Scanning for known vulnerabilities in an embedded distribution

Marta Rybczynska
FOSDEM 2022
20 years in software development and Open Source
- Including 15 years in embedded

PhD in Telecommunications - on network security

Worked in embedded product development, then silicon...
- Now moved to distributions

Guest author at LWN

Contributing to Oniro from April 2021, consulting for OSTC
ABOUT ONIRO

○ **Source-based** distribution, using Yocto/OpenEmbedded

○ Aiming at **IoT space**, distributed operating system for consumer devices

○ For **products** -> with an **LTS** (Long Time Support)

○ Supporting **multi-OS** (Linux, Zephyr etc) and various device types

○ An **Eclipse project** announced in Nov 2021
WHY IS AUTOMATIC SCANNING IMPORTANT? (for us and for you)

• Oniro is a **distribution**, integrating packages from different sources
  - *Typical builds are around 300 packages*
  - So are product builds...
• IoT devices are often **online**
  - Potential attackers can access them
• New vulnerabilities show up **daily**
SECURITY DATABASES AND ABBREVIATIONS

○ **CVE (Common Vulnerabilities and Exposures)**
  - A database of unique IDs of vulnerabilities eg. CVE-2022-12345

○ **NVD (National Vulnerability Database)**
  - Contains CVE data with other information (which version, links to fixes, advisories...)

○ **CPE (Common Platform Enumeration)**
  - A standard for identifying software products. Used by the NVD
SECURITY DATABASE EXAMPLE (NVD)

https://nvd.nist.gov/vuln/detail/CVE-2021-3345
- CVE isn’t the only format
- **Vulnerabilities without CVEs exist**
  - You need to **apply** for a CVE
- **CPE mismatches** are frequent
  - Abbreviated vs non-abbreviated project name
  - Distro/product name only instead of the upstream
  - Backports just after the issue becomes public
TOOLS IN YOCTO/OPENEMBEDDED: cve-check

- Checks for every package in the NVD, gets a list of vulnerabilities
- Reports each vulnerability state (patched, unpatched...)
- Works using the package version
- Developers can mark added fixes, an issue that does not apply etc
- Lists a package only if it has (had) at least one vulnerability
• CVE list of Oniro **differs** from the official Yocto runs
  - Different layers, different choice of packages to build
• **Half of the packages** not on the list
  - No vulnerabilities?
  - CPE mismatch?
• About half of the vulnerabilities in the Linux kernel
  - Focus bias?
Tools in Yocto/OpenEmbedded: Additions

- CVE coverage pass
  - An extension of cve-check
  - Shows if there are CVEs for each CPE
  - Easier to find database mismatches

- Results
  - Mismatches: reported and fixed!
  - Review of packages without CVEs
EXAMPLE: PYTHON ISSUE CVE-2021-29921

- **Issue**
  - Original analysis: affects 3.8.0 to 3.9.5 (excluded)
  - But: a backport fix to 3.8.12
  - Sent and update to cpe_dictionary@nist.gov
  - Fixed!

- **Details**
  - [https://nvd.nist.gov/vuln/detail/CVE-2021-29921](https://nvd.nist.gov/vuln/detail/CVE-2021-29921)
EXAMPLE: LIBGCRYPT CVE-2021-33560, CVE-2021-40528

○ Issue
  - Two issues in one research paper
  - Misunderstanding on which issue gets which CVE: general confusion
  - Upstream commit referring to the other CVE’s description

○ Details
  - https://github.com/openembedded/openembedded-core/commit/0ce5c68933b52d2cfe9eea967d24d57ac82250c3
**FURTHER WORK/IDEAS FOR TOOLS**

- **Machine-readable format for issues list**
  - Cve-check format is text-based today
- **Detect copied-in code and report issues**
  - Example: libraries included in other projects
- **Scan upstream for fixes in stable branches**
- **All tools related to SBOM generation!**
TAKE-AWAYS

- You can check for **known issues in used packages**
  - Source form is “easy”
- **Manual verification** still needed
- Scanning is just the beginning
  - Finds **only what is public**
  - At Oniro we also do hardening and plan for more tools


○ **Websites:**
  - https://oniroproject.org/
  - https://projects.eclipse.org/projects/oniro

○ **Source code:**
  - https://booting.oniroproject.org/
Scanning for known vulnerabilities in an embedded distribution

Marta Rybczynska
FOSDEM 2022