Cyber Resilience Act, already too late to comply?

Alberto Pianon, Carlo Piana – Array

SFScon @Noi Techpark 2024

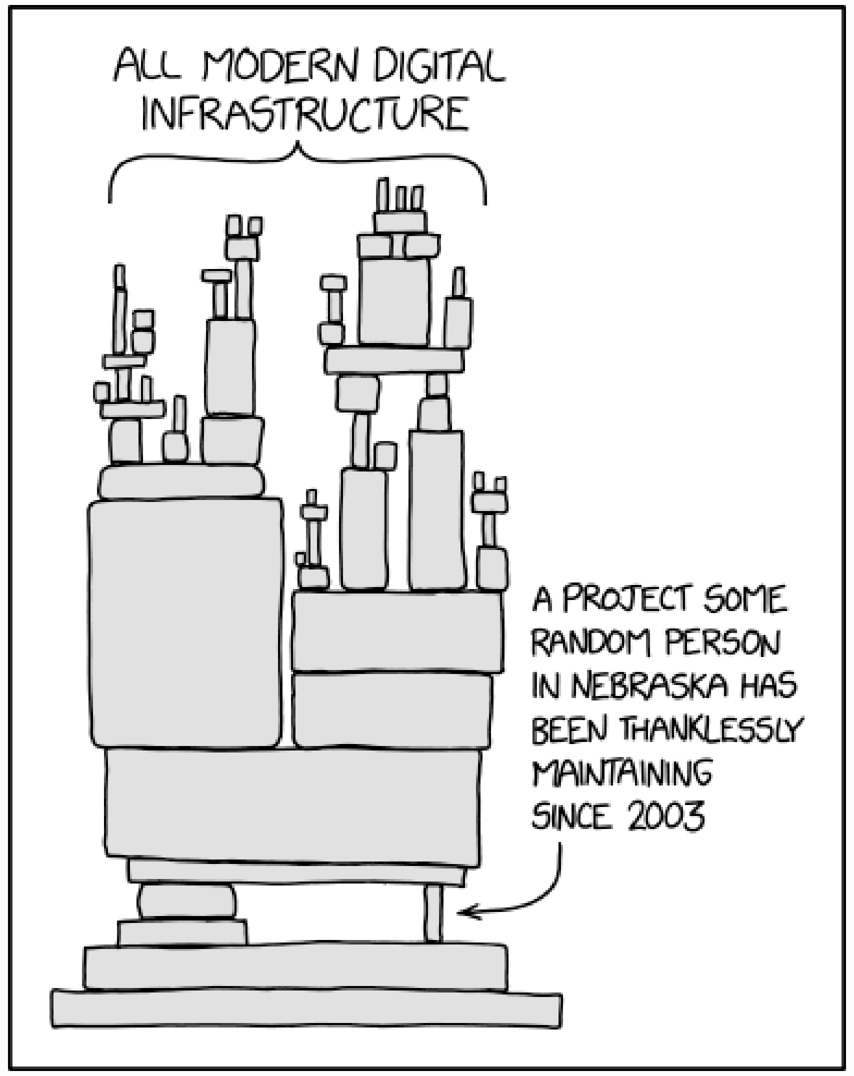
Back to 2003

```
if ((options == (__WCLONE|__WALL)) && (current->uid = 0))
   retval = -EINVAL;
```

- a line with an ~~innocent~~ “typo” was added to Linux 2.5 (dev)
- First(?) supply chain attack to Linux
- Promptly spotted and fixed before 2.6 (stable):
  “given enough eyeballs, all bugs are shallow”

more at: https://lwn.net/Articles/57135/

Fast forward to 2024


ALL MODERN DIGITAL INFRASTRUCTURE
A PROJECT SOME RANDOM PERSON IN NEBRASKA HAS BEEN THANKLESSLY MAINTAINING SINCE 2003

(C) XKCD, licensed under CC-BY-NC-2.5

- the attacker(s) targeted an under-maintaned, one-developer FOSS project, which is a dependency of a security-critical component (openssh->xz)
- through social engeneering, attacker(s) convinced the lonely maintainer that he needed help, and offered that help, gaining his trust
- the attacker was promoted to co-maintaniner, and injected malicious code in an xz release who ended up in all major linux distributions (Debian, Ubuntu, RedHat)
- malicious code injected in xz through a supply chain attack enabled a backdoor to execute arbitrary code with root privileges on any linux machine via ssh, without any valid credentials (!)
- only by sheer luck it was early discovered and fixed, and we avoided the Armageddon

more at https://arstechnica.com/security/2024/04/what-we-know-about-the-xz-utils-backdoor-that-almost-infected-the-world/

We have a sustainability problem!

