Half-Day Vulnerabilities: A Study of the First Days of CVE Entries

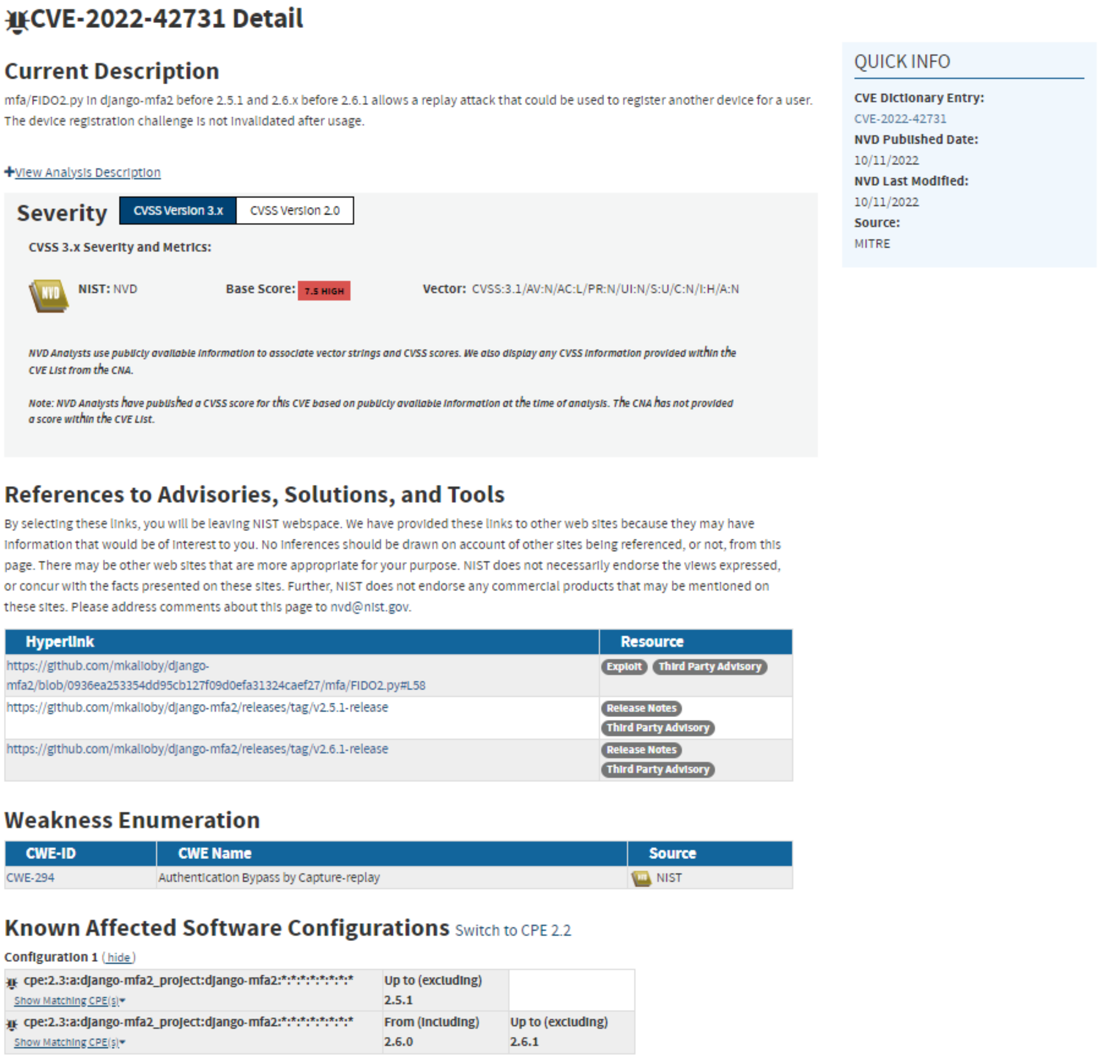
Raphaël Khoury
Kobra Khanmohammadi

UQO

Problem Statement

- The National Vulnerability Disclosure Database is an invaluable source of information for security professionals and researchers.
- Unfortunately, entries are often incomplete at the moment of publication, which hinders it’s use for vulnerability prioritization.
- We perform an empirical analysis of CVE entries that are initially published with an incomplete report.
- We present an novel ticketing system that addresses the problems related to such vulnerabilities .

