Understanding your vulnerability data to optimize your DevOps pipeline flow

27 May 2023

https://www.bsidesdub.ie/

About Me

Chris Madden

Yahoo Paranoids Product Security Engineer

Chris has worked as a software engineer and system architect building secure trustworthy software at scale for embedded and cloud for more than 20 years.

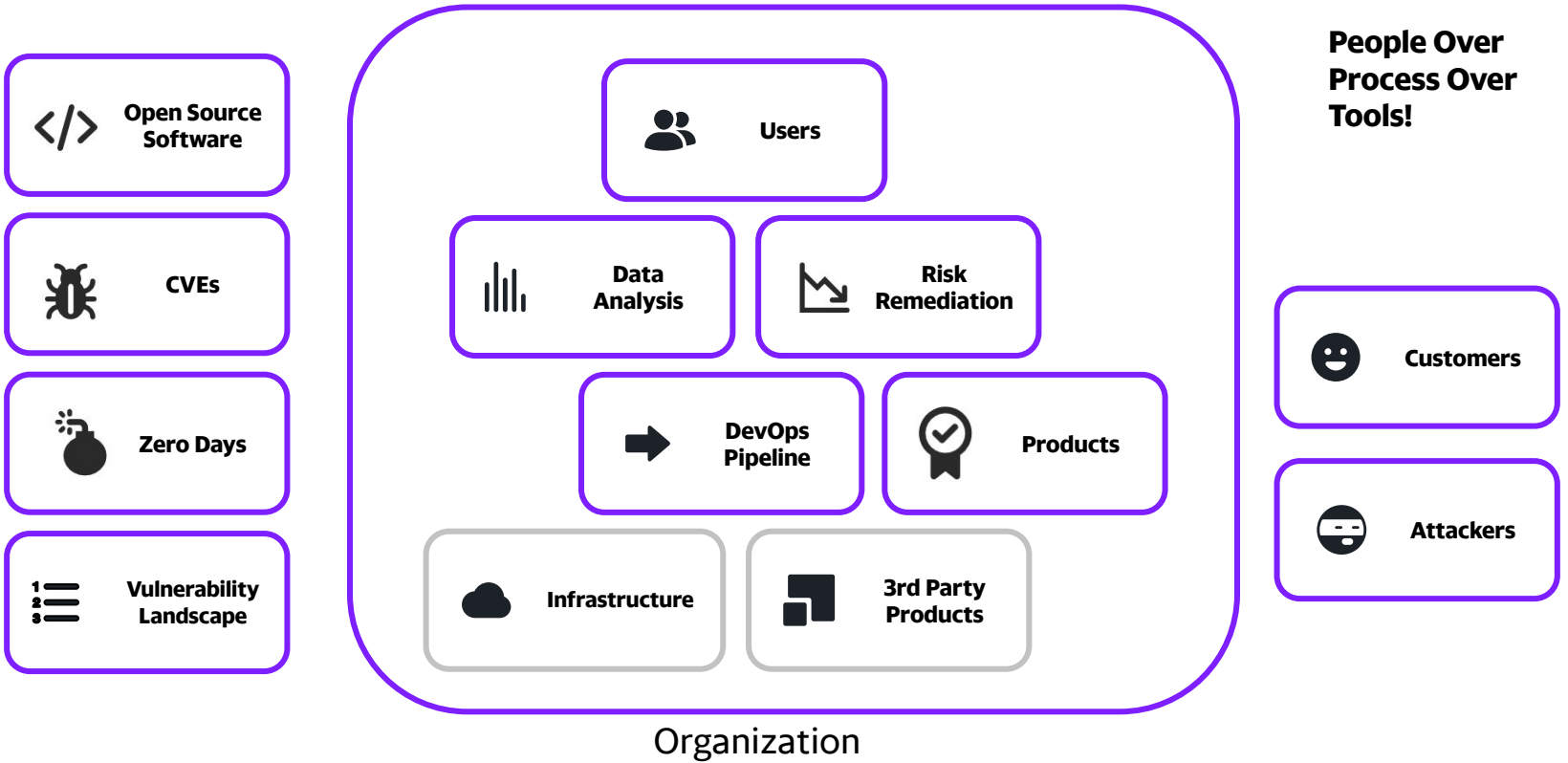
He likes to understand things deeply - and uses data analysis and dumb questions to build that understanding.

He's not big on titles, hierarchy or status quo.

