Open Source CVE Monitoring and Management

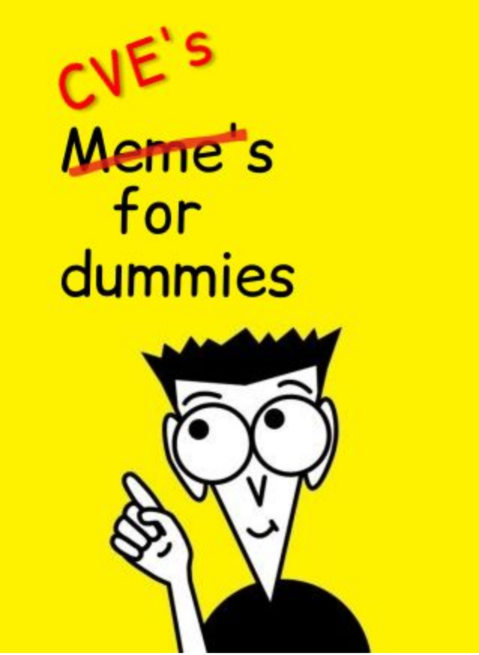
Presented by:

Akshay Bhat
Director of Engineering, Security Solutions

Embedded Linux Conference North America 2019
August 21, 2019

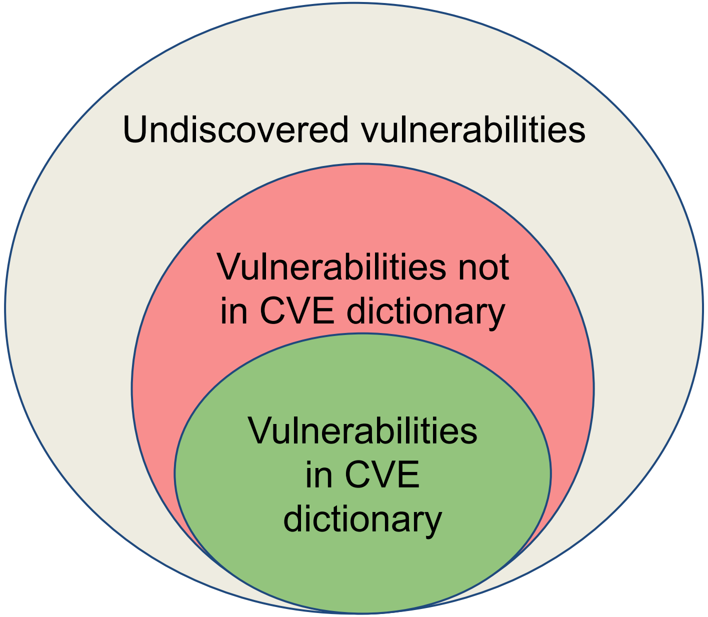
Agenda

- **Introduction to CVE**
  - Monitoring techniques
- **Prioritizing CVE**
- **Strategy for CVE fixes**
- **Quality of CVE data and tools**
- **Best practices, mitigation strategies**



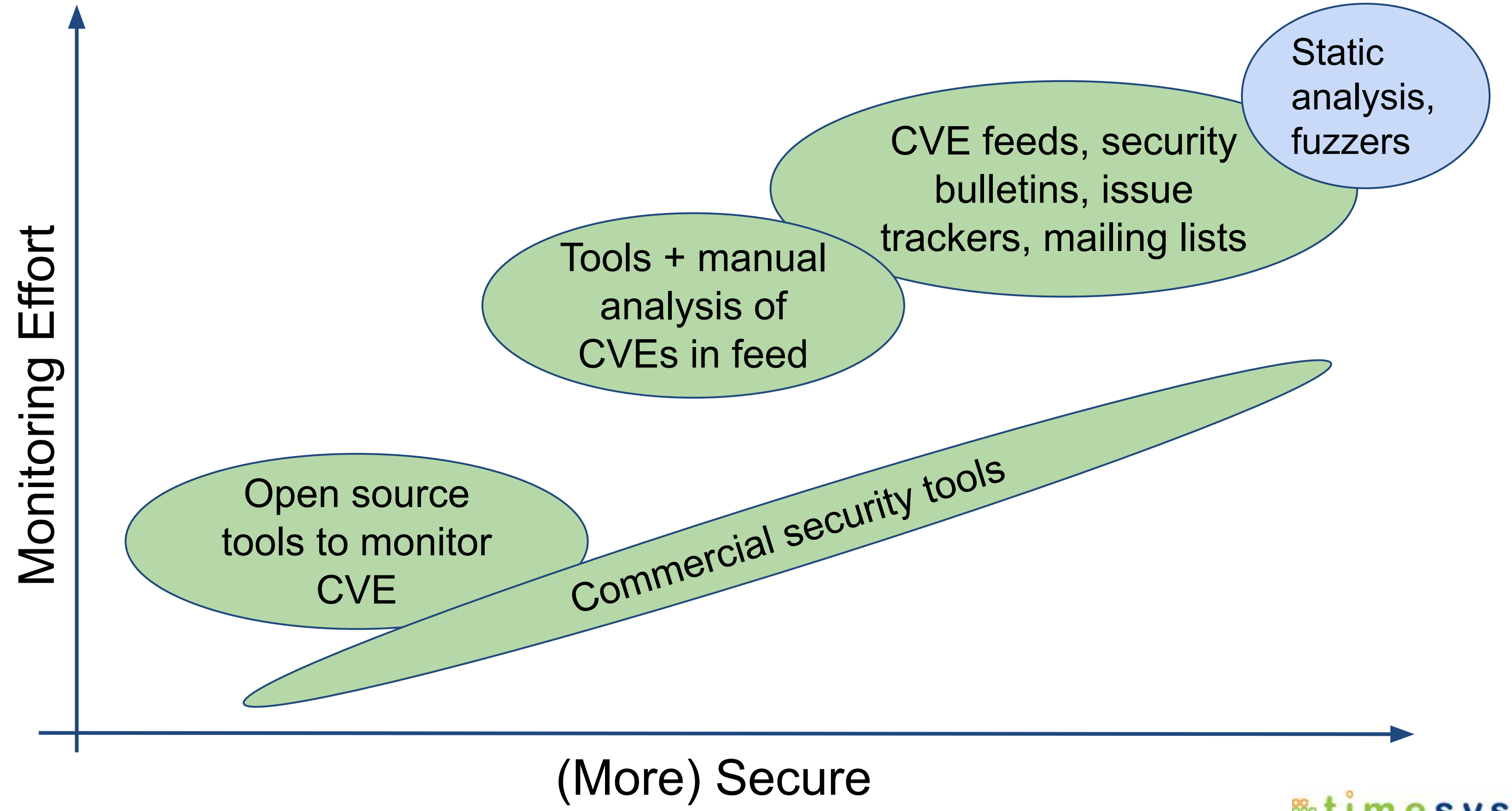
CVE what?