CVE entry



CVE-2022-42731 Detail

Current Description

mfa/FIDO2.py in django-mfa2 before 2.5.1 and 2.6.x before 2.6.1 allows a replay attack that could be used to register another device for a user. The device registration challenge is not invalidated after usage.

+View Analysis Description

Severity

CVSS 3.x Severity and Metrics:

NIST: NVD **Base Score:** 7.5 HIGH **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:H/A:N

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

References to Advisories, Solutions, and Tools

By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sites because they may have information that would be of interest to you. No inferences should be drawn on account of other sites being referenced, or not, from this page. There may be other web sites that are more appropriate for your purpose. NIST does not necessarily endorse the views expressed, or concur with the facts presented on these sites. Further, NIST does not endorse any commercial products that may be mentioned on these sites. Please address comments about this page to nvd@nist.gov.

Hyperlink	Resource
https://github.com/mkalioby/django-mfa2/blob/0936ea253354dd95cb127f09d0efa31324caef27/mfa/FIDO2.py#L58	Exploit Third Party Advisory
https://github.com/mkalioby/django-mfa2/releases/tag/v2.5.1-release	Release Notes Third Party Advisory
https://github.com/mkalioby/django-mfa2/releases/tag/v2.6.1-release	Release Notes Third Party Advisory

Weakness Enumeration

CWE-ID	CWE Name	Source
CWE-294	Authentication Bypass by Capture-replay	NIST

Known Affected Software Configurations Switch to CPE 2.2

Configuration 1 (hide)

cpe:2.3:a:django-mfa2_project:django-mfa2:*:*:*:*:*:*:*:* Show Matching CPE(s)▾	**Up to (excluding) 2.5.1**	
cpe:2.3:a:django-mfa2_project:django-mfa2:*:*:*:*:*:*:*:* Show Matching CPE(s)▾	**From (including) 2.6.0**	**Up to (excluding) 2.6.1**

QUICK INFO

CVE Dictionary Entry:
CVE-2022-42731
NVD Published Date:
10/11/2022
NVD Last Modified:
10/11/2022
Source:
MITRE

Study Setup

- We downloaded the NVD everyday for 3 months, from June to August 2021.
- During this period, the NVD published 40,813 vulnerability reports, covering 14,896 distinct CVEs There were 25,917 updates.
- 846 reports were updates to CVEs initially published before June 2020, sometimes several years earlier.
- 403 entries did not have a CVSS v.3 score. We deleted these entries from our dataset.
- On average, an entry will be updated 4.7 times after the initial publication, but the number of updates varies widely and can be up to 17.

Empirical Analysis

RQ1: How many vulnerabilities are initially reported without a CVSS score each day?

- Makes it difficult to predict the severity of a vulnerability.
- 11 473 out of 40 813 (28%) vulnerability reports published during three months of study had no assigned CVSS base score. These reports represent 5270 out of 14 896 (35%) distinct vulnerabilities. The average number of vulnerabilities reported with no CVSS base score each day is 139.9

Empirical Analysis

RQ2: How long after the CVE is initially published until the CVSS score is finally reported?

- Out of 5270 CVE entries for which no CVSS score was initially provided, 3612 (69%) were eventually updated with a CVSS v.3 base score.
- An additional 6% received an update that did not contain a CVSS score.
- The balance (25%) were never updated.
- On average, the CVSS score is included 11.6 days after publication.

Empirical Analysis

RQ3: How many vulnerabilities (CVEs) are not initially assigned a CPE list?

- The CPE list makes it easy to identify software that is affected by the vulnerability.
- During the period of our study, 7748 out of 14,896 (52%) vulnerabilities were initially reported without a CPE list.
- An average of 133.7 vulnerabilities each day.

Empirical Analysis

RQ4: How long after the CVE is initially published until the related CPE list is finally reported?

- Of the 5128 CVEs that are published without a CPE list, 2649 (51.65%) were eventually updated to include this information.
- An average of 11.5 days elapse from publication to the inclusion of the CPE.
- An additional 5% did receive an updated, but this update did not include the CPE.

