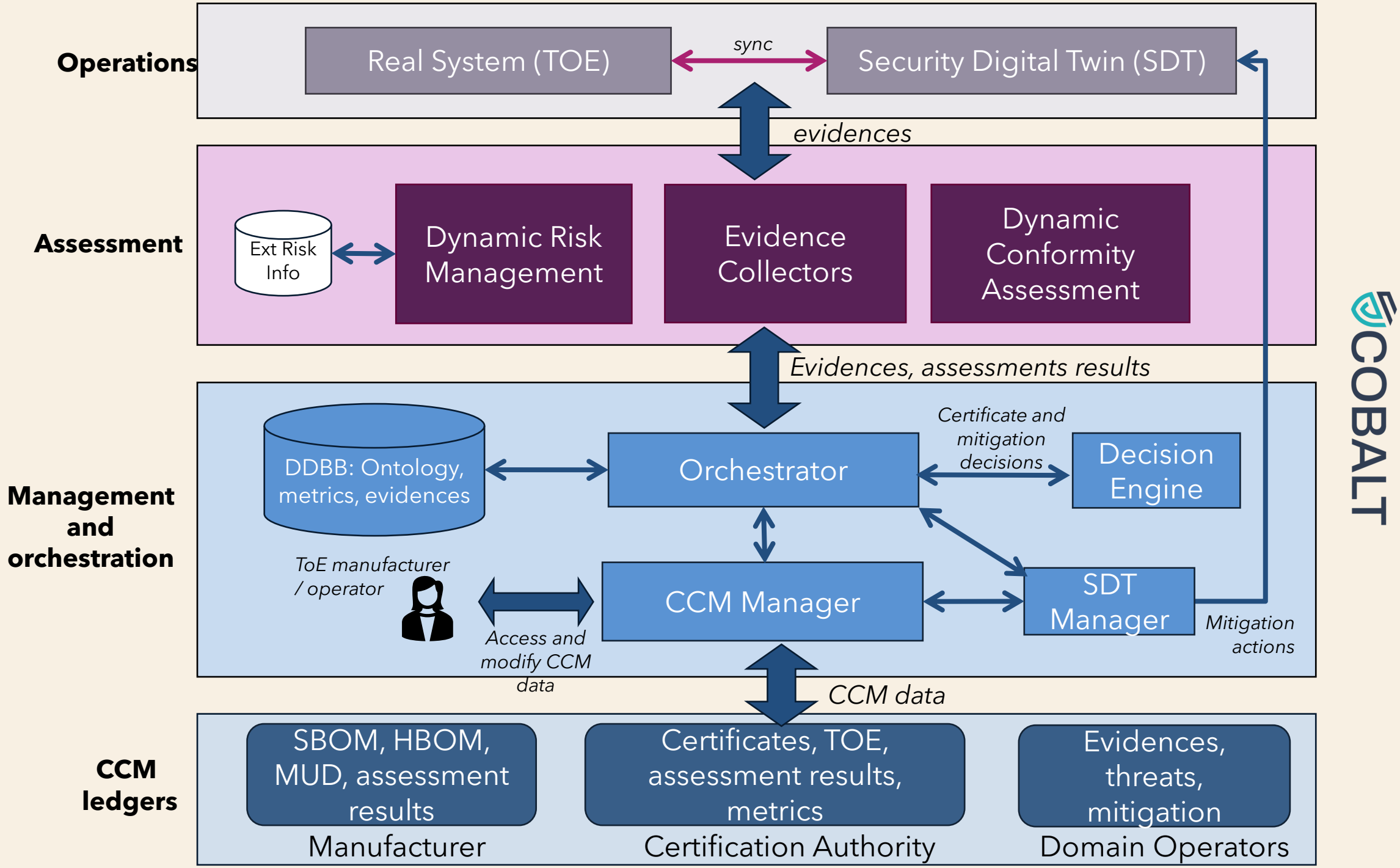
DOSS: SECURE-BY-DESIGN IOT OPERATION WITH SUPPLY CHAIN CONTROL

https://dossproject.eu/

COBALT: CERTIFICATION FOR CYBERSECURITY IN EU ICT USING DECENTRALIZED DIGITAL TWINNING

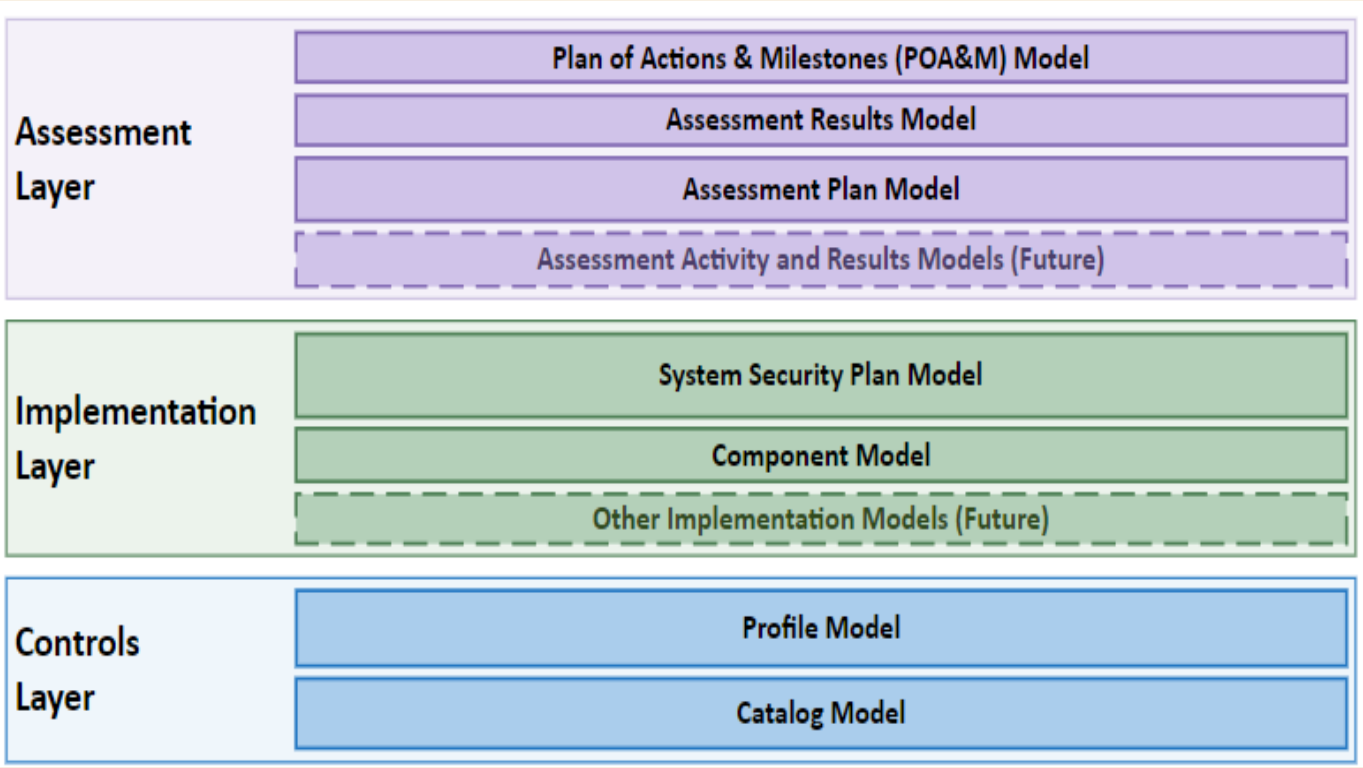
https://horizon-cobalt.eu/


HBOM
(List of HW components)
SBOM - SPDX, CycloneDX, SWID
(list of SW components)
(Extended) MUD
(behavioural info, policies)
Threat MUD
(threats and mitigations)
VEX
(vulnerability tracking)
…
Certificates, evidences
OSCAL-based CCM
COBALT
Metadata
Links to external descriptors
Composition (links to subcomponents info)
Security objectives
multiple, machine-readable formats
(e.g., XML, JSON, YAML)
Updates
Manufacturer
CA
Operators
Other stakeholders


Operations
Real System (TOE)
sync
Security Digital Twin (SDT)
evidences
Assessment
Ext Risk Info
Dynamic Risk Management
Evidence Collectors
Dynamic Conformity Assessment
Evidences, assessments results
Management and orchestration
DDBB: Ontology, metrics, evidences
Orchestrator
Certificate and mitigation decisions
Decision Engine
ToE manufacturer / operator
CCM Manager
SDT Manager
Mitigation actions
Access and modify CCM data
CCM data
CCM ledgers
SBOM, HBOM, MUD, assessment results
Manufacturer
Certificates, TOE, assessment results, metrics
Certification Authority
Evidences, threats, mitigation
Domain Operators

COBALT

USING OSCAL AS A DEVICE SECURITY PASSPORT



https://pages.nist.gov/OSCAL/resources/concepts/layer/#control-layer-overview

Threat MUD, VEX, updates, assesments

Assessment activities, on communicating all assessment findings including supporting **evidence**, and identifying and managing the **remediation** of identified risks to a system identified after assessment activities. This includes the ability to report all findings.