- **Common Vulnerabilities and Exposures**
  - List of entries of publicly known cybersecurity vulnerabilities
- **Does not cover silent "bug" fixes or undiscovered vulnerabilities**
- **Publicly available in the form of feeds**
  - Mitre
  - National Vulnerability Database (NVD)
    - Additional metadata


Undiscovered vulnerabilities
Vulnerabilities not in CVE dictionary
Vulnerabilities in CVE dictionary

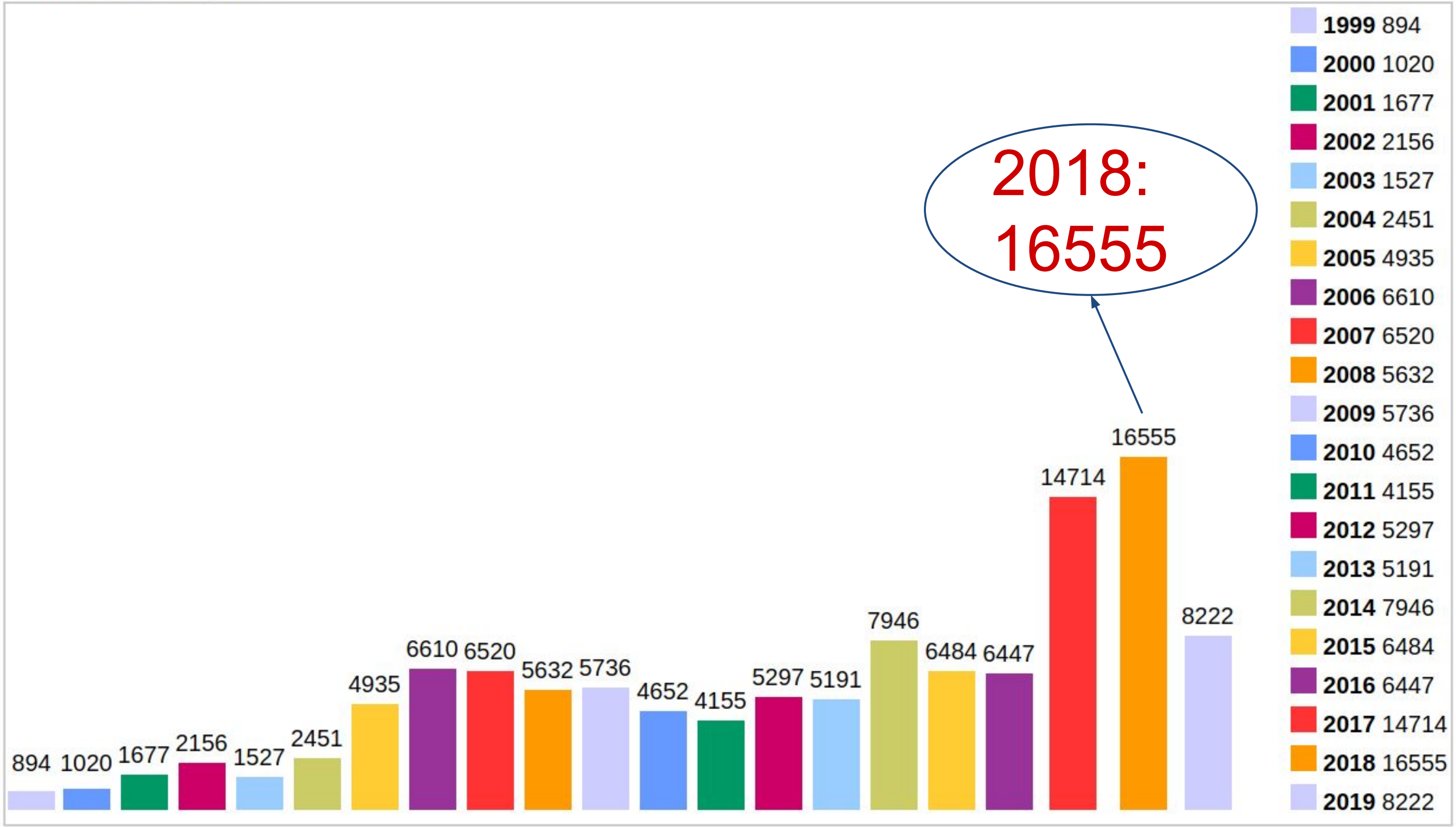
* not to scale

How much does security mean to you?