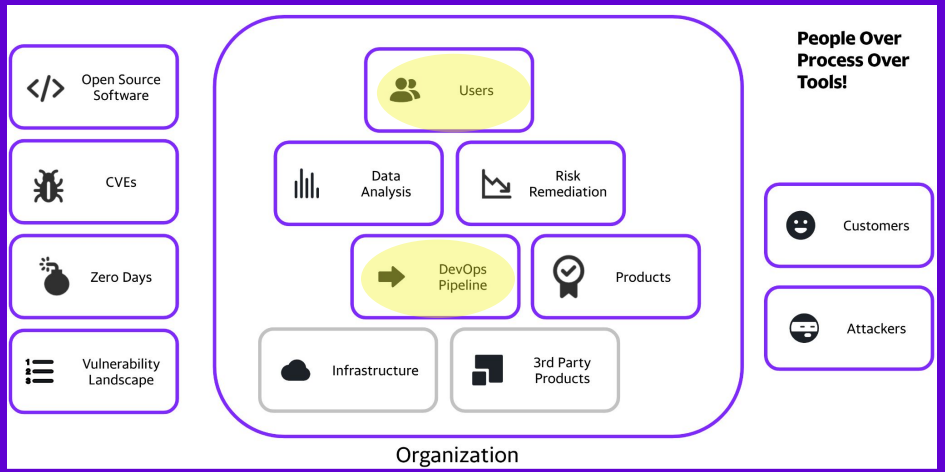
https://www.linkedin.com/in/chrisamadden

yahoo!

Context


Open Source Software
CVEs
Zero Days
Vulnerability Landscape
Users
Data Analysis
Risk Remediation
DevOps Pipeline
Products
Infrastructure
3rd Party Products
Organization
People Over Process Over Tools!
Customers
Attackers

DevOps Pipeline



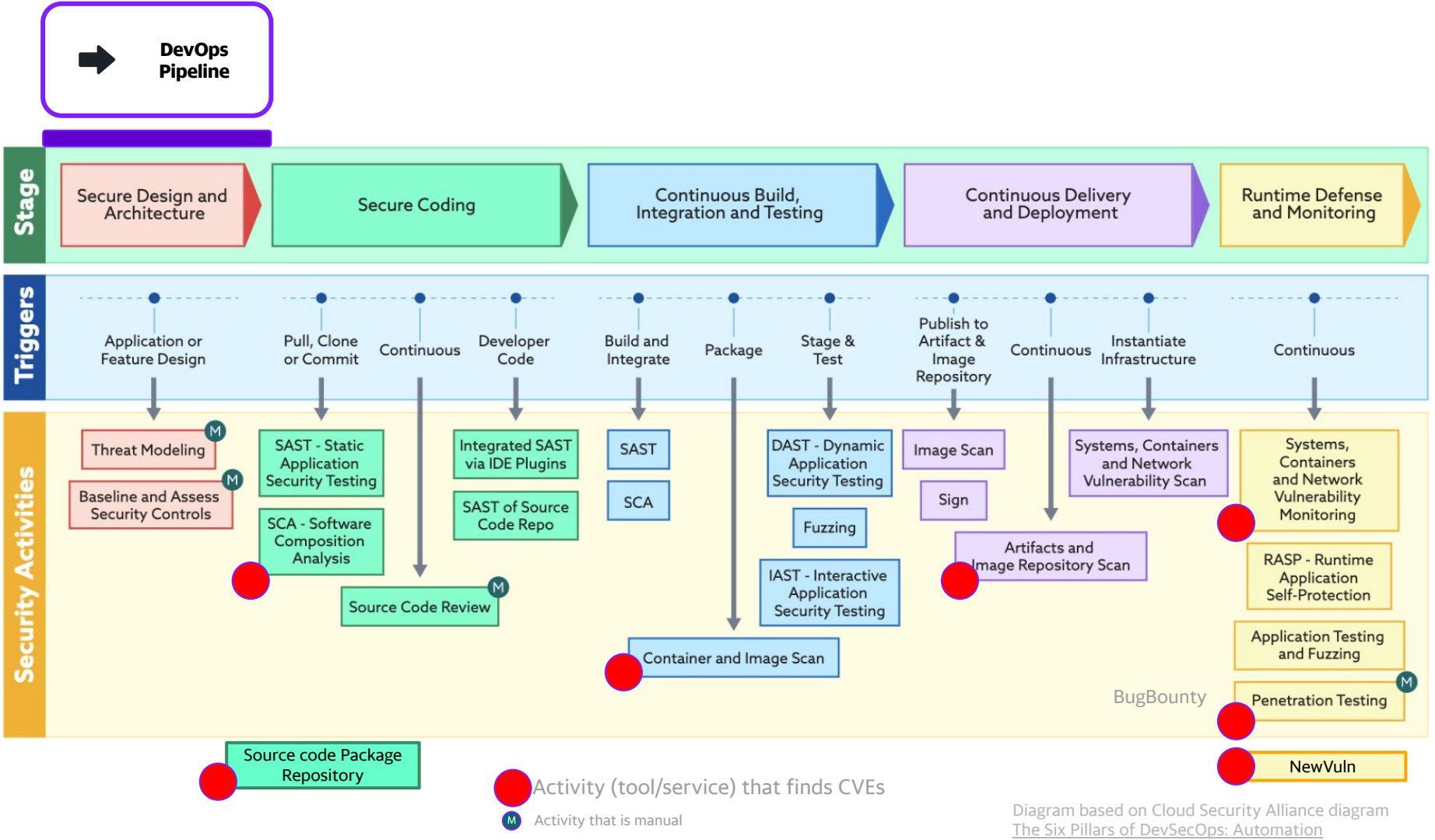
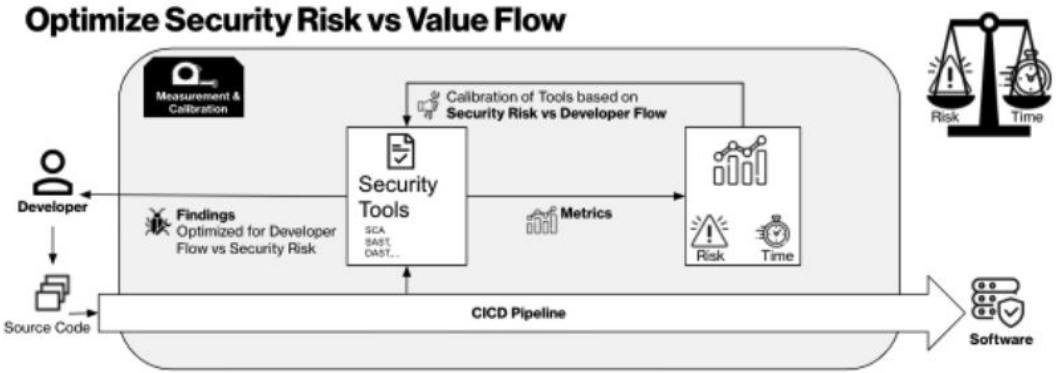