SBOM, HBOM, MUD, link to certificates, how security controls are implemented

Implementation of a system under a specific baseline as well as the individual components that may be incorporated into a system. A **component** is anything that can satisfy a **control**, such as a policy, process, compliance artifact, as well as hardware, software, and services.

Security controls applicable

Cybersecurity frameworks often define a set of **controls intended to reduce the risk to a system**. Organizations and system owners identify which controls are applicable to a system, which may include controls from more than one catalog.

DESIGN AND DEVELOPMENT

Identify and document components contained in the product

- A list of components holds the potential to provide increased **transparency**, **provenance**, and **speed** at which vulnerabilities can be identified and remediated.

  "Am I affected?" and "Where am I affected?"

- How to do it?
  - **BOMs:** A bill of materials (BOM) is a source of information containing a list of items and instructions to design or manufacture a product.
  - Different types: SBOM, HBOM, CBOM, SaaSBOM, ML-BOM, etc.

SBOM

- A formal, machine-readable inventory of software components and dependencies, information about those components, and their hierarchical relationships", like a list of ingredients that make up software components.

- SBOM can be used as a source for SCA tools to find vulnerable third-party libraries in an automated way

Data Field	Description
Supplier Name	The name of an entity that creates, defines, and identifies components.
Component Name	Designation assigned to a unit of software defined by the original supplier.
Version of the Component	Identifier used by the supplier to specify a change in software from a previously identified version.
Other Unique Identifiers	Other identifiers that are used to identify a component, or serve as a look-up key for relevant databases.
Dependency Relationship	Characterizing the relationship that an upstream component X is included in software Y.
Author of SBOM Data	The name of the entity that creates the SBOM data for this component.
Timestamp	Record of the date and time of the SBOM data assembly.

Format	Specification	Known Tools
SPDX	https://spdx.github.io/spdx-spec/	https://tiny.cc/SPDX
CycloneDX	https://cyclonedx.org	https://tiny.cc/CycloneDX
SWID	ISO/IEC 19770-2:2015	https://tiny.cc/SWID

CYCLONEDX (S)BOM

CycloneDX was designed from the ground-up to be a Bill of Materials (BOM) format, capable of capturing complex inventory information for a wide range of cybersecurity and software supply chain use cases.

It is recommended to decouple VEX, VDR, etc. from the BOM.

BOM Type	Description
HBOM	Hardware Bill of Materials
OBOM	Operations Bill of Materials
SaaSBOM	Software-as-a-Service Bill of Materials
SBOM	Software Bill of Materials
VDR	Vulnerability Disclosure Report
VEX	Vulnerability Exploitability eXchange

beyond the minimum required for a SBOM

https://cyclonedx.org/docs/1.5/json/#bomFormat

HBOM

HBOM information is recommended for any product/kit/assembly that can be broken down into lesser components. When arranged in order of assembly level, this breakdown is sometimes referred to as a "hierarchical" bill of materials.

TABLE 3: HBOM DATA FIELD CATEGORIES

Field Category	Definition	Example Data Fields
HBOM Header Information	Identifying information about the HBOM (Finished Good - Descriptive Information and HBOM Author/Dates)	Author, Create/Modify Dates, Product Type, Name, Description, Supplier/OEM
Entity Name	Company Names of Entities in the HBOM	(Contract) Manufacturer Name, Assembly & Test Supplier, Component Manufacturer/Supplier
Entity Location	Company Locations of Entities in the HBOM	Location Details of any Specified Entity
Finished Good Product Details	Finished Good: Technical Information	Product Version
Component Part Information	Component: Descriptive Information	Component Type, Category, Number, Description
Component Part Details	Component: Technical Information	Component Version, Tech Specs
Production Details	Production/Operational Information	% Sourced from Supplier, Lead Times, Quantity, Tech Node

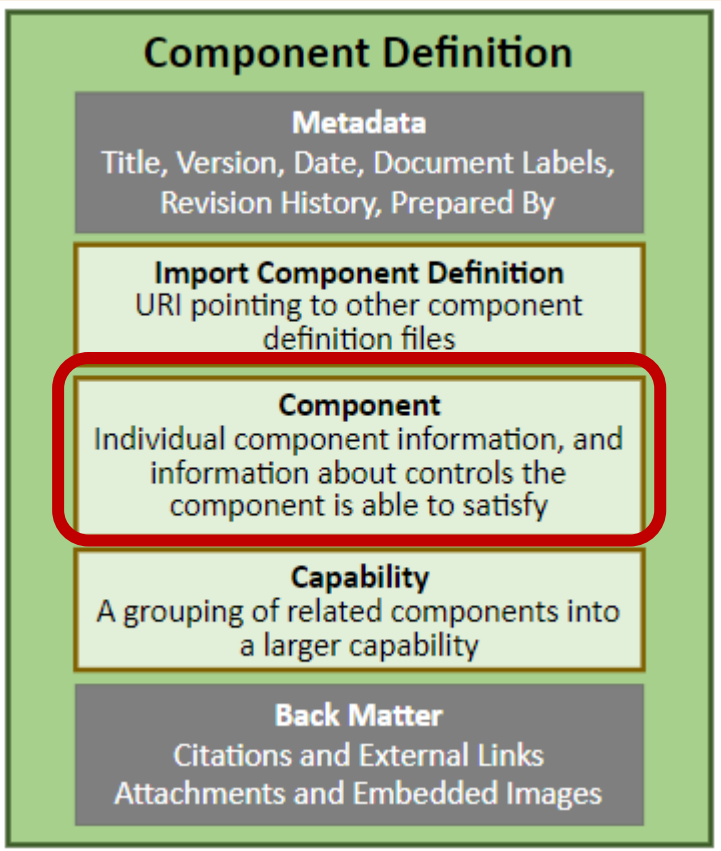
https://www.cisa.gov/sites/default/files/2023-09/A%20Hardware%20Bill%20of%20Materials%20Framework%20for%20Supply%20Chain%20Risk%20Management%20%28508%29.pdf

MAPPING WITH OSCAL

Component definition model: Provides information about a defined component that can be part of an implemented system. Components may be SW, HW, services, APIs, policies, processes, plans, guidance, standards, or other tangible items that enable security and/or privacy. The ***type*** field indicates which of these component types is represented.

Link to SBOM and HBOM depending on component type (even distinguishing formats)

```
▼ component-definition [1]: {
    uuid [1]: uuid,
  ▶ metadata [1]: { … },
  ▶ import-component-definitions [0 or 1]: [ … ],
  ▼ components [0 or 1]: [
      An array of component objects [1 to ∞] {
          uuid [1]: uuid,
          type [1]: string,
          title [1]: markup-line,
          description [1]: markup-multiline,
          purpose [0 or 1]: markup-line,
        ▶ props [0 or 1]: [ … ],
        ▶ links [0 or 1]: [ … ],
        ▶ responsible-roles [0 or 1]: [ … ],
        ▶ protocols [0 or 1]: [ … ],
        ▶ control-implementations [0 or 1]: [ … ],
          remarks [0 or 1]: markup-multiline,
      }
  ],
  ▶ capabilities [0 or 1]: [ … ],
  ▶ back-matter [0 or 1]: { … },
}
```

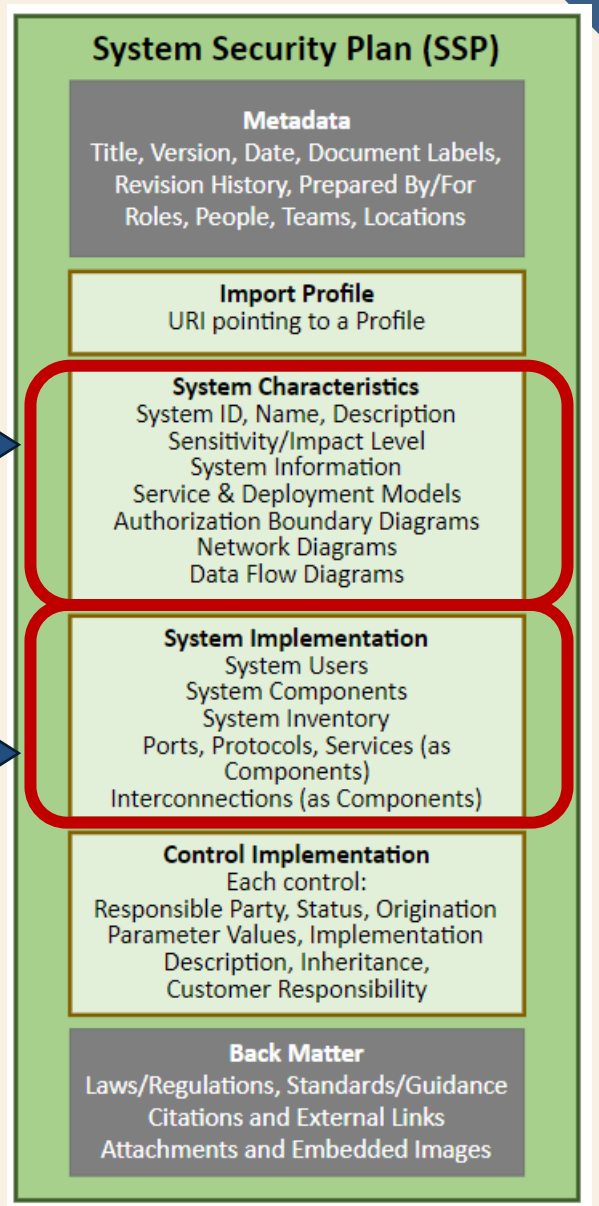


MAPPING WITH OSCAL

Other BOMs: Work In progress
(we started by HBOM, SBOM)

Link to SaaSBOM?
There are some fields here to describe cloud based systems

Link to ML-BOM, MBOM, OBOM?