Monitoring Effort
(More) Secure
Open source tools to monitor CVE
Commercial security tools
Tools + manual analysis of CVEs in feed
CVE feeds, security bulletins, issue trackers, mailing lists
Static analysis, fuzzers

The CVE challenge — growing vulnerabilities

Vulnerabilities By Year



2018: 16555

Image source: cvedetails.com

CVE content

- **CVE-ID**
- **Description of the issue**
- **Estimated severity (CVSS - Common Vulnerability Scoring System)**
  - Low to Critical, 0.0 to 10.0
- **Estimated impact and domain scores**
  - e.g. “Attack Vector”, “User Interaction”, “Scope”, “Confidentiality”, …
- **Affected products, version numbers (CPEs - Common Platform Enumeration)**
  - eg: cpe:2.3:a:openssl:openssl:1.1.0g:*:*:*:*:*:*:*
    - Key piece for automation
- **List of reference links**
  - Exploits, patches, bug entry, mitigation, advisories...
- **Vulnerability Type (CWE - Common weakness enumeration)**
  - e.g. “buffer overflow”, “pointer issues”

Example: CVE-2018-18074

Current Description

The Requests package before 2.20.0 for Python sends an HTTP Authorization header to an http URI upon receiving a same-hostname https-to-http redirect, which makes it easier for remote attackers to discover credentials by sniffing the network.

Known Affected Software Configurations

cpe:2.3:a:python-requests:requests:*:*:*:*:*:*:*:*
Up to (excluding) 2.20.0

Impact

CVSS v3.0 Severity and Metrics:
Base Score: 9.8 CRITICAL

Vector:
AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 3.9

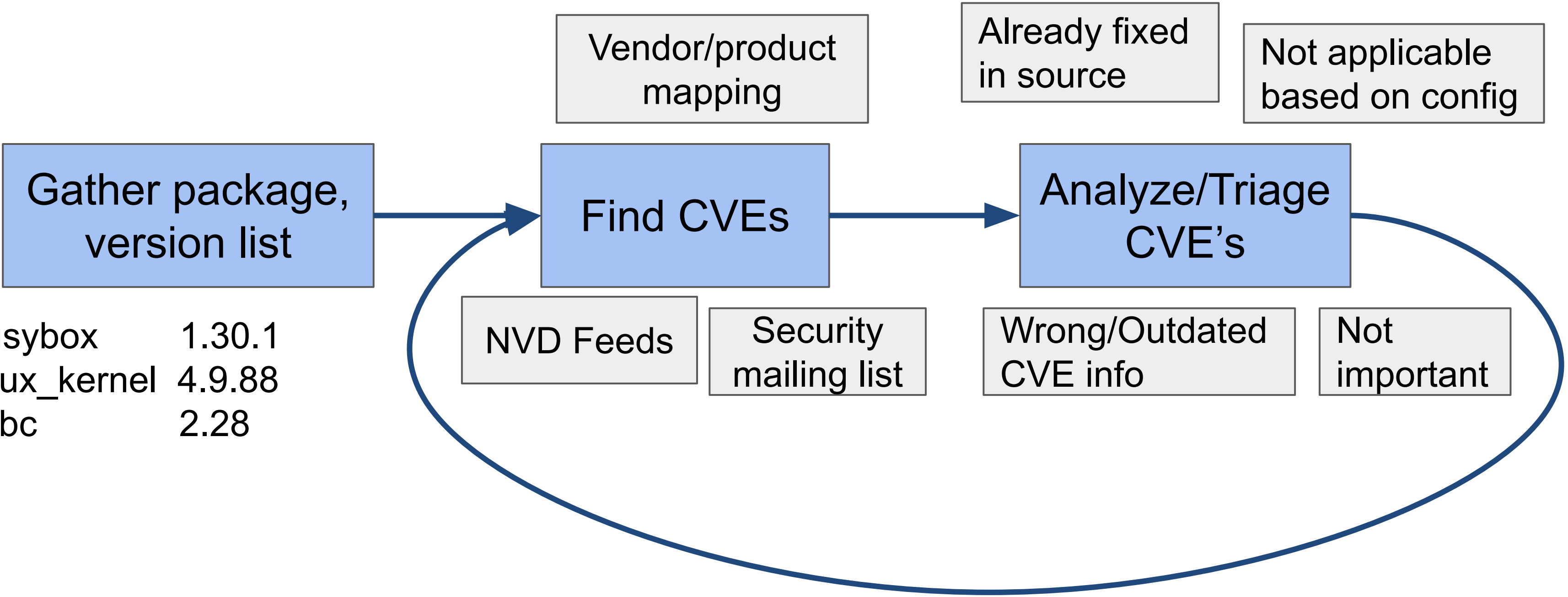
Attack Vector (AV): Network
Attack Complexity (AC): Low
Privileges Required (PR): None
User Interaction (UI): None
Scope (S): Unchanged
Confidentiality (C): High
Integrity (I): High
Availability (A): High

