

# OpenEoX

- unified machine-readable approach to software and hardware product lifecycle data representation

Przemyslaw "Rogue" Roguski Principal Product Security Engineer



# What we'll discuss today

- What does product term stand for?
- Product identification and versioning
- Lifecycle? What is it?
- OpenEoX lifecycle data remedium
- Example approach
- OpenEoX challenges



#### What does product term stand for?



Product - can mean different things depending on the perspective of the usage or maintenance.

For one person something that is a component, for another can be a product.

It's OK as long as it's possible to precisely identify what we are talking about.



#### **Product identification**



But I need to also know product version...

\* tomorrow there will be a dedicated session regarding this: Challenges in Open Source Software Identification



#### **Product identification**

Product versioning interpretation can be problematic. Different products can use different versioning.

# Default X. Y. Z (Major.Minor.Patch) <u>Semantic Versioning</u> (SemVer):

 ${\bf X}$  - refers to the major version of the product, where usually an easy update between major versions is not possible

 ${\bf Y}$  - refers to the minor versions also called product streams, mostly used in products where many streams are supported at the same time

**Z** - refers to the asynchronous updates (patches) released as z-stream updates associated with a specific supported y-stream

# Example:

Red Hat OpenShift Container Platform release 4.18.5 major version stream update number 4 18 5

> :pe:/a:redhat:openshift:4.18::el8 :pe:/a:redhat:openshift:4.18::el9



# Lifecycle? Why should I care?

# In short:

Lifecycle is a combination of **product versioning** and **product support model**.

# Full definition:

Every product type (software, hardware, managed service and other deliverables) can be associated with a lifecycle model. It can contain definitions of various support models (different levels of maintenance) in association to the product versioning convention. The lifecycle support model is dynamic and changes over time, from the product's initial release (General Availability) to its discontinuation (End-of-Life).

During the product lifecycle, support models may switch from one state to another and may even run in parallel to meet individual requirements. Those requirements may depend on the product type, the vendor offerings, as well as geographical related regulations.



#### Product Lifecycle

Different lifecycle phases scope and important lifecycle points of time - examples

- **Full Support**: New features, bug fixes, security fixes, implementation support, and many more based on the vendor full support scope definition
- End\_of\_Maintenance: based on the vendor maintenance scope definition
- End\_of\_Feature: No new features are added
- End\_of\_Security\_Support: No security bugs are fixed
- End\_of\_Vulnerabity\_Support: No more vulnerabilities patches
- End\_of\_Bug\_Fixes: No more bugs are fixes
- End\_of\_Certification\_Support\_for\_Existing\_Products: No certification is done for existing products.
- End\_of\_Certification\_Support\_for\_new\_Products: No certification is done for new products.
- End\_of\_Documentation\_Support: No updates to documentation
- End\_of\_Support\_Channel\_Email: Can't contact support via email anymore
- End\_of\_Support\_Channel\_Telephone: Can't contact support via telephone anymore
- End\_of\_Support\_Channel\_Tickets: Can't contact support anymore by opening tickets
- End\_of\_Migration\_Support: No more support for scripts and tools that help to migrate to the product
- End\_of\_...
- End\_of\_Life: indicates the last day when product is officially supported in any way



## Product Lifecycle

Different lifecycle phases handling

Consecutive phases:





Different lifecycle phases

#### Overlapping phases:





#### OpenEoX

# OpenEoX - lifecycle data remedium?

The OASIS OpenEoX TC aims to standardize and promote OpenEoX, a unified machine-readable approach to managing and sharing End-of-Life (EOL) and End-of-Support (EOS) information for commercial and open source software and hardware.

OpenEoX is going to be compatible with other security related data like SBOM (Software Bill of Materials) or CSAF VEX (Common Security Advisory Format) to allow consumer easy usage of all data together to cover activities related to for example:

- Security Support
- Compliance Support
- Product Planning/Management
- Sales and Marketing planning



#### OpenEoX and SBOM and CSAF/VEX

#### Data is linked using **CPEs** and **purls**.





# Red Hat product lifecycle challenge - Red Hat OpenShift Container Platform Life example

Version	General availability	Full support ends	Maintenance support ends	Extended Update Support Add-On - Term 1 ends	Extended Update Support Add-On - Term 2 ends	Extended life phase ends
Full Support						
4.18	February 25, 2025	GA of 4.19 + 3 Months	August 25, 2026	February 25, 2027	February 25, 2028	N/A
4.17	October 1, 2024	May 25, 2025	April 1, 2026	N/A	N/A	N/A
Maintenance Support						
4.16	June 27, 2024	January 1, 2025	December 27, 2025	June 27, 2026	June 27, 2027	N/A
4.15	February 27, 2024	September 27, 2024	August 27, 2025	N/A	N/A	N/A
4.14	October 31, 2023	May 27, 2024	May 1, 2025	October 31, 2025	October 31, 2026	N/A
Extended Support						
4.12	January 17, 2023	August 17, 2023	July 17, 2024	January 17, 2025 <sup>13</sup>	January 17, 2026	N/A



category: product name **name**: Red Hat OpenShift Container Platform 4 product uuid: 670 product\_type: cloud businness unit: Cloud computing based on: Red Hat Enterprise Linux all phases: [ ... next slide... ] versions: version uuid: 412 product version: 4.12 product identification helper: cpe: cpe:/a:redhat:openshift:4.12::el8 previous version: 4.11 next version: 4.13 current phase: 4 phases: [...

phase\_id: 1 type: **General availability** date: 2023-01-17T00:00:00.000Z

phase\_id: 2 type: **End-of-Full-Support** date: 2023-08-17T00:00:00.000Z

phase\_id: 3 type: **End-of-Maintenance** date: 2024-07-17T00:00:00.000Z

phase\_id: 4 type: **End-of-Extended-Support** date: 2026-01-17T00:00:00.000Z

phase\_id: 5 type: **End-of-Life** date: 2026-01-17T00:00:00.000Z



# all\_phases: phase\_id: 1,

type: General availability, phase\_descriprion: The release GA date. display\_name: additional\_text:

# phase\_id: 2,

type: End-of-Full-Support, phase\_descriprion: Describes the last day of the Full Support and contains support scope, **scope**: [

```
new_features,
bug_fixes,
critical_security_patches, important_security_patches, moderate_security_patches
implementation_support
]
visibility: public
```



#### Challenges

# OpenEoX - challenges

- Universal taxonomy
- Single scope vs. multiple scope of the maintenance
- Date ranges vs. single date filed as end of some phase
- ...

# Red Hat OpenEoX adoption challenges

- Universal usage across all organizations
- Internal vs. External data
- Customization scope
- ...











in linkedin.com/company/red-hat





youtube.com/user/RedHatVideos

y twitter.com/RedHat