Empirical Analysis

RQ5: How many vulnerabilities have no proposed mitigation approaches, including update or workaround?

Forces a difficult choice between running a vulnerable software and foregoing use of a tool.

- An average of 894 out of 40,813 (2%) vulnerabilities were initially reported with no mitigation included in the report.

Empirical Analysis

RQ6: Are there manufacturers (CPE) that are more likely to report a vulnerability without a CVSS score and\or a mitigation?

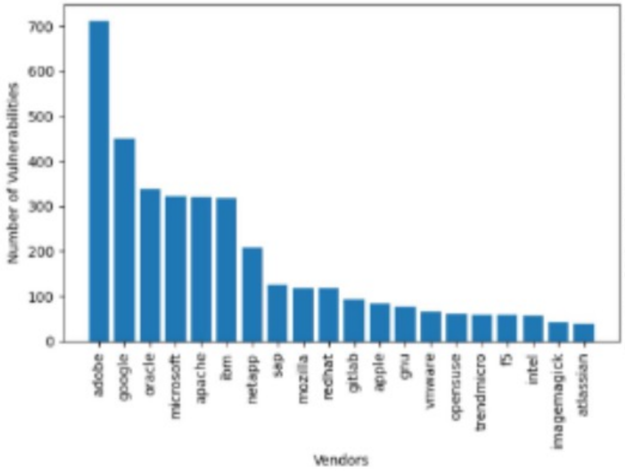
Could be a valuable differentiating factor.

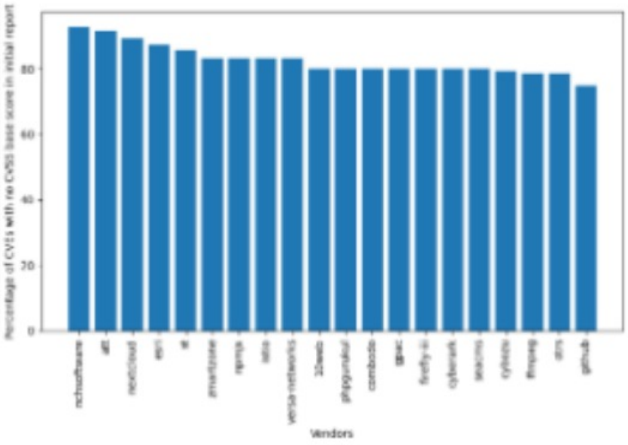
- We extracted the top top 20 vendors with the highest percentage of vulnerabilities initially reported with no CVSS score, as well as the top 20 vendors with the highest percentage of CVEs submitted with a CVSS score from the onset.

Empirical Analysis

RQ6: Are there manufacturers (CPE) that are more likely to report a vulnerability without a CVSS rating and\or a mitigation?

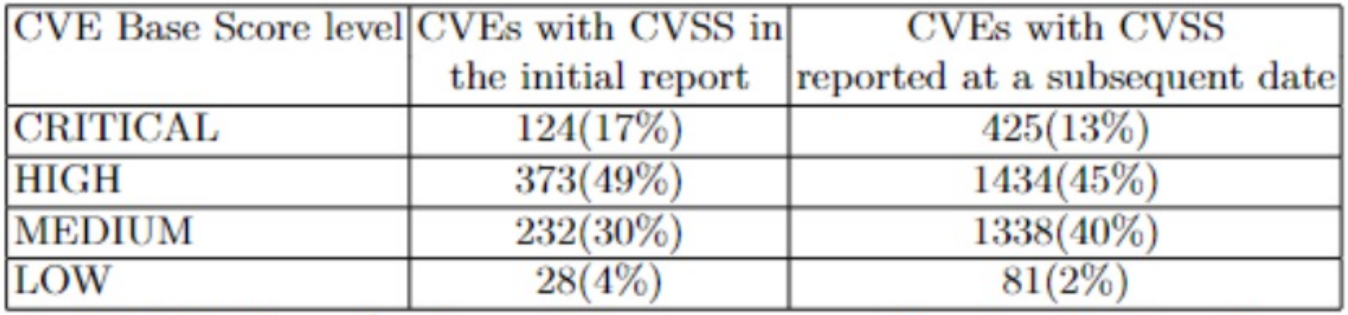
- Across all vendors, the average percentage of vuln. without a CVSS score is 6.6%. For the top vendors its is 82%.