How to monitor CVEs? Linux Distro model

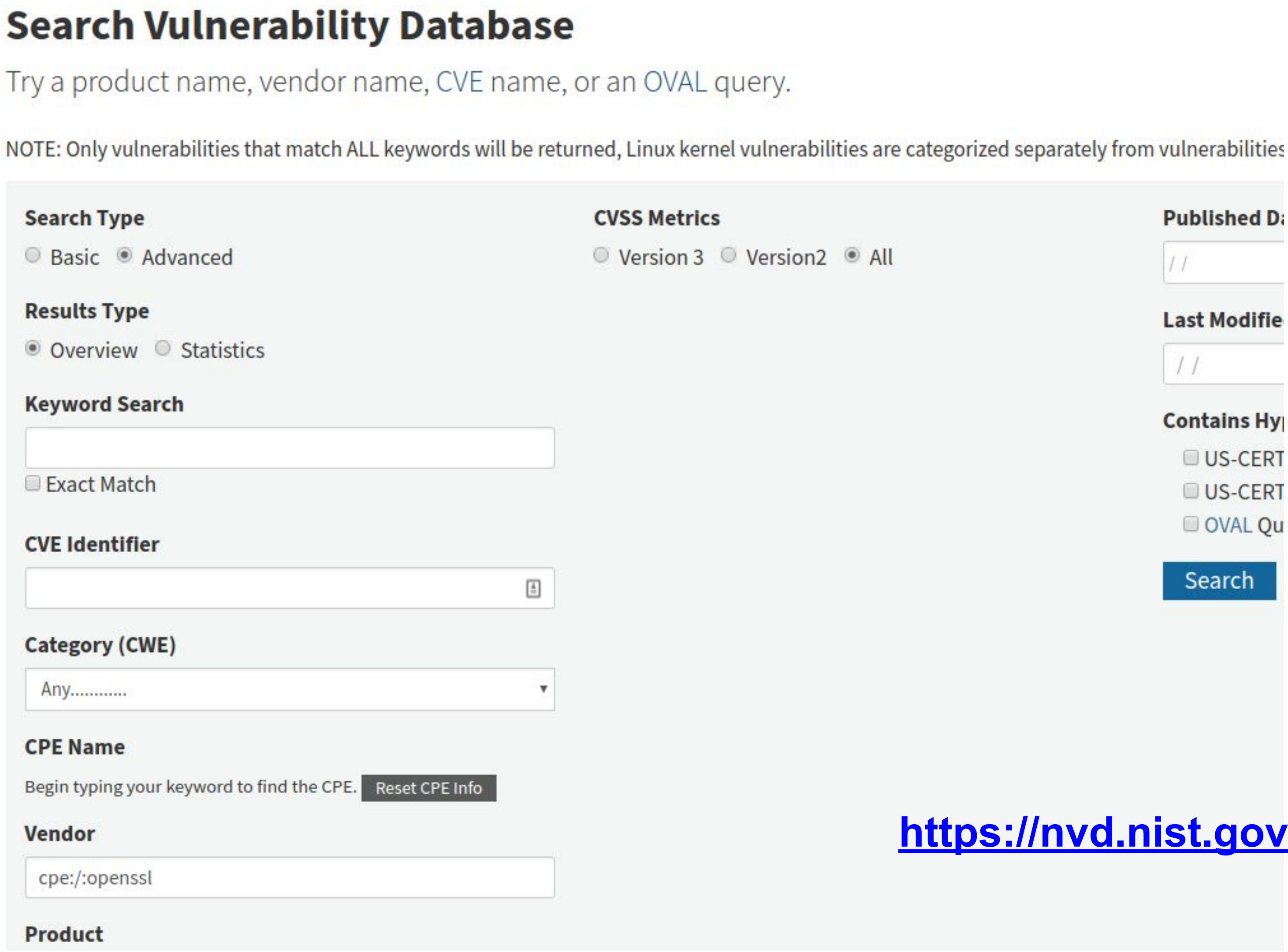
- **Follow what works for Ubuntu, Debian?**
- **Manually review each CVE from NVD feed (+ mailing lists + release notes, etc.)**
  - triage, tag
- **Monitor patches/new versions/re-analysis**
- **Issue security advisories**

Not practical for embedded developers delivering products!

DIY CVE monitoring



busybox	1.30.1
linux_kernel	4.9.88
glibc	2.28



Search Vulnerability Database

Try a product name, vendor name, CVE name, or an OVAL query.

NOTE: Only vulnerabilities that match ALL keywords will be returned, Linux kernel vulnerabilities are categorized separately from vulnerabilities

Search Type

○ Basic ● Advanced

Results Type

● Overview ○ Statistics

Keyword Search

□ Exact Match

CVE Identifier

Category (CWE)

Any............

CPE Name

Begin typing your keyword to find the CPE. Reset CPE Info

Vendor

cpe:/:openssl

Product

CVSS Metrics

○ Version 3 ○ Version2 ● All

Published Da

/ /

Last Modifie

/ /

Contains Hy

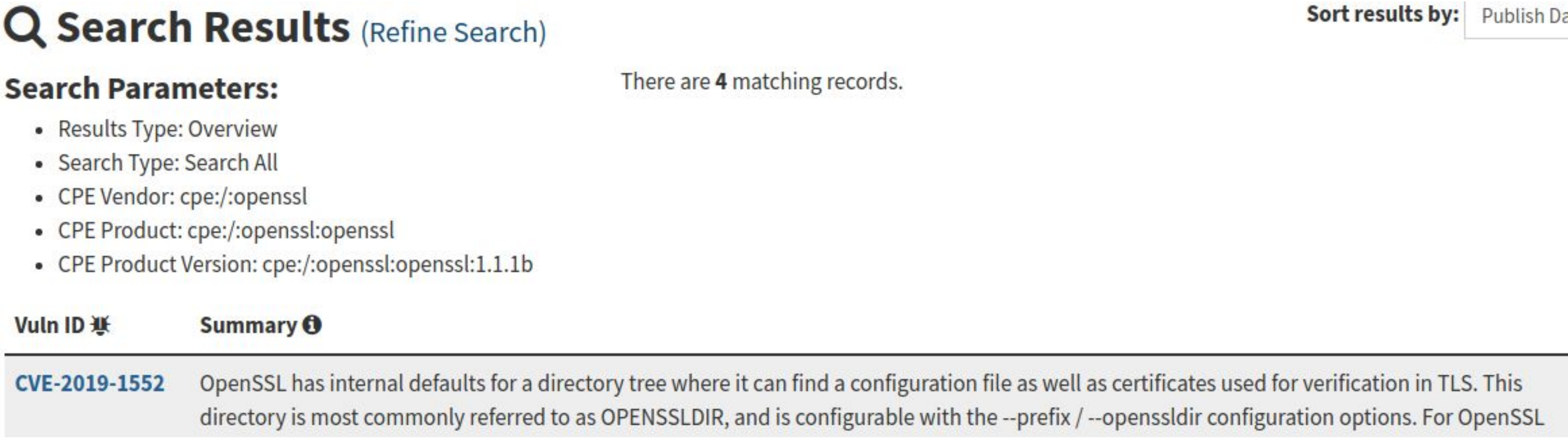
□ US-CERT

□ US-CERT

□ OVAL Qu

Search

https://nvd.nist.gov/vuln/search



Search Results (Refine Search)

Sort results by: Publish Da

Search Parameters:

- Results Type: Overview
- Search Type: Search All
- CPE Vendor: cpe:/:openssl
- CPE Product: cpe:/:openssl:openssl
- CPE Product Version: cpe:/:openssl:openssl:1.1.1b

There are **4** matching records.