DESIGN AND DEVELOPMENT

Identify and document vulnerabilities contained in the product

- Similarly to the list of components, a list of vulnerabilities provides transparency to the supply chain, and ensures a smooth analysis of cascade effects in case a vulnerability is exploited

- How to do it?

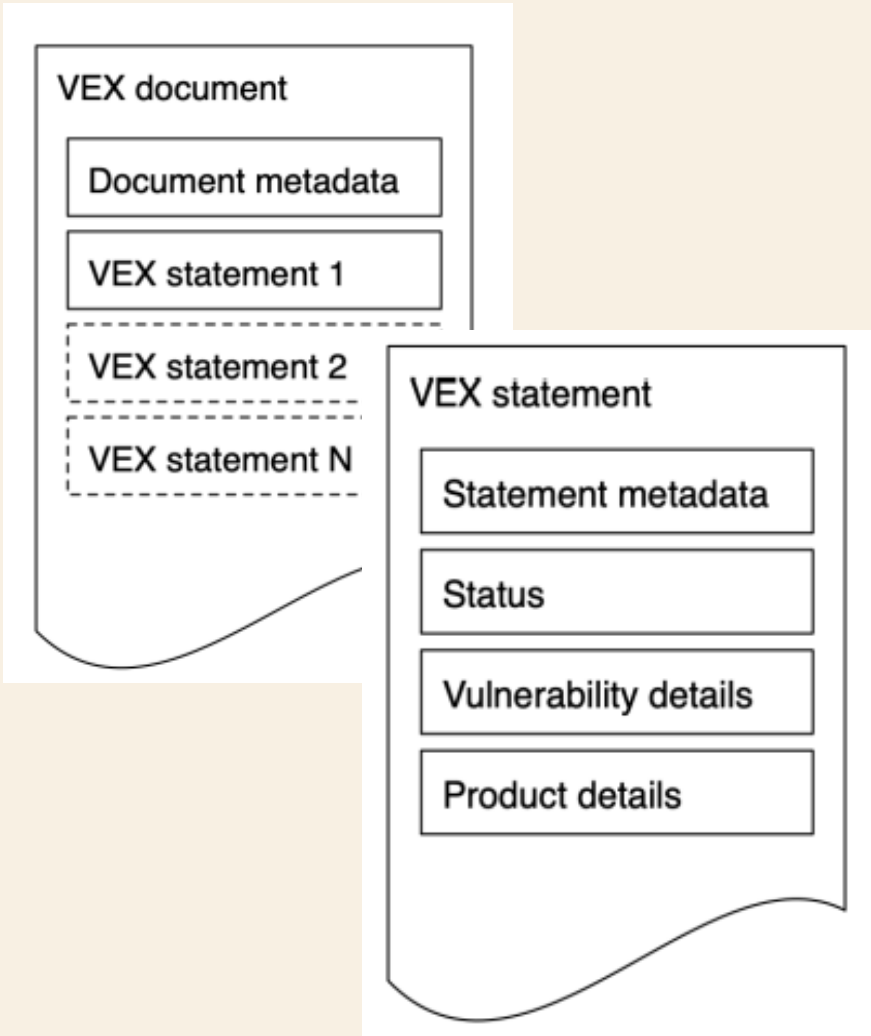
  - **Vulnerability Exploitability eXchange (VEX):** List vulnerabilities a product is not affected by. It does not require the vendor to check dependencies.

  - **Vulnerability Disclosure Report (VDR):** lists vulnerabilities that affects a product or its dependencies.

- Even if that information can be included in the SBOM, it is recommended to decouple from it as new vulnerabilities can be discovered more often than new components are added to the product.

VULNERABILITY EXPLOITABILITY EXCHANGE (VEX)

- Minimum data elements of a VEX document (2023):
  - **Document metadata**: ID, version, authors, tooling, Timestamp.
  - **VEX statement:** convey a single [status] that applies to a single [vul_id] for one or more [product_id]s
    - **Metadata**: ID, version, timestamp
    - **Product details:** Product identifier(s), subcomponent id, supplier
    - **Vulnerability details:** Identifier of the Vulnerability (CVE or other identifier) and vulnerability description (e.g. CVE description).
    - **Product Status details** (i.e., status information about a vulnerability in that product)
      - NOT AFFECTED – No remediation is required regarding this vulnerability.
      - AFFECTED – Actions are recommended to remediate or address this vulnerability.
      - FIXED – These product versions contain a fix for the vulnerability.
      - UNDER INVESTIGATION – It is not yet known whether these product versions are affected by the vulnerability. An update will be provided in a later release.
- CSAF, CycloneDX, and OpenVEX can contain or be used to generate VEX documents. Both CSAF and CycloneDX provide information beyond what is required for VEX.



https://www.cisa.gov/sites/default/files/2023-04/minimum-requirements-for-vex-508c.pdf

VEX-CYCLONE DX

```
{
  "bomFormat": "CycloneDX",
  "specVersion": "1.4",
  "version": 1,
  "vulnerabilities": [
    {
      "id": "CVE-2020-25649",
      "source": {
        "name": "NVD",
        "url": "https://nvd.nist.gov/vuln/detail/CVE-2020-25649"
      },
      "references": [
        {
          "id": "SNYK-JAVA-COMFASTERXMLJACKSONCORE-1048302",
          "source": {
            "name": "SNYK",
            "url": "https://security.snyk.io/vuln/SNYK-JAVA-COMFA
          }
        }
      ],
      "ratings": [
        {
          "source": {
            "name": "NVD",
            "url": "https://nvd.nist.gov/vuln-metrics/cvss/v3-cal
          },
          "score": 7.5,
          "severity": "high",
          "method": "CVSSv31",
          "vector": "AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:H/A:N"
        },
        {
        {
      ],
      "cwes": [
```

```
    "cwes": [
      611
    ],
    "description": "com.fasterxml.jackson.core:jackson-databind is a libr
    "detail": "XXE Injection is a type of attack against an application t
    "recommendation": "Upgrade com.fasterxml.jackson.core:jackson-databir
    "advisories": [
      {
        "title": "GitHub Commit",
        "url": "https://github.com/FasterXML/jackson-databind/commit/612f
      },
      {
        "title": "GitHub Issue",
        "url": "https://github.com/FasterXML/jackson-databind/issues/2589
      },
      {
        "title": "RedHat Bugzilla Bug",
        "url": "https://bugzilla.redhat.com/show_bug.cgi?id=1887664"
      }
    ],
    "created": "2020-12-03T00:00:00.000Z",
    "published": "2020-12-03T00:00:00.000Z",
    "updated": "2021-10-26T00:00:00.000Z",
    "credits": {
      "individuals": [
        {
          "name": "Bartosz Baranowski"
        }
      ]
    },
    "analysis": {
      "state": "not_affected",
      "justification": "code_not_reachable",
      "response": ["will_not_fix", "update"],
      "detail": "Automated dataflow analysis and manual code review indic
    },
    "affects": [
      {
        "ref": "urn:cdx:3e671687-395b-41f5-a30f-a58921a69b79/1#pkg:maven/
      }
```

https://github.com/oasis-tcs/csaf/blob/master/csaf_2.0/examples/csaf/csaf_vex/2022-evd-uc-01-ui-001.json

VULNERABILITY DISCLOSURE REPORT (VDR)

- VDRs are an attestation of **all vulnerabilities affecting a product**, or its' dependencies, along with an analysis of the impact.
- VDRs may come from a software supplier or a third party.
- **A VDR should contain:**
  - All vulnerabilities affecting a product or its' dependencies
  - Analysis describing the impact (or lack thereof) that a reported vulnerability has on a product or dependency
  - Plans to address the vulnerability
  - Signing the VDR with a trusted, verifiable, private key that includes a timestamp indicating the date and time of the VDR signature