Empirical Analysis

RQ7:Is there a statistically significant difference in CVSS score values between vulnerabilities that are initially reported without a CVSS score and those that are?



CVE Base Score level	CVEs with CVSS in the initial report	CVEs with CVSS reported at a subsequent date
CRITICAL	124(17%)	425(13%)
HIGH	373(49%)	1434(45%)
MEDIUM	232(30%)	1338(40%)
LOW	28(4%)	81(2%)

Table 1: Distribution of vulnerability score according to whether the score is initially present or not.

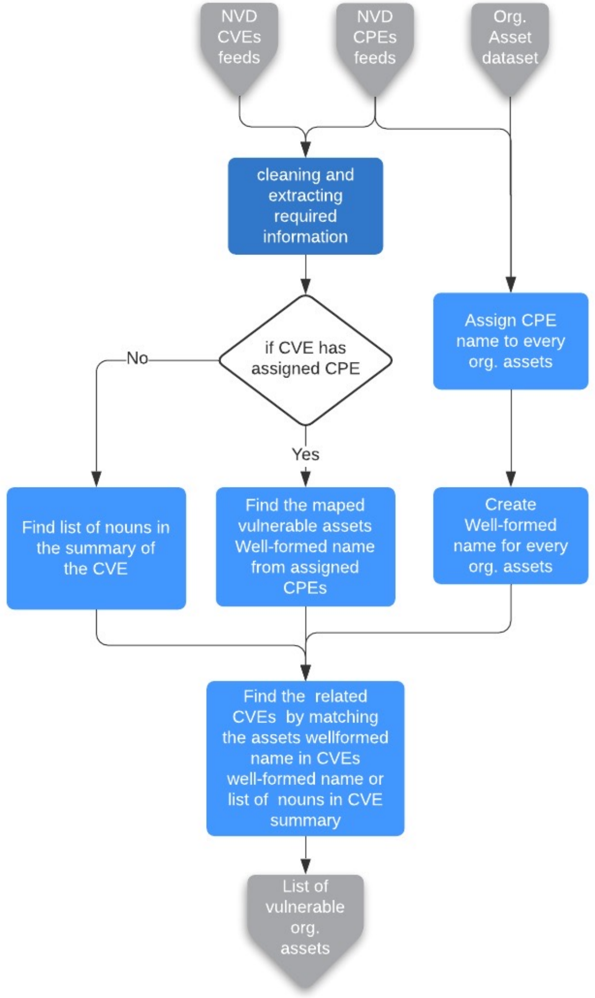
- A Wilcoxon-Mann-Whitney test shows no statiscally significat divergence.

Empirical Analysis: Key Findings

- It is surprisingly common for CVE entries to be published with key information missing, notably the CVSS score (35%), the CPE (52%) and the mitigation (2%).
- The information is often missing for several days, on average 11.6 days in the case of CVSS and 11.5 days in the case of the CPE.
- Only 2% of vulnerabilities are not assigned a mitigation.
- Vulnerabilities with missing information do not differ from those for which all data is provided at the onset.

CVE Matching System

- We developed a new CVE matching system that palliates the absence of a CPE list in the CVE entry.
- Takes as input the (1) feed from NVD, (2) the CPE dictionary and a (3) company asset list.
- If the CPE is present, matching can be done using the CPE.
- Otherwise, the matching syst. scans the vulnerability description to determine the affected software.