Diagram based on Cloud Security Alliance diagram
The Six Pillars of DevSecOps: Automation

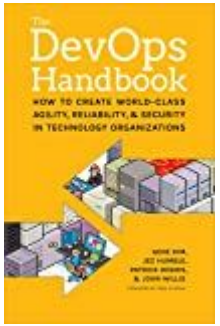
DevOps Pipeline

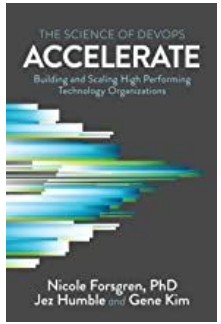
DevSecOps Model: Flow/Systems Thinking



System Level - Customer Focused:

- The system is our DevSecOps pipeline
- Value is delivered via the sw we deliver to customers via our DevSecOps pipeline





The Three Ways: The Principles Underpinning DevOps

- The First Way: Flow/Systems Thinking
- The Second Way: Amplify Feedback Loops
- The Third Way: Culture of Continual Experimentation and Learning

The First Way: Flow/Systems Thinking The First Way emphasizes the performance of the entire system, as opposed to the performance of a specific silo of work or department

We want to optimize SoftWare flow through the pipeline vs Risk

Users

As a **security person**, I exist to enable developers deliver software/value securely

As **anyone**, I want to optimize software flow versus Risk by fixing the vulnerabilities that need to be fixed first

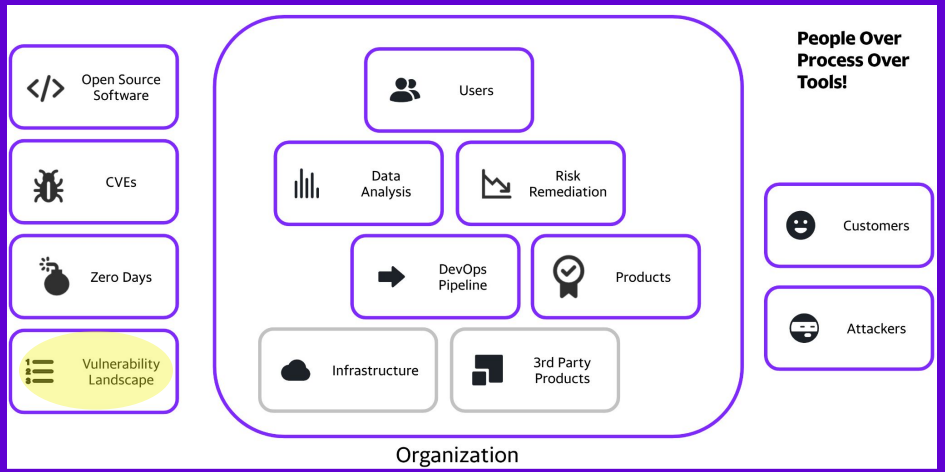
As a **developer**, I don't care about your security tool or team, I care about delivering software of high assurance (quality + security) quickly