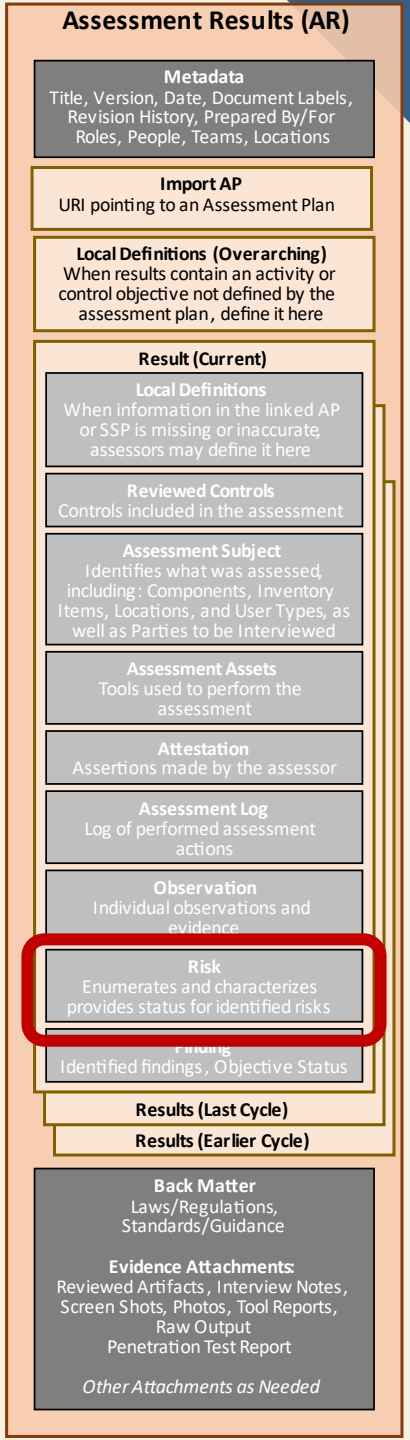
https://csrc.nist.gov/pubs/sp/800/161/r1/final

MAPPING WITH OSCAL

•**Metadata**: file's title, publication version, publication date, and OSCAL version.
•**Import AP**: Identifies the OSCAL-based assessment plan (AP) for this assessment.
•**Local Definitions**
•**Results**: Describes the assessment findings, identified risks, and recommended remediation. Also identifies false positive results, risk adjustments, and operationally required risks, as well as when the results should expire.

- **Local Definitions**
- **Reviewed Controls**: Identifies the controls actually reviewed by this assessment.
- **Assessment Subject**: Identifies the in-scope elements of the system, including locations, components, inventory items, and users.
- **Assessment Assets**: Identifies the assessor's assets used to perform the assessment, including the team, tool, and rules of engagement content.
- **Attestation**: Assertions made by the assessor.
- **Assessment Log**: Log of performed assessment actions.
- **Observation**: Individual observations and related evidence.
- **Risk**: Identifies individual risks, including weakness description, risk statement, and other risk characteristics.
- **Finding**: Identifies findings resulting from observations and risks, and can include the control objective status.

•**Back Matter**: attachments, citations, and embedded content such as graphics.

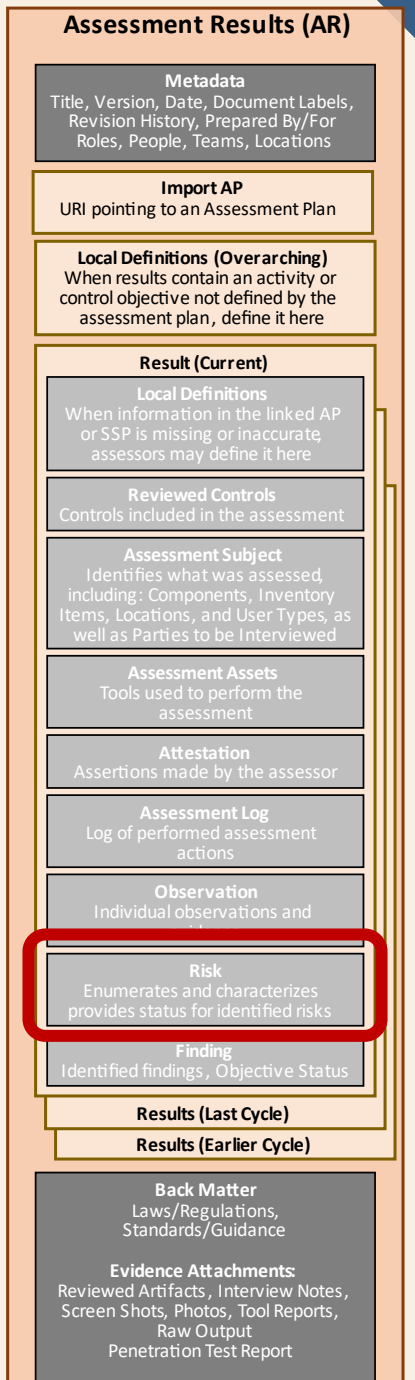


- **AR model**: This model is typically used by anyone performing assessment or continuous monitoring activities on a system to determine the degree to which that system complies with one or more frameworks.
- **Some information is repeated in VEX/VDR and AR:** Threat ID (CVE) and status (fixed, no affected, affected…)

```
▼ assessment-results [1]: {
    uuid [1]: uuid,
  ▶ metadata [1]: { … },
  ▶ import-ap [1]: { … },
  ▶ local-definitions [0 or 1]: { … },
  ▼ results [1]: [
      An array of result objects [1 to ∞] {
        uuid [1]: uuid,
        title [1]: markup-line,
        description [1]: markup-multiline,
        start [1]: date-time-with-timezone,
        end [0 or 1]: date-time-with-timezone,
      ▶ props [0 or 1]: [ … ],
      ▶ links [0 or 1]: [ … ],
      ▶ local-definitions [0 or 1]: { … },
      ▶ reviewed-controls [1]: { … },
      ▶ attestations [0 or 1]: [ … ],
      ▶ assessment-log [0 or 1]: { … },
      ▶ observations [0 or 1]: [ … ],
      ▼ risks [0 or 1]: [
          An array of risk objects [1 to ∞] {
            uuid [1]: uuid,
            title [1]: markup-line,
            description [1]: markup-multiline,
            statement [1]: markup-multiline,
          ▶ props [0 or 1]: [ … ],
          ▶ links [0 or 1]: [ … ],
            status [1]: token,
          ▶ origins [0 or 1]: [ … ],
          ▶ threat-ids [0 or 1]: [ …
          ▶ characterizations [0 or 1]: [ … ],
          ▶ mitigating-factors [0 or 1]: [ … ],
            deadline [0 or 1]: date-time-with-timezone,
          ▶ remediations [0 or 1]: [ … ],
          ▶ risk-log [0 or 1]: { … },
          ▶ related-observations [0 or 1]: [ … ],
```

Link to VEX, VDR identified vulnerabilities and its status

VEX, VDR info



CERTIFICATION AND DEPLOYMENT

apply effective and regular tests and reviews of the security

ensure an appropriate level of cybersecurity, without any known vulnerabilities

secure by default configuration

- Previous information can be used to smooth the certification process, supporting certificate composition
  - List of components, including certificates granted
  - List of vulnerabilities associated with each component
- Previous assessments and evidence can be also stored and shared to support an agile certification
- Product deployment can benefit from the certification results
  - A secure configuration can be established from the assessment and be enforced during deployment
  - Different configurations can be considered for different contexts
- How to do it?
  - **OSCAL (NIST)** provides machine-readable representations of control catalogues, control baselines, system security plans, and assessment plans and results.
  - **MUD file:** IETF standard to express device behaviour at network layer. MUD policies can be enforced during device deployment

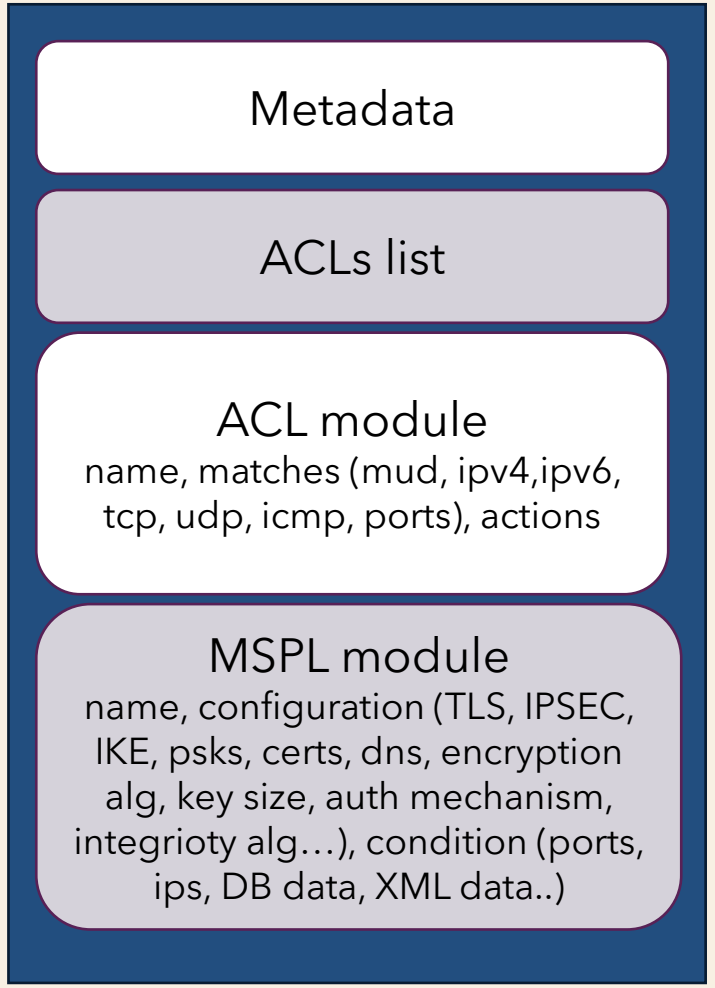
(EXTENDED) MUD FILE

- MUD is intended to be created by manufacturer

- Defines behavioral information associated with a device
  - IETF Standard: Only network-limited content (ports, protocols)
  - UMU Extended: Parental control, filtering, data protection, channel protection. IETF compatible.