NVD CVEs feeds

NVD CPEs feeds

Org. Asset dataset

cleaning and extracting required information

Assign CPE name to every org. assets

if CVE has assigned CPE

No

Yes

Find list of nouns in the summary of the CVE

Find the maped vulnerable assets Well-formed name from assigned CPEs

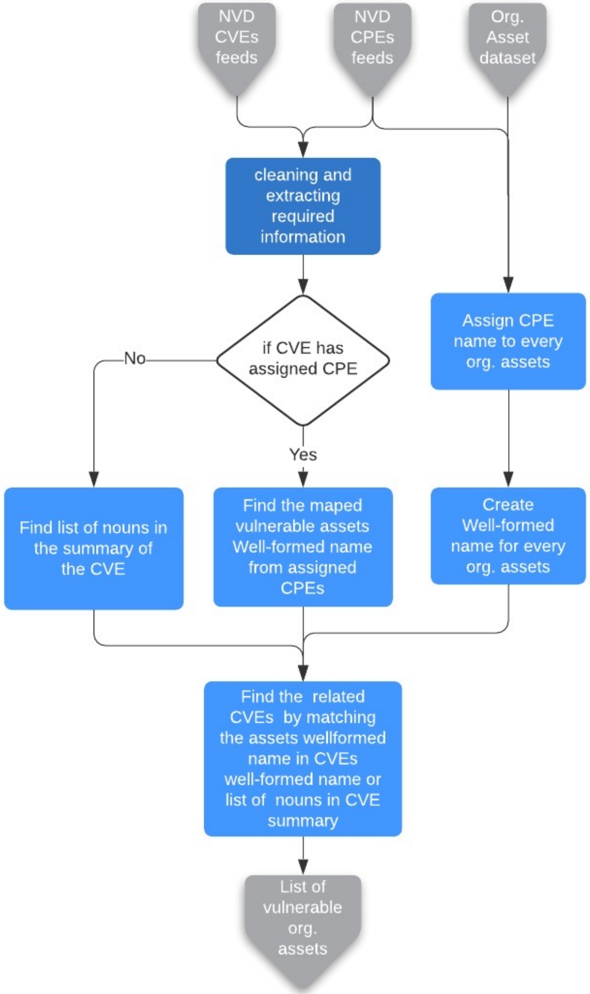
Create Well-formed name for every org. assets

Find the related CVEs by matching the assets wellformed name in CVEs well-formed name or list of nouns in CVE summary

List of vulnerable org. assets

CVE Matching System

- We introduce the notion of a **well formed named (WFN)**.
- The WFN is a formatted string in the format name:vendor:version, that can replace the CPE if it is absent.
- Well formed named are created from the asset list be deleting special characters, common words and numbers.
- R2D2 Beta version 3.0.1.16 becomes : r2d2:Geotab:3.0.1.16


NVD CVEs feeds
NVD CPEs feeds
Org. Asset dataset
cleaning and extracting required information
if CVE has assigned CPE
No
Yes
Assign CPE name to every org. assets
Find list of nouns in the summary of the CVE
Find the maped vulnerable assets Well-formed name from assigned CPEs
Create Well-formed name for every org. assets
Find the related CVEs by matching the assets wellformed name in CVEs well-formed name or list of nouns in CVE summary
List of vulnerable org. assets

CVE Matching System

- We then extract a list of nouns from the summary description of the CVE. All nouns are stemmed and common words are removed.
- The resulting list of stemmed words is then checked against the list of well-formed names to look for matches.
- False positives are possible in case the name a well-formed name contains a common word, ex. SQL server generates a false positive on CVEs that contain the word “SQL injection”.

CVE Matching System

- We then extract a list of nouns from the summary description of the CVE. All nouns are stemmed and common words are removed.
- The resulting list of stemmed words is then checked against the list of well-formed names to look for matches.
- False positives are possible in case the name a well-formed name contains a common word, ex. SQL server generates a false positive on CVEs that contain the word “SQL injection”.