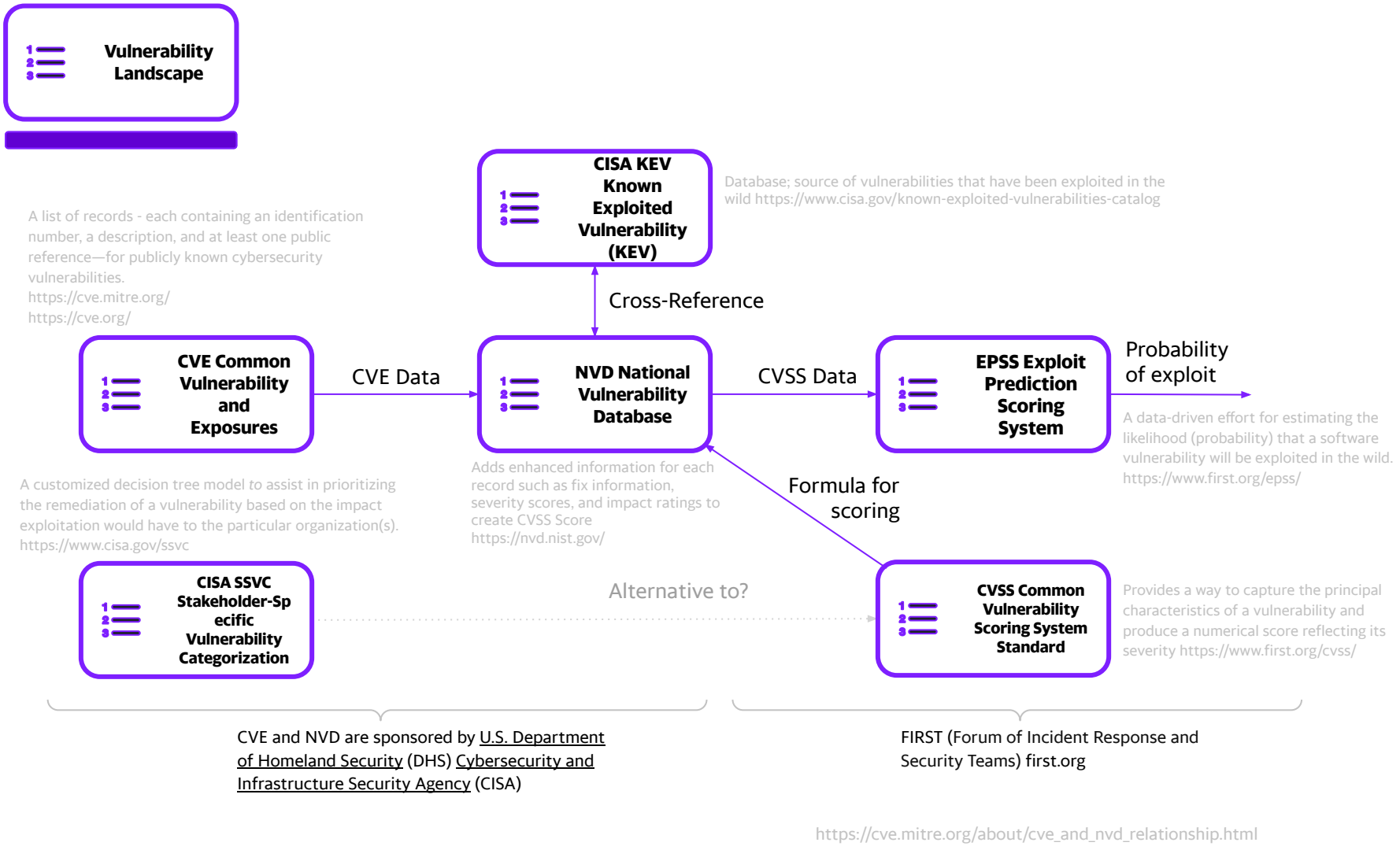
"Customers don't care about your solution. They care about their problems" Dave McClure

As a **CISO**, I want to know risk and remediation per Asset and for the organization

As a **developer/leader,** I want a unified prioritized personalized achievable view (across tools and teams) of what to fix first