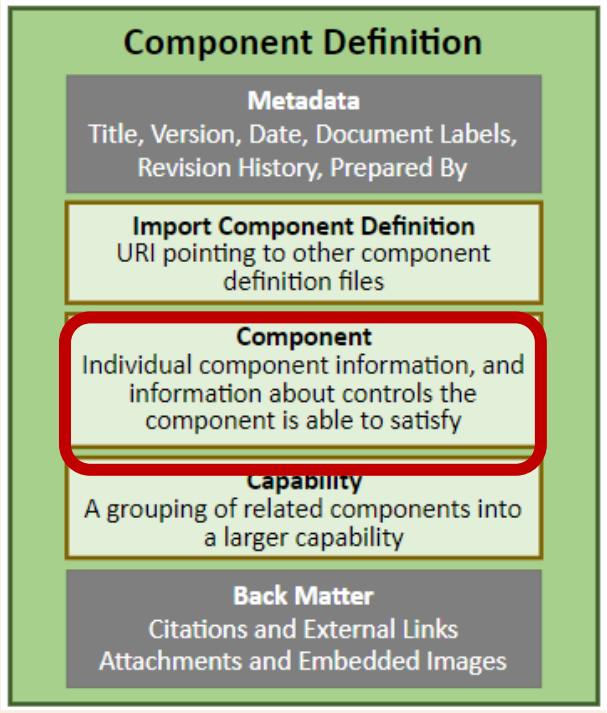
- **Metadata** : version, url, dates, signature, model, manufacturer, etc.
- **ACLs list:** list of ACLs and MSPLs from and to device (only names)
- **ACL module:** Specific policies for access control at network layer following MUD standard (YANG)
- **MSPL module:** Specific policies for filtering, parental control and data protection (MSPL, extension)

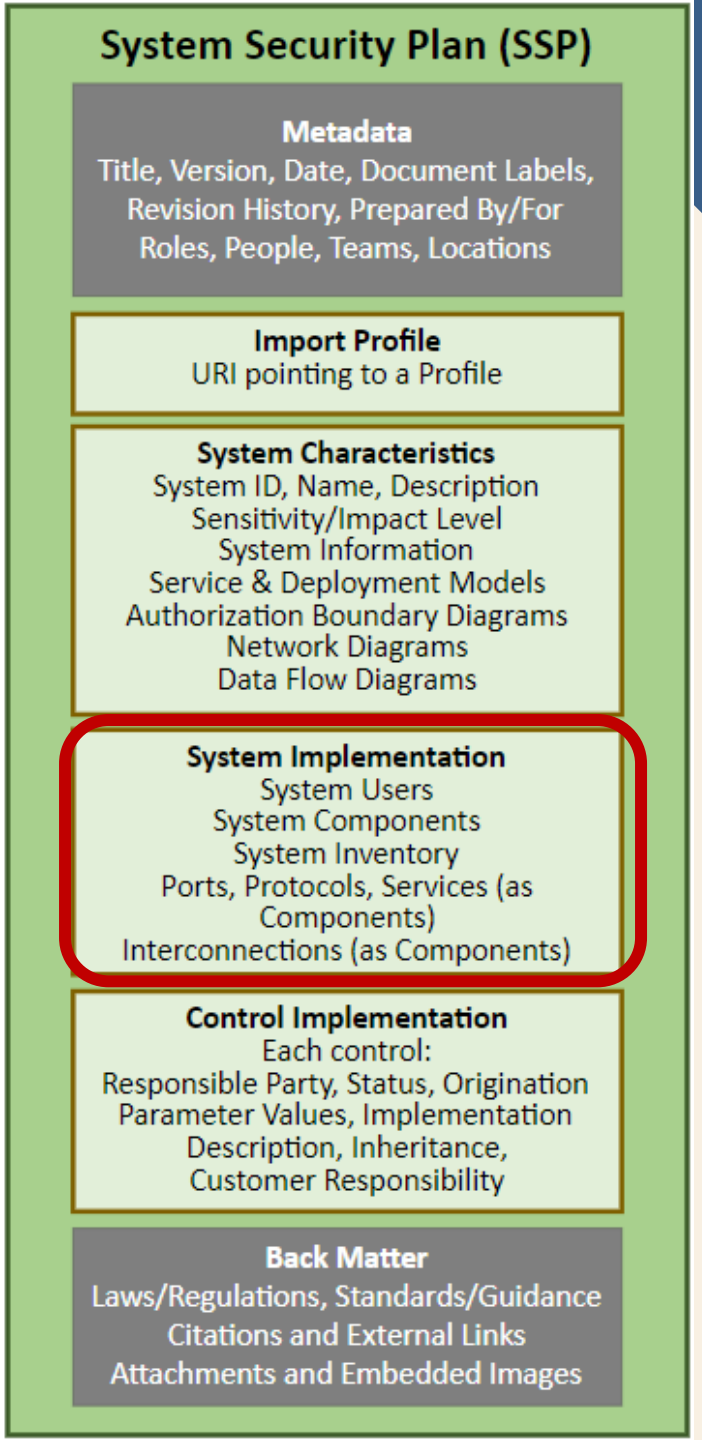
- MUD can be obtained during the bootstrapping of the device, so policies are enforced to reduce the attack surface and provide a secure configuration before the device can access to the domain



MAPPING WITH OSCAL

- **Metadata** (common) : document's title, publication version, publication date, and OSCAL version, roles, parties and locations.
- **Import**: Identifies a component definition from another resource, from which related information is referenced.
- **Component**: Provides information about a defined component that can be part of an implemented system.
- **Capability**: Identifies a capability as a grouping of multiple components or capabilities.
- **Back Matter** (common) : attachments, citations, and embedded content such as graphics.



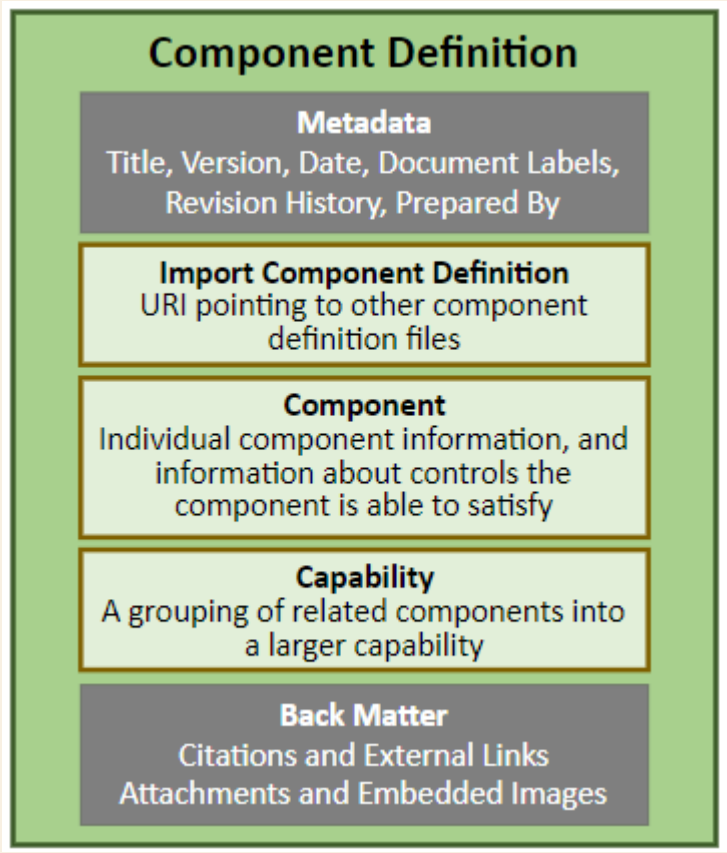


MAPPING WITH OSCAL COMPONENT/SSP

```
component-definition:
  uuid: a7ba800c-a432-44cd-9075-0862cd66da6b
  metadata:
  import-component-definitions:  #link to other components (composability)
  components:
  - uuid: 91f646c5-b1b6-4786-9ec3-2305a044e217
    type: software
    title: subcomponent_1
    description: >-
      subcomponent_1 is a component for doing something.
    purpose: Provides something
    links:
      - href: <https://mudfile.json> #link to mud of this component mud-file
        media-type: application/json
        text: mud-file
    responsible-roles:   #liability
    - role-id: provider
      party-uuid: ef7c799a-c50e-49ab-83e0-515e989e6df1
    protocols:  #this is already in MUD. We can remove protocols section (or
    - uuid: 2b4a1b3a-cbc5-4cc8-bde6-7437c28c4e54
      name: process1 #maybe link with ace name in MUD to facilitate matching
      title: Primary daemon process for example system.
      port-ranges:
      - start: 27017
        end: 27017
        transport: TCP
    control-implementations:
    - uuid: 49f0b690-ed9f-4f32-aae0-625b77aa6d27
      source: <https://MODERATE-baseline_profile.json> #link to profile
      description: >-
        component implementations for NIST SP 800-53rev5.
      implemented-requirements:
      - uuid: cf8338c5-fb6e-4593-a4a8-b3c4946ee2a0
        control-id: sc-8.1
        description: >-
          component supports TLS 1.x to encrypt data in transit. To implement
  capabilities: ~
  back-matter: ~
```