Vuln ID	Summary
CVE-2019-1552	OpenSSL has internal defaults for a directory tree where it can find a configuration file as well as certificates used for verification in TLS. This directory is most commonly referred to as OPENSSLDIR, and is configurable with the --prefix / --openssldir configuration options. For OpenSSL

CVE monitoring in Yocto

Built-in support for automatic checking CVEs.

Add to conf/local.conf:

```
INHERIT += "cve-check"
```

Sample report:

```
PACKAGE NAME: linux-yocto
PACKAGE VERSION: 5.0.19+gitAUTOINC+c2e34d9ab2_00638cdd8f
CVE: CVE-2018-7754
CVE STATUS: Unpatched
CVE SUMMARY: The aoedisk_debugfs_show function in drivers/block/aoe/aoeblk.c..
CVSS v3 BASE SCORE: 5.5
VECTOR: LOCAL
MORE INFORMATION: https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2018-7754
```

Note: Contains host and target packages CVE; sifting is cumbersome

I have a CVE list, now what?

Prioritize based on filters:

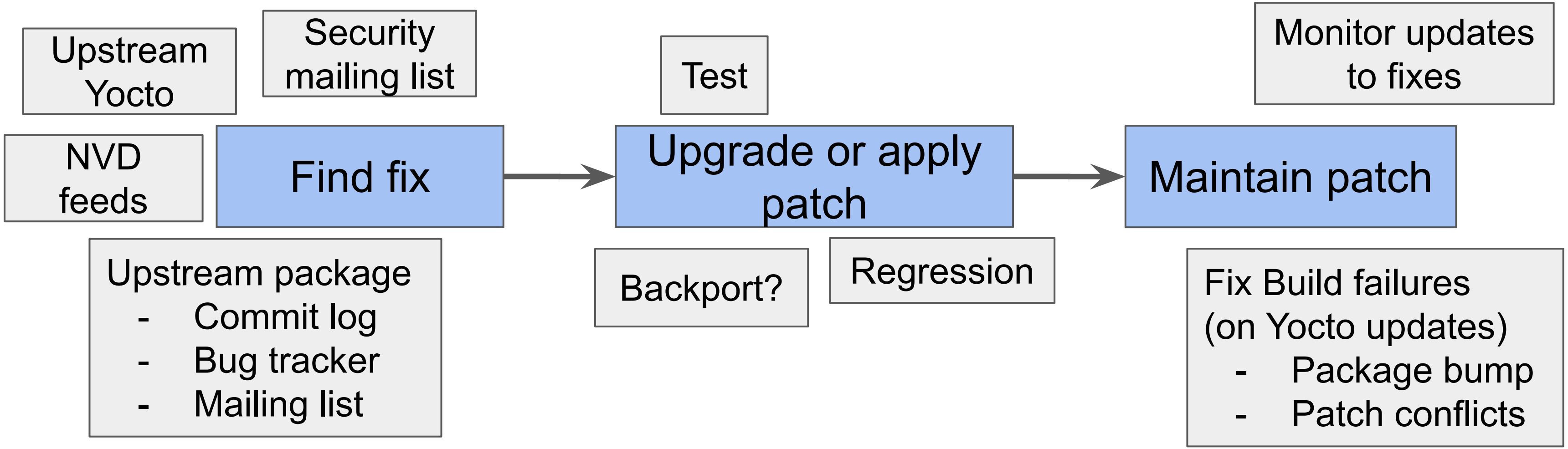
- **CVSS score**
  - Common Vulnerability Scoring System
  - Low, Medium, High, Critical
- **Attack Vector**
  - Network, Adjacent, Local, Physical
- **Exploit availability**
- **Patch/Mitigation availability**
- **Not applicable (eg: kernel config)**

Fix ASAP!

Filter type (incremental)	Unfixed CVE count
None	658 (incl. 339 kernel)
Kernel config	432
High/Critical CVSS	239
Network Attack vector	158
Public Exploits	33

Example CVE list based on a older NXP i.MX Rocko release.

DIY CVE Patching



Upstream Yocto

Security mailing list

NVD feeds

Find fix

Upstream package
- Commit log
- Bug tracker
- Mailing list

Test

Upgrade or apply patch

Backport?