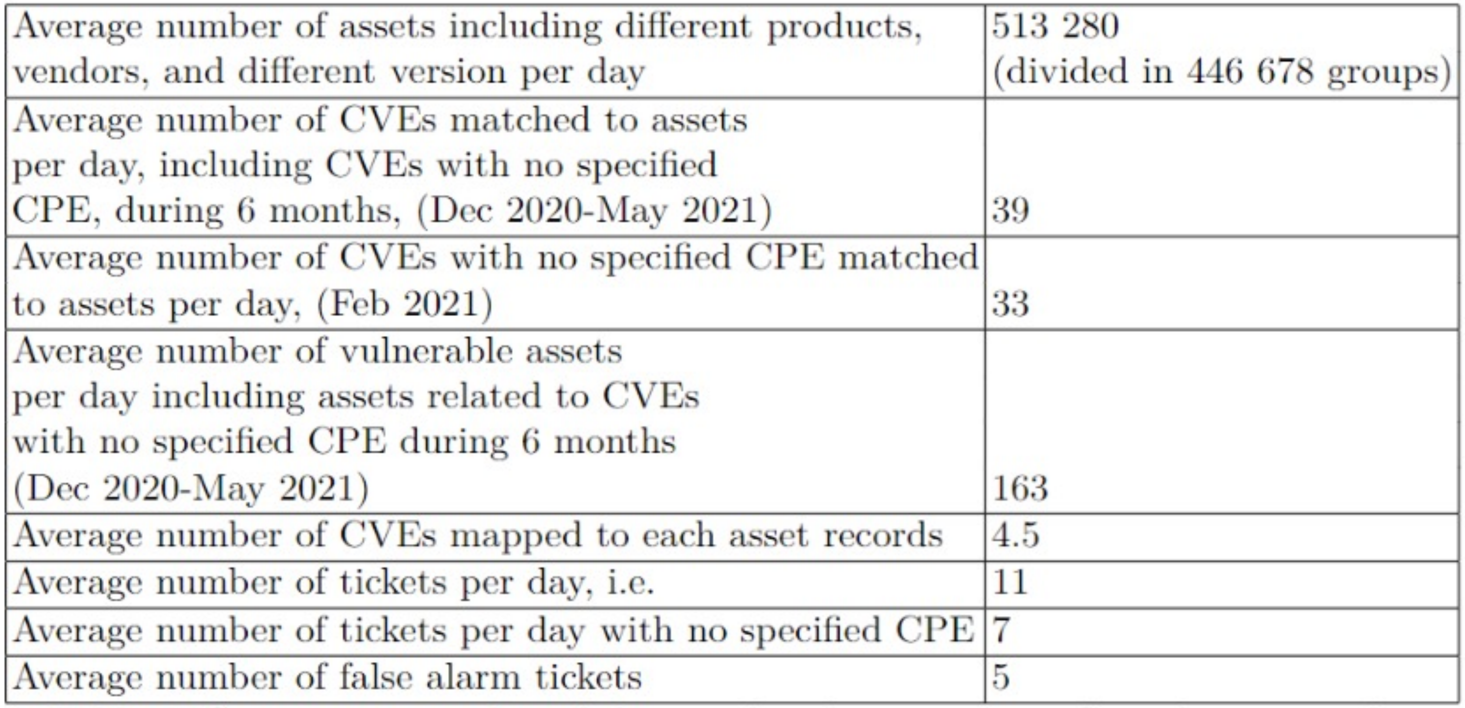
CVE Matching System

- Identifying each vulnerability in a company asset and reporting it in a separate ticket is inadequate: it could lead to an large number of tickets.
- Multiple vulnerabilities may relate to the same software, and have a common mitigation: usually applying a patch.
- The CVE matching system thus groups vulnerabilities that relate to the same software in a common ticket.

Case Study

- We implemented this framework at Geotab, a fleet tracking firm based in Toronto.
- The CVE matching framework was used for 6 months, from December 2020 to may 2021.
- Geotab's asset list consists in over 500 000 entries. When grouped by vendor and products (ignoring versions), there are 446 678 entries.
- Each day, an average 39 groups (163 software) have at least 1 vulnerability.
- If an asset contains vulnerabilities, an average 4.5 CVEs relate to that asset. The CVE matching system groups them in a single ticket.

Case Study



Average number of assets including different products, vendors, and different version per day	513 280 (divided in 446 678 groups)
Average number of CVEs matched to assets per day, including CVEs with no specified CPE, during 6 months, (Dec 2020-May 2021)	39
Average number of CVEs with no specified CPE matched to assets per day, (Feb 2021)	33
Average number of vulnerable assets per day including assets related to CVEs with no specified CPE during 6 months (Dec 2020-May 2021)	163
Average number of CVEs mapped to each asset records	4.5
Average number of tickets per day, i.e.	11
Average number of tickets per day with no specified CPE	7
Average number of false alarm tickets	5

Table 2: Statistics on vulnerabilities related to assets in Geotab case study

Future Work

- Integrate existing projects that predict other missing information (CVSS score, CWE) from the data present in a CVE report.
- Predict severity and exploitability.

Thank You

Questions?