(this is what ChatGPT thinks open source sustainability looks like)

And Finally, CRA Has Come

Cyber

Resilience

Act

- essential requirements (+ related obligations):
  - design, development, and production of [software]
  - vulnerability handling processes during the lifecycle of [software]
- conformity assessment: self-assessment, standards, certification, third-party assessment, depending on the level of risk (and [not] being FOSS)

In 36 months it will be in full force.

Who must comply

- manufacturers, importers, distributors of [Software]
- OSS projects?
  - ○ No, unless you are directly monetizing them (Recital 18)
  - ○ manufacturers integrating (monetizing) OSS -> due diligence on OSS components (Art. 13.5)
  - ○ voluntary assessment programs (possibly also initiated or financed by manufacturers: Recital 21, Art.25)
  - ○ manufacturers/OSS monetizers: obligation to share vulnerability fixes upstream (Art. 13.6)
  - ○ OSS Stewards supporting the development of FOSS (eg. foundations) ➔ limited obligations

Art. 2.48

> *‘free and open-source software’ means software the source code of which is openly shared and which is made available under a free and open-source licence which provides for all rights to make it freely accessible, usable, modifiable and redistributable*

Recital 18

the provision of products with digital elements qualifying as free and open-source software that are not monetised by their manufacturers should not be considered a commercial activity.

Furthermore, the supply of products with digital elements qualifying as free and open-source software components intended for integration by other manufacturers into their own products with digital elements should only be considered as making available on the market if the component is monetised by its original manufacturer.

Art. 13.5

manufacturers shall exercise due diligence when integrating components sourced from third parties so that those components do not compromise the cybersecurity of the product with digital elements, including when integrating components of free and open-source software that have not been made available on the market in the course of a commercial activity.

Recital 21

> *In order to support and facilitate the due diligence of manufacturers that integrate free and open-source software components that are not subject to the essential requirements set out in this Regulation into their products with digital elements, the Commission should be able to establish voluntary security attestation programmes, either by a delegated act supplementing this Regulation or by requesting a European cybersecurity certification scheme […].*
>
> *The security attestation programmes should be conceived in such a way that not only natural or legal persons developing or contributing to the development of a product with digital elements qualifying as free and open-source software can initiate or finance a security attestation but also third parties, such as manufacturers that integrate such products into their own products with digital elements, users, or Union and national public administrations.*

Art 2.14

‘open-source software steward’ means a legal person, other than a manufacturer, that has the purpose or objective of systematically providing support on a sustained basis for the development of specific products with digital elements, qualifying as free and open-source software and intended for commercial activities, and that ensures the viability of those products;

But!

- Someone's got to fix the unfixed!
- Yes, but who?

The >> 1M$ Question

Will CRA finally force OSS integrators/monetizers to do the same as the xz attackers, but for good, not for evil?

- identify OSS critical components thanklessly maintained by lonely volunteers with only two (tired) eyeballs (so bugs are not shallow)
- push them to accept help, and offer it:
  - fund voluntary assessment programs?
  - fix vulnerabilities and share fixes upstream?
  - fund maintenance?
  - in the Open Source way?

Takeaway Points

- How is too late and why?
- What is the way forward?
- Resources to ensure formal compliance, including due diligence obligations, are scarce and likely to be quickly exhausted.
- If the final integrator must also take care of most of the actual fixing of the upstream issues, that would be too much for many (and would be wasteful use of resources)
- Just waiting for the upstream to fix the unfixed is not an option
- A collectively proactive approach is required

Questions?

Thank You For Your Attention

https://array.eu

Array   Alberto Pianon   Carlo Piana

Presentation made using Reveal.js and a Markdown workflow with reveal-md
Soup image: Bonniebartilomo, CC BY-SA 4.0, via Wikimedia Commons