Regression

Monitor updates to fixes

Maintain patch

Fix Build failures (on Yocto updates)
- Package bump
- Patch conflicts

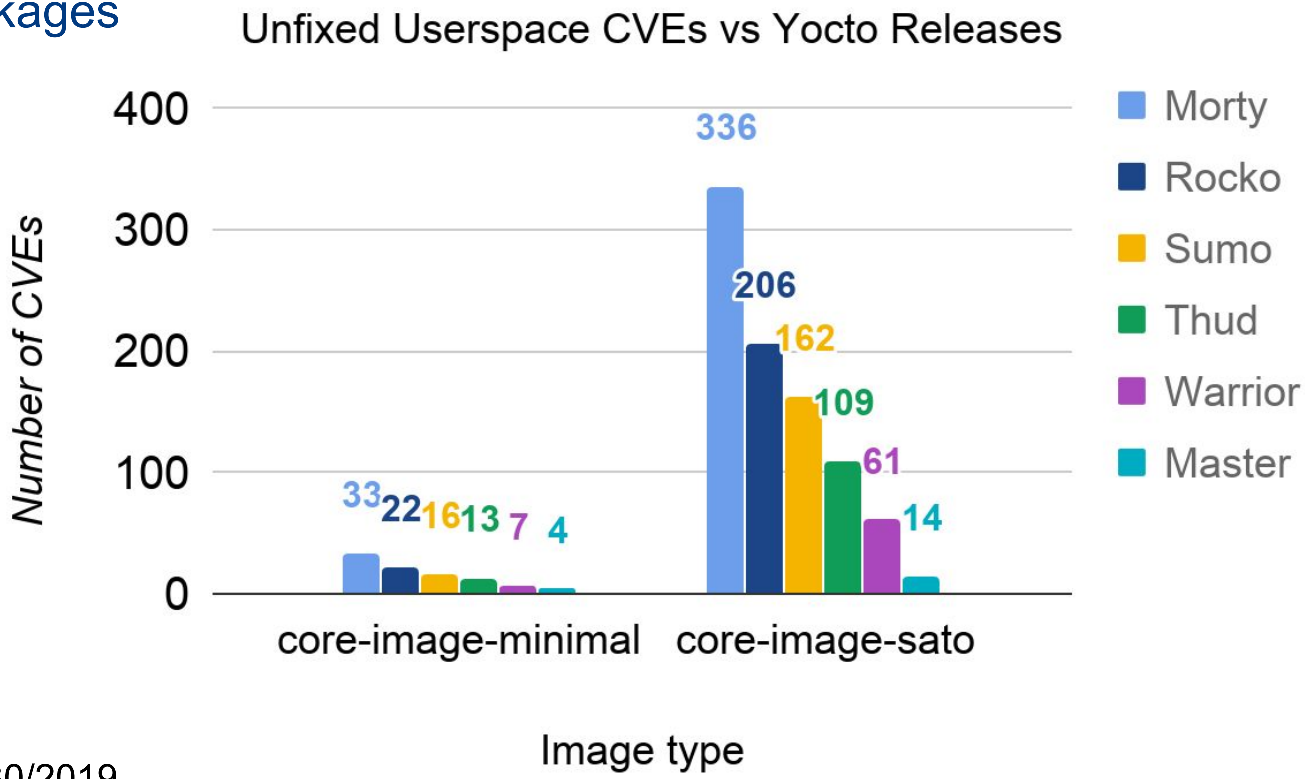
Upgrade vs. Backport

- **Upgrade**
  - API changes
  - License changes
- **Backport**
  - Complexity
- **Testing**
  - POC (proof of concept exploit)
  - Package tests (Yocto ptest)
- **Practicality**
  - Linux LTS kernel
    - 4.9.x kernel => ~1 release every 5 days! Product test cycles are longer than that!!
    - ~ 1-2 CVE fixes per release

Reasons to upgrade

Factors:

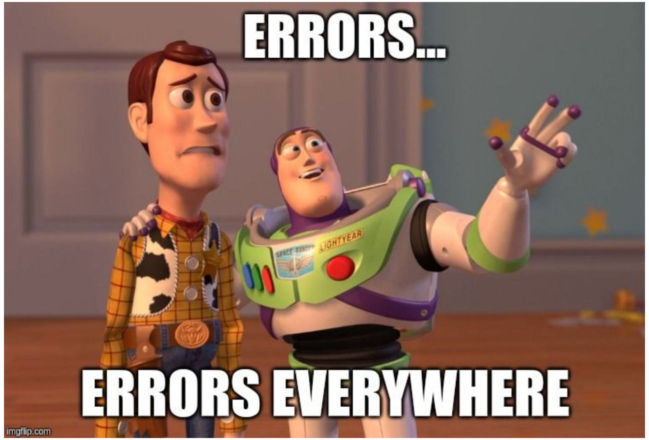
- Number of packages
- Release date



*approx numbers: as of 7/30/2019

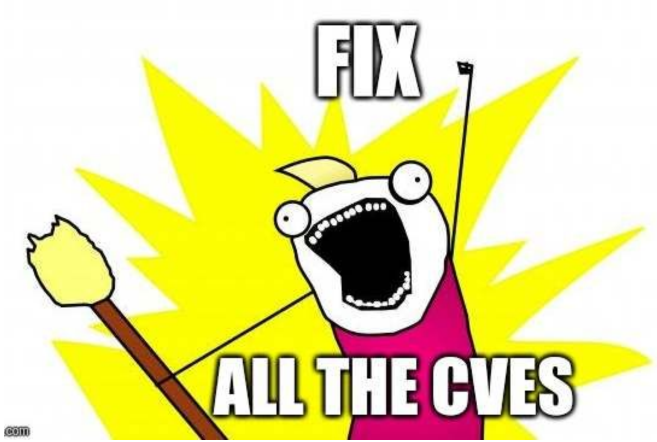
CVE data quality (False positives and misses)

- **Inconsistent naming**
  - arm-trusted-firmware, arm_trusted_firmware, trusted_firmware-a
- **Typos**
  - Version number
    - CVE-2016-1234: 2.2.3 instead of 2.23 (corrected now)
  - CVE product name
    - CVE-2016-1494: python instead of rsa (corrected now)
- **Incorrect/incomplete analysis**
  - CVE-2018-14618:
    up to 7.61.1 instead of 7.15.4 to 7.61.1
- **Outdated information**
  - Kernel CVEs (more later)
- **No version or cpe information**
  - CVE-2018-10845:
    cpe:2.3:a:gnu:gnutls:-:*:*:*:*:*:*:*



Yocto solutions

- **CVE_PRODUCT: recipe name to NVD name mapping**
  - `curl_7.65.3.bb: CVE_PRODUCT = "curl libcurl"`
  - `openssl_1.1.1c.bb: CVE_PRODUCT = "openssl:openssl"`
  - `python-urllib3.inc: CVE_PRODUCT = "urllib3"`
- **CVE_VERSION: recipe version to NVD version mapping**
  - `krb5_1.17.bb: CVE_VERSION = "5-${PV}"`
- **Tracks patched CVEs**
  - CVE ID in patch header (preferred)
  - CVE ID in file name