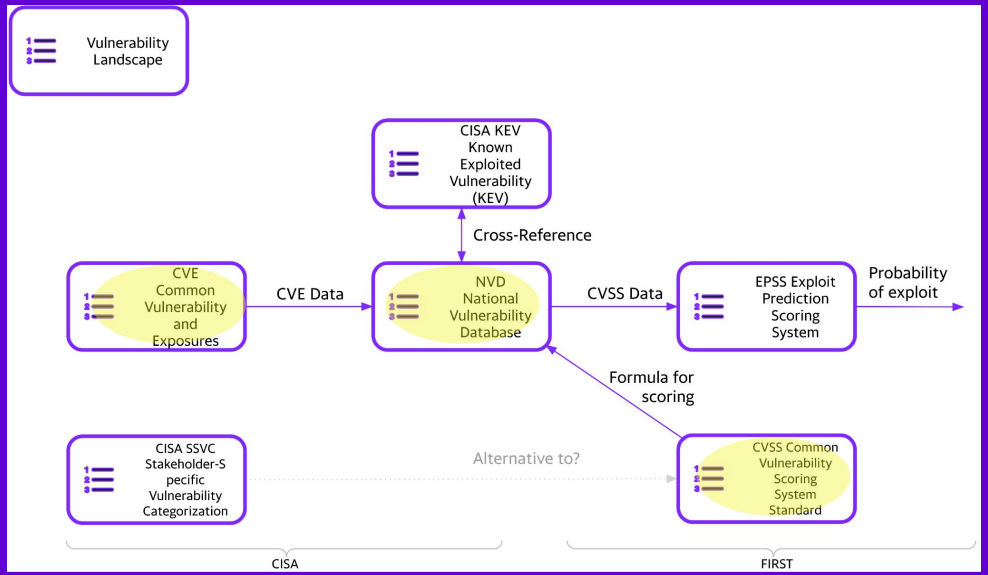
Vulnerability Landscape



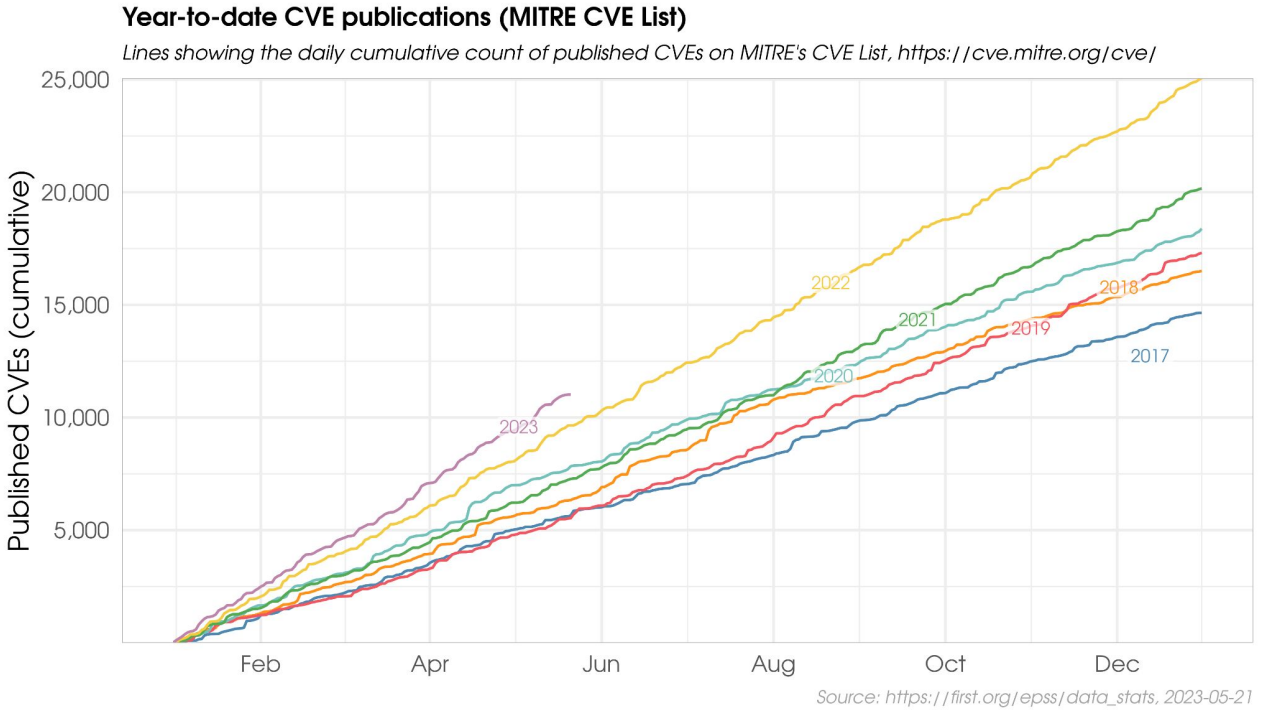

Vulnerability Landscape
A list of records - each containing an identification number, a description, and at least one public reference—for publicly known cybersecurity vulnerabilities.
https://cve.mitre.org/
https://cve.org/
CISA KEV Known Exploited Vulnerability (KEV)
Database; source of vulnerabilities that have been exploited in the wild https://www.cisa.gov/known-exploited-vulnerabilities-catalog
Cross-Reference
CVE Common Vulnerability and Exposures
CVE Data
NVD National Vulnerability Database
CVSS Data
EPSS Exploit Prediction Scoring System
Probability of exploit
A data-driven effort for estimating the likelihood (probability) that a software vulnerability will be exploited in the wild.
https://www.first.org/epss/
Adds enhanced information for each record such as fix information, severity scores, and impact ratings to create CVSS Score
https://nvd.nist.gov/
Formula for scoring
A customized decision tree model to assist in prioritizing the remediation of a vulnerability based on the impact exploitation would have to the particular organization(s).
https://www.cisa.gov/ssvc
CISA SSVC Stakeholder-Specific Vulnerability Categorization
Alternative to?
CVSS Common Vulnerability Scoring System Standard
Provides a way to capture the principal characteristics of a vulnerability and produce a numerical score reflecting its severity https://www.first.org/cvss/
CVE and NVD are sponsored by U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA)
FIRST (Forum of Incident Response and Security Teams) first.org
https://cve.mitre.org/about/cve_and_nvd_relationship.html

CVE CVSS

Common Vulnerabilities and Exposures (CVE)
Common Vulnerability Scoring System (CVSS)



CVE Publications



The rate of new CVEs is increasing!

Vulnerability Landscape

NVD National Vulnerability Database

CVE CVSS Example Base Score

CVE-2021-44228 Detail

MODIFIED

This vulnerability has been modified since it was last analyzed by the NVD. It is awaiting reanalysis which may result in further changes to the information provided.

Description

Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages, and parameters do not protect against attacker controlled LDAP and other JNDI related endpoints. An attacker who can control log messages or log message parameters can execute arbitrary code loaded from LDAP servers when message lookup substitution is enabled. From log4j 2.15.0, this behavior has been disabled by default. From version 2.16.0 (along with 2.12.2, 2.12.3, and 2.3.1), this functionality has been completely removed. Note that this vulnerability is specific to log4j-core and does not affect log4net, log4cxx, or other Apache Logging Services projects.

Severity

CVSS Version 3.x | CVSS Version 2.0

CVSS 3.x Severity and Metrics:

NIST: NVD **Base Score:** 10.0 CRITICAL **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We

CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available informa

a score within the CVE List.

CVSS v3.1 Severity and Metrics:
Base Score: 10.0 CRITICAL
Vector: AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H
Impact Score: 6.0
Exploitability Score: 3.9

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

References to Advisories, Solutions, and Too

By selecting these links, you will be leaving NIST webspace. We have provided these … have information that would be of interest to you. No inferences should be drawn on acc… 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
http://packetstormsecurity.com/files/165225/Apache-Log4j2-2.14.1-Remote-Code-Execution.html	Third Party Advisory, VDB Entry
http://packetstormsecurity.com/files/165260/VMware-Security-Advisory-2021-0028.html	Third Party Advisory, VDB Entry
http://packetstormsecurity.com/files/165261/Apache-Log4j2-2.14.1-Information-Disclosure.html	Exploit, Third Party Advisory

https://nvd.nist.gov/vuln/detail/CVE-2021-44228

QUICK INFO

CVE Dictionary Entry:
CVE-2021-44228
NVD Published Date:
12/10/2021
NVD Last Modified:
04/03/2023
Source:
Apache Software Foundation

This CVE is in CISA's Known Exploited Vulnerabilities Catalog

Reference CISA's BOD 22-01 and Known Exploited Vulnerabilities Catalog for further guidance and requirements.

Vulnerability Name	Date Added	Due Date	Required Action
Apache Log4j2 Remote Code Execution Vulnerability	12/10/2021	12/24/2021	For all affected software assets for which updates exist, the only acceptable remediation actions are: 1) Apply updates; OR 2) remove affected assets from agency networks. Temporary mitigations using one of the measures provided at https://www.cisa.gov/uscert/ed-22-02-apache-log4j-recommended-mitigation-measures are only acceptable until updates are available.