Link to MUD

Duplicated (MUD info)



OPERATION AND UPDATE

Address, remediate and disclose vulnerabilities

Provide and securely distribute updates

Apply effective and regular tests and reviews of the security

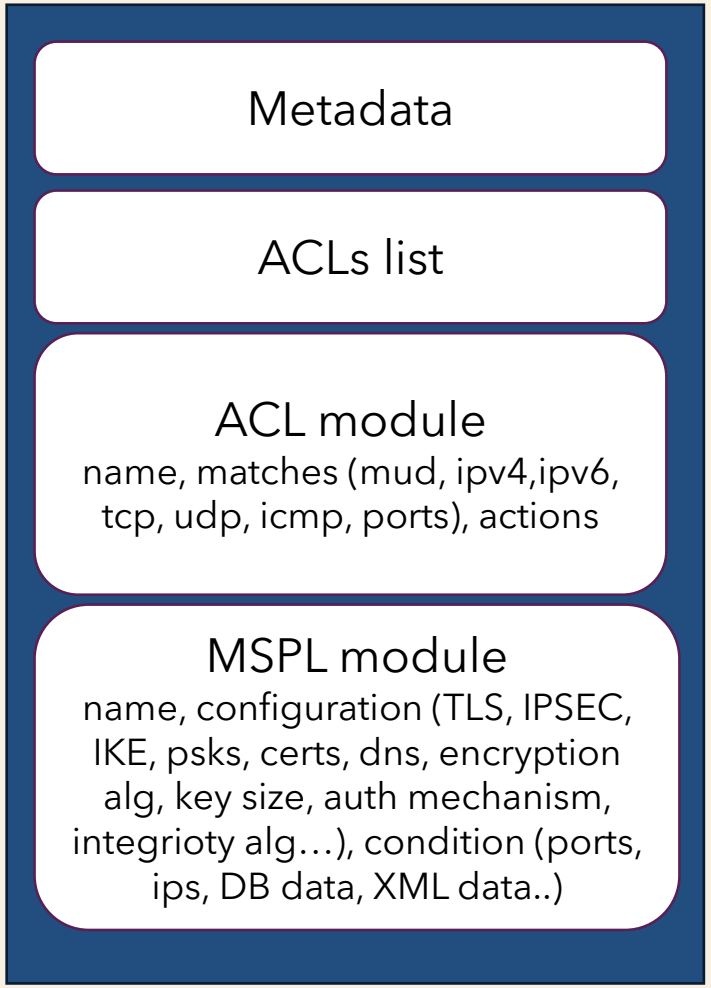
- Vulnerability and updates may change the security level achieved in a certificate
- Vulnerability handling and upgrading should be connected with certification
  - Disclosure of vulnerabilities to the manufacturer and CA → Possible recertification
  - Secure patching/mitigation approved by CA → Maintain security level

How to do it?

- **Threat MUD:** NIST document based on IETF MUD to share mitigations associated with vulnerabilities
- Vulnerability management: SIEM, IDS, CTI sharing

(EXTENDED) THREAT MUD

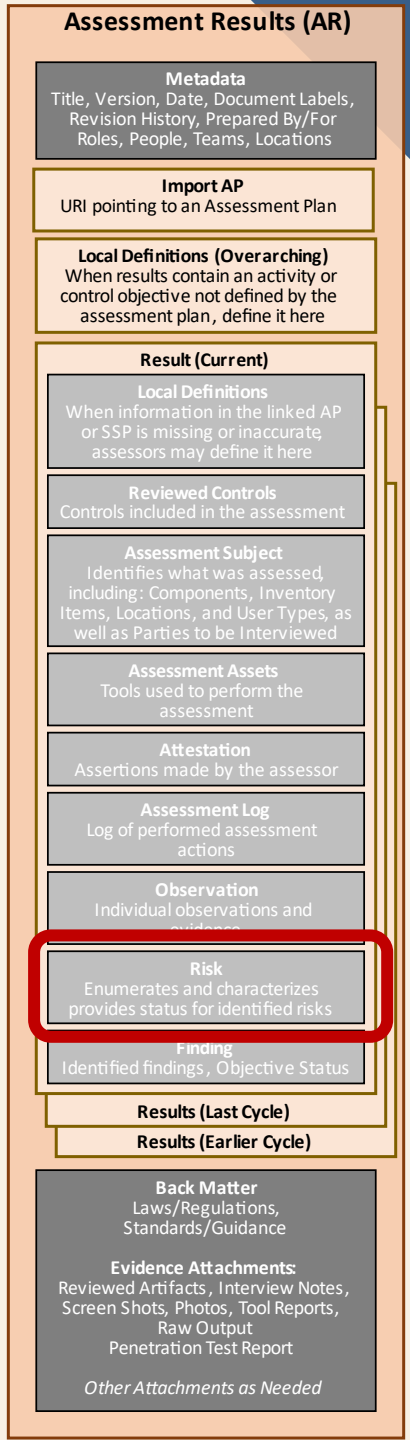
- Similar to MUD but linked to a threat instead of a device
- VEX and VDR give more info about the threat, status and affected products, Threat MUD focuses on mitigations.
- **Metadata** : version, url, dates, signature, threat intelligence provider, threat name, cvss vector, documentation
- **ACLs list**: list of ACLs and MSPLs from and to device (only names)
- **ACL module**: Specific policies for access control at network layer following MUD standard (YANG)
- **MSPL module**: Specific policies for filtering, parental control and data protection (extended MSPL)



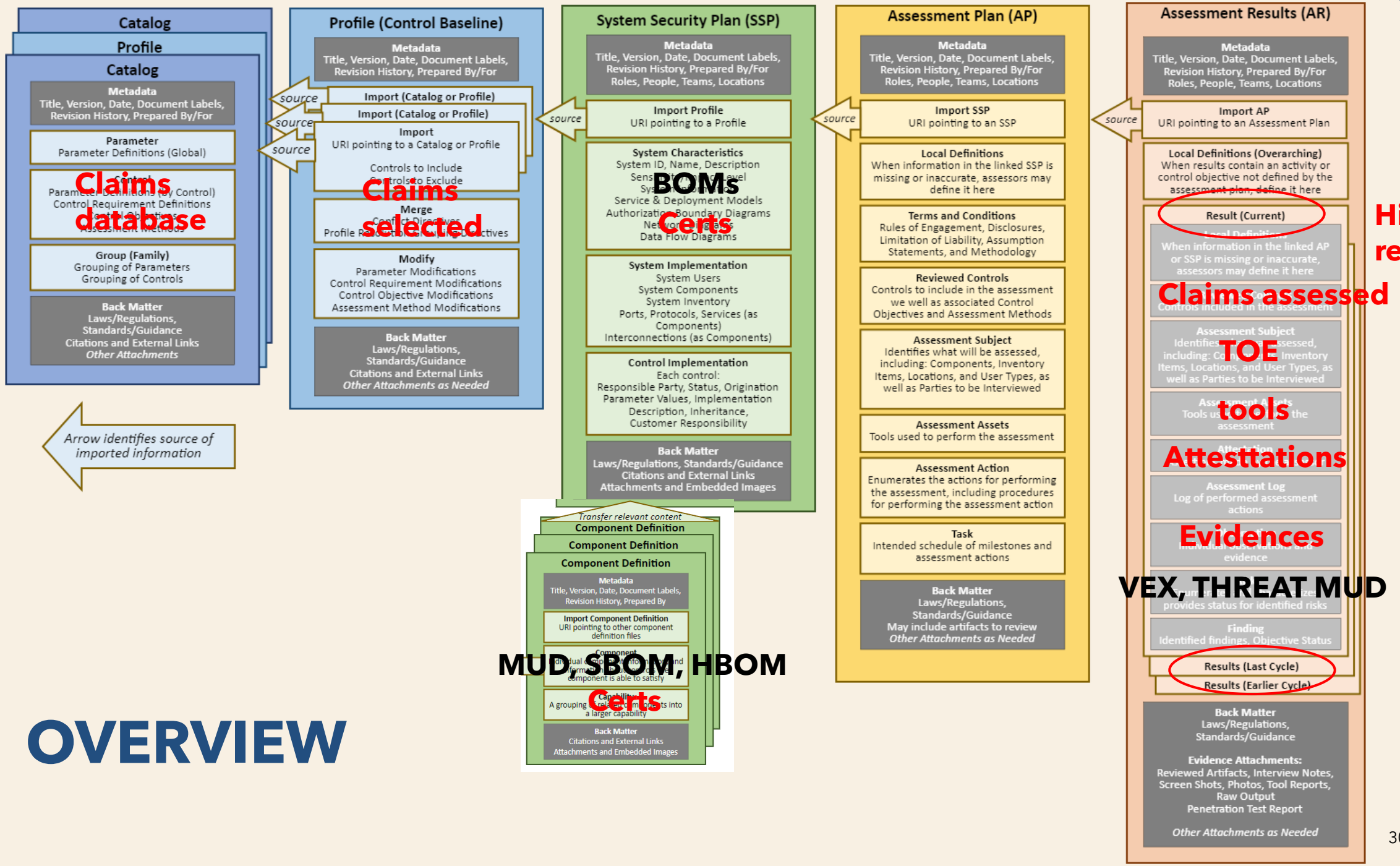
MAPPING WITH OSCAL

```
▶ observations [0 or 1]: [ … ],
▼ risks [0 or 1]: [
      An array of risk objects [1 to ∞] {
          uuid [1]: uuid,
          title [1]: markup-line,
          description [1]: markup-multiline,
          statement [1]: markup-multiline,
        ▶ props [0 or 1]: [ … ],
        ▶ links [0 or 1]: [ … ],
          status [1]: token,
        ▶ origins [0 or 1]: [ … ],
        ▶ threat-ids [0 or 1]: [ … ],
        ▶ characterizations [0 or 1]: [ … ],
        ▶ mitigating-factors [0 or 1]: [ … ],
          deadline [0 or 1]: date-time-with-timezone,
        ▼ remediations [0 or 1]: [
              An array of response objects [1 to ∞] {
                  uuid [1]: uuid,
                  lifecycle [1]: token,
                  title [1]: markup-line,
                  description [1]: markup-multiline,
                ▶ props [0 or 1]: [ … ],
                ▶ links [0 or 1]: [ … ],
                ▶ origins [0 or 1]: [ … ],
                ▶ required-assets [0 or 1]: [ … ],
                ▶ tasks [0 or 1]: [ … ],
                  remarks [0 or 1]: markup-multiline,
              }
          ],
        ▶ risk-log [0 or 1]: { … },
        ▶ related-observations [0 or 1]: [ … ],
      }
  ],
▶ findings [0 or 1]: [ … ],
```