Yocto CVE report “bugs” YMMV

- **CVE_PRODUCT not specified in older releases**

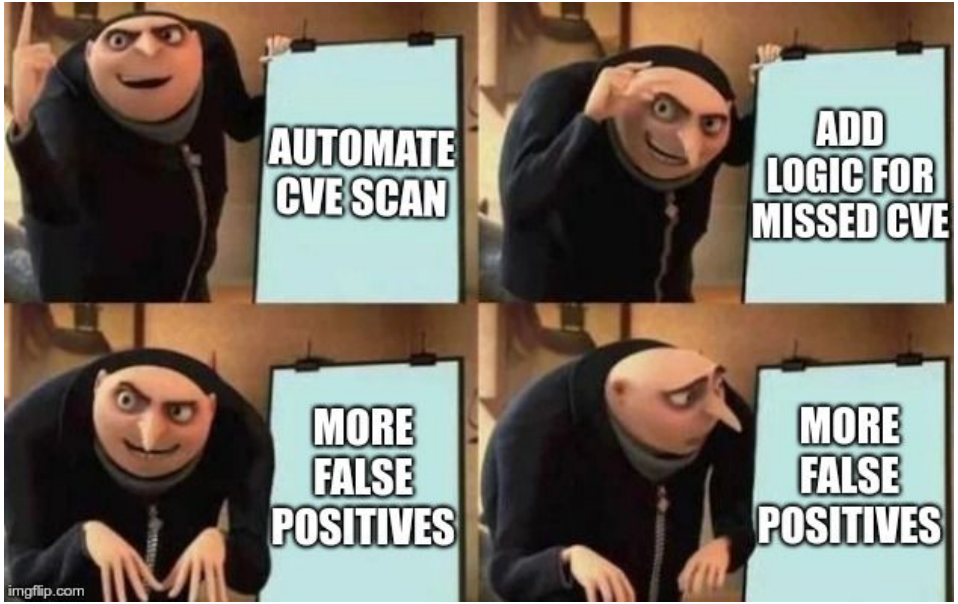
Release	**Missing CVE_PRODUCT (*relative to warrior)**	**Missed CVEs (*relative)**
morty	22	151 (96 High/Critical)
rocko	11	95 (75 High/Critical)
sumo	9	62 (44 High/Critical)
thud	7	21 (13 High/Critical)

*Tracking recipes included in poky with no other meta layers

Yocto CVE check improvements YMMV

- **cve-check-tool replaced by cve-update-db (JSON feeds)**
  - Master branch only! (link1, link2)
- **CVE result improvements**
  - cve-check-tool (string compare) vs. cve-update-db (>=, <= etc.)

Recipe	Rev	Previously missed
wpa-supplicant	2.6	3
python	3.5.5	2
sumo	2.30	5



If you see something, do something!

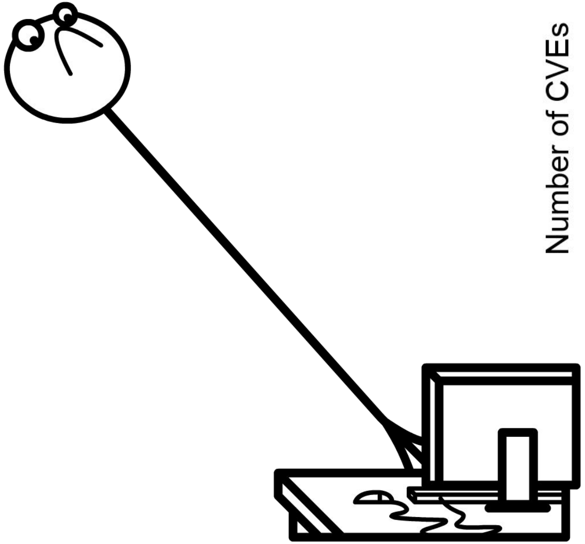
Don’t just fix it for you

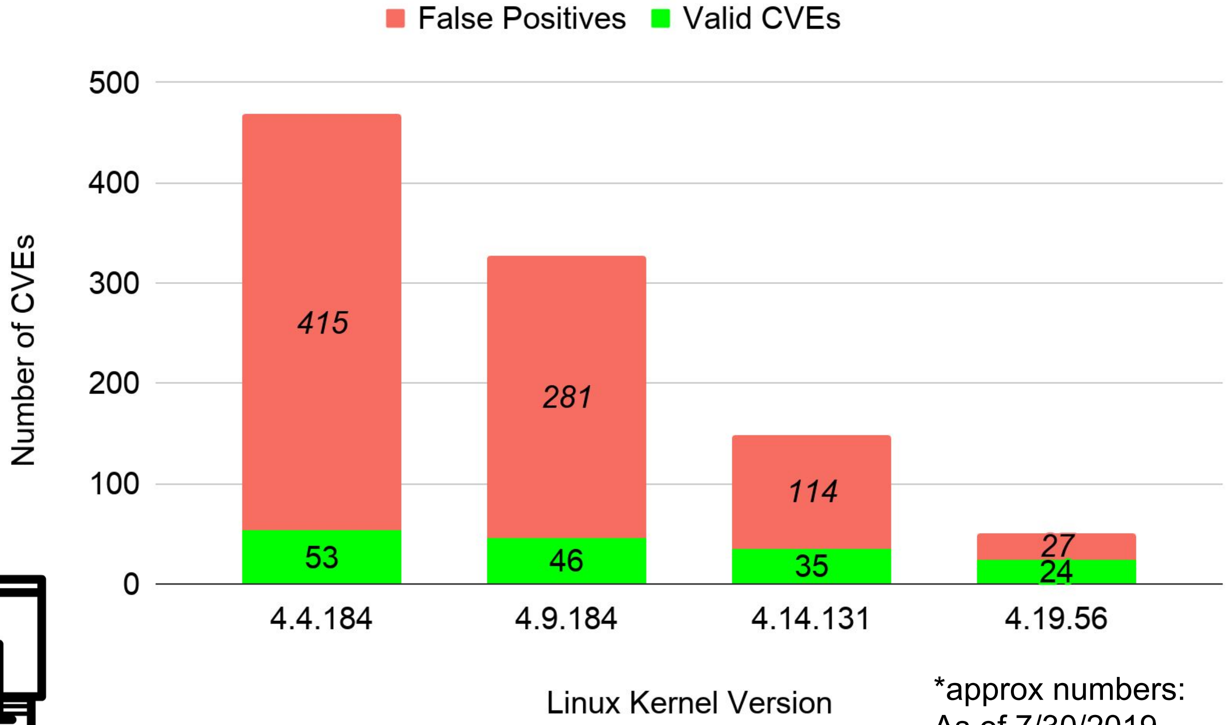
- CPE error: nvd@nist.gov
  - Error fixed and reflected within an hour!
- CVE summary/reference errors: https://cveform.mitre.org/
- Yocto – Missing CVE product:
  - Submit patch

#contribute!