Weakness Enumeration

CWE-ID	CWE Name	Source
CWE-917	Improper Neutralization of Special Elements used in an Expression Language Statement ('Expression La	NIST
CWE-20	Improper Input Validation	Apache Software Foundation
CWE-502	Deserialization of Untrusted Data	Apache Software Foundation
CWE-400	Uncontrolled Resource Consumption	Apache Software Foundation

Base Score **10.0 (Critical)**

Attack Vector (AV)
Network (N) | Adjacent (A) | Local (L) | Physical (P)

Attack Complexity (AC)
Low (L) | High (H)

Privileges Required (PR)
None (N) | Low (L) | High (H)

User Interaction (UI)
None (N) | Required (R)

Scope (S)
Unchanged (U) | Changed (C)

Confidentiality (C)
None (N) | Low (L) | High (H)

Integrity (I)
None (N) | Low (L) | High (H)

Availability (A)
None (N) | Low (L) | High (H)

https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

CVE CVSS Base Score is determined by 8 parameters and their values

Vulnerability Landscape

NVD National Vulnerability Database

CVSS Score vs Exploitation

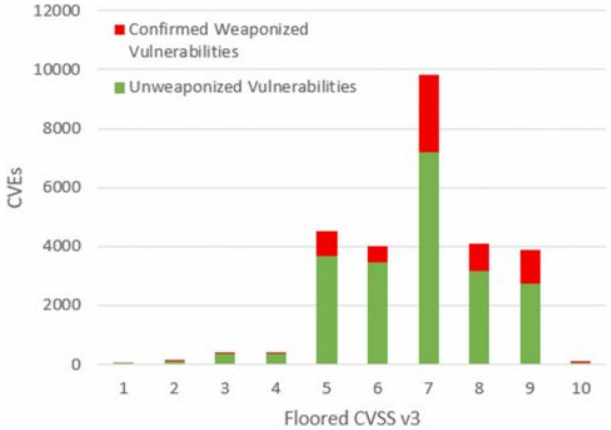
CVSS score performs no better than randomly picking vulnerabilities to fix and may lead to negligible risk reductions
Comparing Vulnerability Severity and Exploits Using Case-Control Studies, 2014

There's no inherent correlation between the vulnerability and if threat actors are exploiting them in terms of those severity ratings Gartner, Nov 2021

Tenable

According to Tenable Research (2022), 56% of all vulnerabilities are scored as High (CVSS score of 7.0–8.9) or Critical (CVSS score of 9.0–10.0), regardless of whether they are likely to ever be exploited. And, ***since more than 75% of all vulnerabilities with a score of 7 or above have never had an exploit published against them, security teams using CVSS to prioritize their efforts are wasting the majority of their time chasing after the wrong issues*** (using CVSS v3.* score)

Henry Howland, Drew University

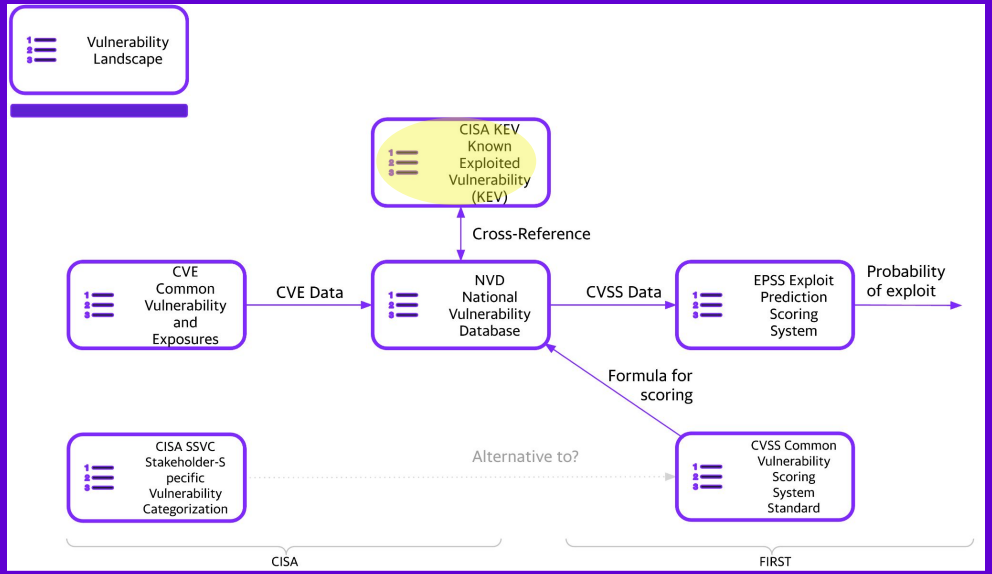


These findings for CVSS v3 fall in line with studies of CVSS v2, which similarly found that ***remediating all vulnerabilities with a high severity was largely ineffective at stopping cyber-attacks*** *[5, 31].*
CVSS: Ubiquitous and Broken, February 2022

CVSS Score is not a good Predictor of Exploitability - so don't use it alone to Prioritize!