Link (or prop) to Threat MUD for updated remediations



https://pages.nist.gov/OSCAL-Reference/models/v1.1.2/assessment-results/json-reference/


Catalog
Profile
Catalog
Metadata
Title, Version, Date, Document Labels, Revision History, Prepared By/For
Parameter
Parameter Definitions (Global)
Control
Parameter Definitions (by Control)
Control Requirement Definitions
Control Objectives
Assessment Methods
Group (Family)
Grouping of Parameters
Grouping of Controls
Back Matter
Laws/Regulations, Standards/Guidance
Citations and External Links
Other Attachments
Claims database
Profile (Control Baseline)
Metadata
Title, Version, Date, Document Labels, Revision History, Prepared By/For
source
Import (Catalog or Profile)
Import (Catalog or Profile)
Import
URI pointing to a Catalog or Profile
Controls to Include
Controls to Exclude
Claims selected
Merge
Conflict Directives
Profile Parameter Modifications
Profile Control Objectives
Modify
Parameter Modifications
Control Requirement Modifications
Control Objective Modifications
Assessment Method Modifications
Back Matter
Laws/Regulations, Standards/Guidance
Citations and External Links
Other Attachments as Needed
Arrow identifies source of imported information
System Security Plan (SSP)
Metadata
Title, Version, Date, Document Labels, Revision History, Prepared By/For Roles, People, Teams, Locations
source
Import Profile
URI pointing to a Profile
System Characteristics
System ID, Name, Description
Service & Deployment Models
Authorization Boundary Diagrams
Network Diagrams
Data Flow Diagrams
BOMs
certs
System Implementation
System Users
System Components
System Inventory
Ports, Protocols, Services (as Components)
Interconnections (as Components)
Control Implementation
Each control:
Responsible Party, Status, Origination
Parameter Values, Implementation Description, Inheritance, Customer Responsibility
Back Matter
Laws/Regulations, Standards/Guidance
Citations and External Links
Attachments and Embedded Images
Transfer relevant content
Component Definition
Component Definition
Component Definition
Metadata
Title, Version, Date, Document Labels, Revision History, Prepared By
Import Component Definition
URI pointing to other component definition files
Component
Capability
A grouping of related components into a larger capability
Back Matter
Citations and External Links
Attachments and Embedded Images
Certs
MUD, SBOM, HBOM
Assessment Plan (AP)
Metadata
Title, Version, Date, Document Labels, Revision History, Prepared By/For Roles, People, Teams, Locations
source
Import SSP
URI pointing to an SSP
Local Definitions
When information in the linked SSP is missing or inaccurate, assessors may define it here
Terms and Conditions
Rules of Engagement, Disclosures, Limitation of Liability, Assumption Statements, and Methodology
Reviewed Controls
Controls to include in the assessment we well as associated Control Objectives and Assessment Methods
Assessment Subject
Identifies what will be assessed, including: Components, Inventory Items, Locations, and User Types, as well as Parties to be Interviewed
Assessment Assets
Tools used to perform the assessment
Assessment Action
Enumerates the actions for performing the assessment, including procedures for performing the assessment action
Task
Intended schedule of milestones and assessment actions
Back Matter
Laws/Regulations, Standards/Guidance
May include artifacts to review
Other Attachments as Needed
Assessment Results (AR)
Metadata
Title, Version, Date, Document Labels, Revision History, Prepared By/For Roles, People, Teams, Locations
source
Import AP
URI pointing to an Assessment Plan
Local Definitions (Overarching)
When results contain an activity or control objective not defined by the assessment plan, define it here
Result (Current)
When information in the linked AP or SSP is missing or inaccurate, assessors may define it here
Claims assessed
Assessment Subject
TOE
tools
Attesttations
Assessment Log
Log of performed assessment actions
Evidences
evidence
VEX, THREAT MUD
provides status for identified risks
Finding
Identified findings. Objective Status
Results (Last Cycle)
Results (Earlier Cycle)
Historical results
Back Matter
Laws/Regulations, Standards/Guidance
Evidence Attachments:
Reviewed Artifacts, Interview Notes, Screen Shots, Photos, Tool Reports, Raw Output
Penetration Test Report
Other Attachments as Needed

OVERVIEW

SYNERGIES WITH EU REGULATION

EUROPE TOWARDS CYBERSECURITY CERTIFICATION AND LIFECYCLE MANAGEMENT

	NIS2	**OSCAL for transparency**	To **promote cooperation and information exchange** among EU Member States to prevent and respond to cybersecurity incidents Address **supply chain security** Establish relationships with high-risk third-party service partners/providers/vendors and **make them aware of risks**
	CSA	**OSCAL for composition**	To create a **common framework for the cybersecurity certification** of any ICT product, service, or process **Monitoring** compliance with certification requirements Use of **repositories** listing vulnerabilities
	CRA	**OSCAL for security lifecycle management**	To enhance cybersecurity and cyber resilience in the EU Mandates manufacturers **to manage security throughout the product's lifecycle** (updates for 5 years, handling of new vulnerabilities) Establishes **essential requirements**

CRA AND LIFECYCLE MANAGEMENT - ESSENTIAL CYBERSECURITY REQUIREMENTS

Manufacturers:

- identify and document vulnerabilities and components contained in the product, including software bill of materials - Development
- apply effective and regular tests and reviews of the security –Development, certification, operation
- address and remediate vulnerabilities - Operation
- disclose information about fixed vulnerabilities - Operation
- coordinated vulnerability disclosure; - Operation
- sharing of information about potential vulnerabilities - Operation
- provide for mechanisms to securely distribute updates - Upgrading
- providing security updates – Upgrading

Products with digital elements:

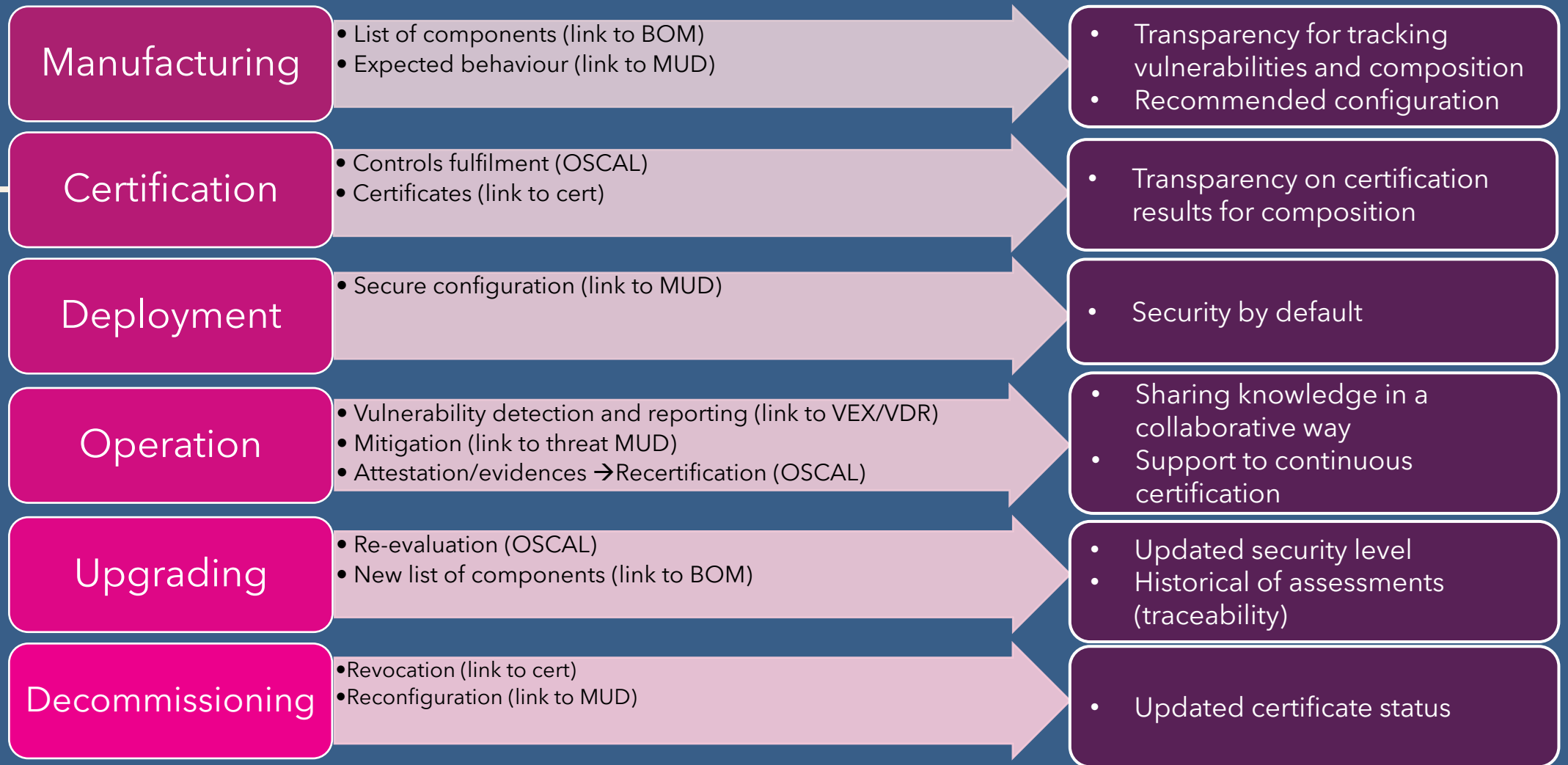
- designed, developed and produced to ensure an appropriate level of cybersecurity – Development, certification
- without any known exploitable vulnerabilities; - Development, certification
- secure by default configuration - Deployment
- appropriate exploitation mitigation mechanisms and techniques; - Operation
- provide security related information by monitoring relevant internal activity - Operation
- automatic updates to address vulnerabilities – Upgrading

RECAP

- **Lifecycle management**
  - How to coordinate all the exchanged information?
  - How to represent all the exchanged information?
  - How to support certification?
- **EU projects aiming to solve these questions : DOSS and COBALT**
  - Common model (DSP)
  - Based on OSCAL and linked with other formats

RECAP

OSCAL based DSP



https://dossproject.eu/

https://horizon-cobalt.eu/

THANKS

Dr. Sara N. Matheu

Dr. Antonio Skarmeta

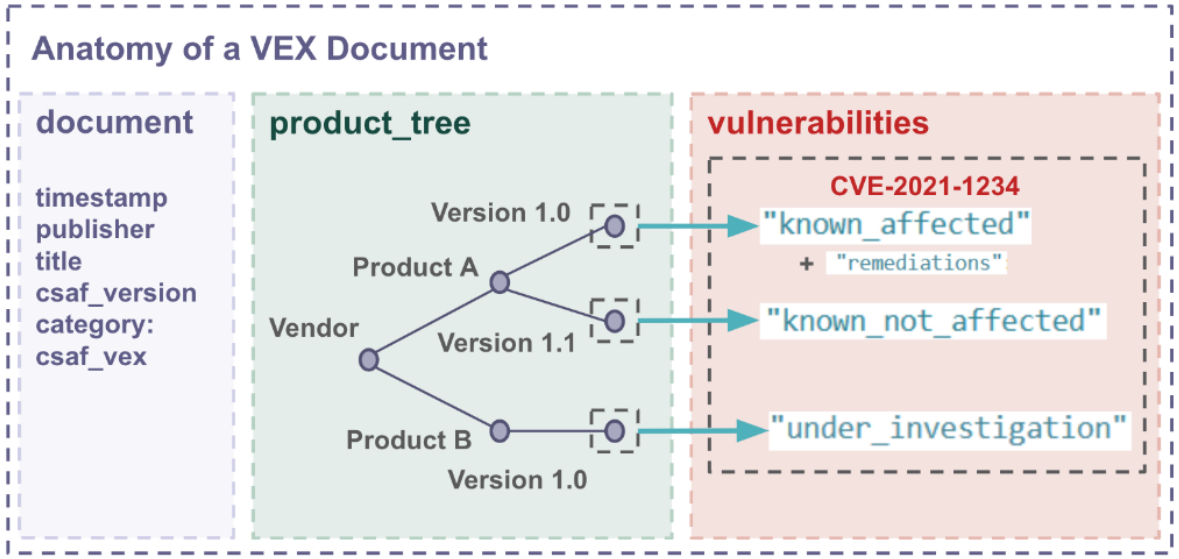
University of Murcia, Spain

VEX - CSAF

```
{
  "document": {
    "category": "csaf_vex",
    "csaf_version": "2.0",
    "notes": [
      {
    ],
    "publisher": {
    "title": "Example VEX Document Use Case 4",
    "tracking": {
  },
  "product_tree": {
  "vulnerabilities": [
    {
      "cve": "CVE-2021-44228",
      "notes": [
        {
          "category": "description",
          "text": "Apache Log4j2 2.0-beta9 through 2.15.0 (excludi
          "title": "CVE description"
        }
      ],
      "product_status": {
        "known_affected": [
          "CSAFPID-0002",
          "CSAFPID-0003",
          "CSAFPID-0004"
        ]
      },
      "references": [
        {
          "category": "external",
          "summary": "NVD - CVE-2021-44228",
          "url": "https://nvd.nist.gov/vuln/detail/CVE-2021-44228"
        }
      ],
      "remediations": [
        {
          "category": "vendor_fix",
          "details": "Update to version 4.2 or later.",
          "product_ids": [
            "CSAFPID-0002",
            "CSAFPID-0003",
            "CSAFPID-0004"
          ]
        }
      ],
      "scores": [
```

//Versions of the product (CSAFPID-0002 , etc)

```
      "scores": [
        {
          "cvss_v3": {
            "attackComplexity": "LOW",
            "attackVector": "NETWORK",
            "availabilityImpact": "HIGH",
            "baseScore": 10,
            "baseSeverity": "CRITICAL",
            "confidentialityImpact": "HIGH",
            "integrityImpact": "HIGH",
            "privilegesRequired": "NONE",
            "scope": "CHANGED",
            "userInteraction": "NONE",
            "vectorString": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H",
            "version": "3.1"
          },
          "products": [
            "CSAFPID-0002",
            "CSAFPID-0003",
            "CSAFPID-0004"
          ]
        }
```



https://github.com/oasis-tcs/csaf/blob/master/csaf_2.0/examples/csaf/csaf_vex/

https://github.com/openvex/spec

OpenVEX

- An OpenVEX document is composed of a JSON-LD structure that contains the document metadata and one or more VEX statements (more inline with 2023 VEX version).
- OpenVEX is designed to be SBOM format agnostic. It can reference software described in both SPDX and CycloneDX Software Bills of Materials.

```
{
  "@context": "https://openvex.dev/ns/v0.2.0",
  "@id": "https://openvex.dev/docs/example/vex-9fb3463de1b57",
  "author": "Wolfi J Inkinson",
  "role": "Document Creator",
  "timestamp": "2023-01-08T18:02:03.647787998-06:00",
  "version": "1",
  "statements": [
    {
      "vulnerability": {
        "name": "CVE-2014-123456"
      },
      "products": [
        {"@id": "pkg:apk/distro/git@2.39.0-r1?arch=armv7"},
        {"@id": "pkg:apk/distro/git@2.39.0-r1?arch=x86_64"}
      ],
      "status": "fixed"
    }
  ]
}
```

RECAP

- **Lifecycle management is a complex process involving many stakeholders and a big amount of dynamic data**. Some issues still open:
  - How to coordinate all the exchanged information?
  - How to represent all the exchanged information?
- **EU projects aiming to solve these questions : DOSS and COBALT**
  - Definition of a common model to represent the security information and definition of workflows for its management
  - Based on OSCAL and linked with other formats (MUD, BOM, VEX, VDR, etc)
- **Supporting EU regulation** (NIS, CSA, CRA)
- **Supporting certification** not as an isolated process after manufacturing, but as a process in line with the lifecycle management and the information exchanged throughout its lifecycle
  - Reusing information from previous assessments, supporting agile certification and providing trust and transparency
  - Involving all lifecycle stakeholders (manufacturers, domain owners, CA, users…) to gather as much information as possible
  - Establishing the secure configuration for deployment
  - Proactive security management throughout the device lifecycle by providing detailed information about vulnerabilities and mitigations