Linux kernel CVEs

- **Typically, new CVE is listed as affecting all versions till latest**
- **Kernel maintainers do a fantastic job at backporting fixes to LTS**
  - NVD CPE info not updated when patches backported





*approx numbers:
As of 7/30/2019

Delays in CVE reporting / analysis

CVE-2019-6690 (python-gnupg)

1/19: Vulnerability discovered (private)

1/20: PoC created

1/22: Applied for CVE, vendor notified

1/23: CVE-2019-6690 assigned

1/23: Vendor responded, fix committed

1/25: Disclosed on oss-security (public)

3/21: NVD publishes CVE

4/2 : NVD analysis - adds cpe tags

68 days from being public to NVD analysis

CVE-2019-5436 (libcurl)

4/29: Reported on hackerone (private)

4/29: Fix developed (private)

5/15: Disclosed on distros list (private)

5/20: Fix appears on github

5/22: Disclosed on oss-security (public)

5/28: NVD publishes CVE

5/29: NVD analysis - adds cpe tags

7 days from being public to NVD analysis

Fun stats on delays

Year	NVD publish date to Initial analysis (average)	Redhat “public” date to NVD publish date (average)*
2017	11.6 days	101 days
2018	34.5 days	92 days
2019	10.4 days	25 days

*Notes:

- Redhat only tracks subset of products
- Sometimes CVE requested years after bug is reported and/or fixed!
  Example: CVE-2019-3901
  NVD publish date: 2019-04-22
  Patched in kernel: 2016-04-26

Leveraging work done by others!

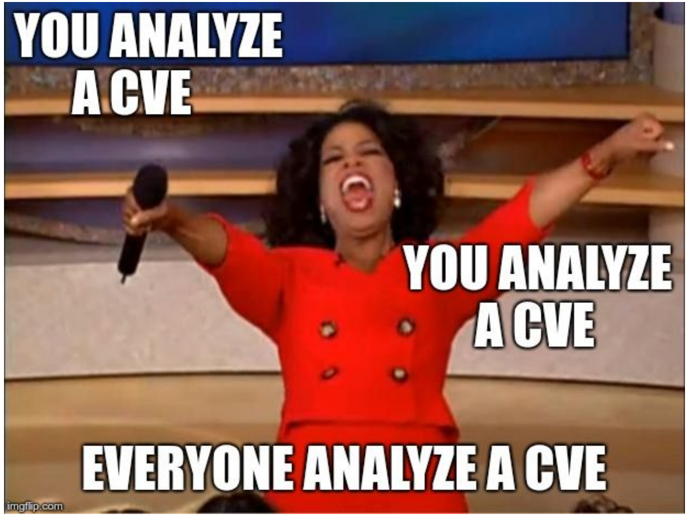
- **Debian tracker**
  - Tags: NOT-FOR-US, Minor issue, unimportant
    https://salsa.debian.org/security-tracker-team/security-tracker

- **Ubuntu tracker**

  -
    ```
    Introduced by: c7321cac2
    Fixed by     : 898471b92
    ```

  https://git.launchpad.net/ubuntu-cve-tracker/

- **CIP kernel CVE tracker**
  - Based on Ubuntu/Debian feeds
    https://github.com/cip-project/cip-kernel-sec



Secure boot and chain of trust

ROM	i.MX, Snapdragon SoC specific CVEs. eg: CVE-2017-7936
Second stage bootloader	Multiple CVEs based on bootloader
Arm trusted firmware	7 CVEs
3rd stage bootloader: u-boot	22 CVEs
OP-TEE	9 CVEs
Linux kernel	NaN ;)
User space	Openssl: 208 CVEs

SoC CVEs

- **Snapdragon 410 processor/firmware**
  - 246 CVEs (sd_410_firmware, sd410_firmware)
- **Intel CVEs**
  - converged_security_management_engine_firmware: 20
  - trusted_execution_engine_firmware: 13
  - active_management_technology: 6
  - core_i3: 14
  - manageability_engine_firmware: 5

Layered approach

- **Secure by design**
  - Hardware lockdown (serial console, jtag)
  - Secure boot, chain of trust
  - Secure storage and communications
  - Access control and hardening
  - Secure OS – OP-TEE / Arm TrustZone
  - Secure firmware update
  - Reduce attack surface
  - Security audit / pen testing
- **Stay secure**
  - Vulnerability monitoring and patching
  - Periodic upgrade
  - Audit log monitoring

Tools wishlist

- **Filters**
  - Kernel config based filtering
- **Workflow management**
  - Custom notes
- **Collaboration**
  - Team sharing
- **Report comparison**
  - New CVEs, History
- **Early notification**
  - Sources other than NVD
- **Patch notification**
  - Track fixes

Try: Vigiles (Free version available)

https://www.timesys.com/vigiles/

Take away

No magic bullet!

- **Design in security and firmware upgrades**
  - Reduce attack surface
- **Monitor vulnerabilities, triage, patch, update**
- **Be-aware of limitation of tools and NVD data**
  - Automate where possible
- **Contribute back to improve NVD data, tools**

Questions?

Visit us at: **Booth #23**

Thank you