CISA KEV

Cybersecurity and Infrastructure Security Agency (CISA) Known Exploited Vulnerabilities (KEV)



Vulnerability Landscape

CISA KEV Known Exploited Vulnerability (KEV)

For the benefit of the cybersecurity community and network defenders—and to help every organization better manage vulnerabilities and keep pace with threat activity— ***CISA maintains the authoritative source of vulnerabilities that have been exploited in the wild: the Known Exploited Vulnerability (KEV) catalog***

Why?

"many vulnerabilities classified as "critical" are highly complex and have never been seen exploited in the wild - in fact, less than 4% of the total number of CVEs have been publicly exploited. *But threat actors are extremely fast to exploit their vulnerabilities of choice: of those 4% of known exploited CVEs, 42% are being used on day 0 of disclosure; 50% within 2 days; and 75% within 28 days."* *BOD-22-01*

CVSS used as a measure of Risk

In 2019, the US Department of Homeland Security (DHS) issues a Binding Operational Directive (Binding Operational Directive 19-02, "*Vulnerability Remediation Requirements for Internet-Accessible Systems*") to all federal agencies describing how they must patch:

- Critical vulnerabilities (CVSS 9.0-10.0) within 15 days of detection
- High Severity vulnerabilities (CVSS 7.0-8.9) within 30 days of detection

CISA Guidance

"All federal civilian executive branch (FCEB) agencies are required to remediate vulnerabilities in the KEV catalog within prescribed timeframes under Binding Operational Directive (BOD) 22-01, Reducing the Significant Risk of Known Exploited Vulnerabilities. Although not bound by BOD 22-01, every organization, including those in state, local, tribal, and territorial (SLTT) governments ***and private industry can significantly strengthen their security and resilience posture by prioritizing the remediation of the vulnerabilities listed in the KEV catalogue as well. CISA strongly recommends all stakeholders include a requirement to immediately address KEV catalogue vulnerabilities as part of their vulnerability management plan****.*

CISA KEV advice: Remediate vulnerabilities in the KEV catalog immediately

Vulnerability Landscape

CISA KEV Known Exploited Vulnerability (KEV)

What it looks like

CISA KEV

CYBERSECURITY & INFRASTRUCTURE SECURITY AGENCY

AMERICA'S CYBER DEFENSE AGENCY

Search

Topics | Spotlight | Resources & Tools | News & Events | Careers | About | REPORT A CYBER ISSUE

Home

SHARE:

Known Exploited Vulnerabilities Catalog

CVE	Vendor/Project	Product	Vulnerability Name	Date Added to Catalog	Short Description	Action	Due Date
CVE-2021-45046	Apache	Log4j2	Apache Log4j2 Deserialization of Untrusted Data Vulnerability	2023-05-01	Apache Log4j2 contains a deserialization of untrusted data vulnerability due to the incomplete fix of CVE-2021-44228, where the Thread Context Lookup Pattern is vulnerable to remote code execution in certain non-default configurations.	Apply updates per vendor instructions.	2023-05-22
Notes	https://logging.apache.org/log4j/2.x/security.html						
CVE-2021-44228	Apache	Log4j2	Apache Log4j2 Remote Code Execution Vulnerability	2021-12-10	Apache Log4j2 contains a vulnerability where JNDI features do not protect against attacker-controlled JNDI-related endpoints, allowing for remote code execution.	For all affected software assets for which updates exist, the only acceptable remediation actions are: 1) Apply updates; OR 2) remove affected assets from agency networks. Temporary mitigations using one of the measures provided at https://www.cisa.gov/uscert/ed-22-02-apache-log4j-recommended-mitigation-measures are only acceptable until updates are available.	2021-12-24

https://www.cisa.gov/known-exploited-vulnerabilities-catalog Downloadable as a file in different formats

NIST NVD

NIST

Information Technology Laboratory

NATIONAL VULNERABILITY DATABASE

VULNERABILITIES

CVE-2023-28252 Detail

Description

Windows Common Log File System Driver Elevation of Privilege Vulnerability

This CVE is in CISA's Known Exploited Vulnerabilities Catalog

Reference CISA's BOD 22-01 and Known Exploited Vulnerabilities Catalog for further guidance and requirements.

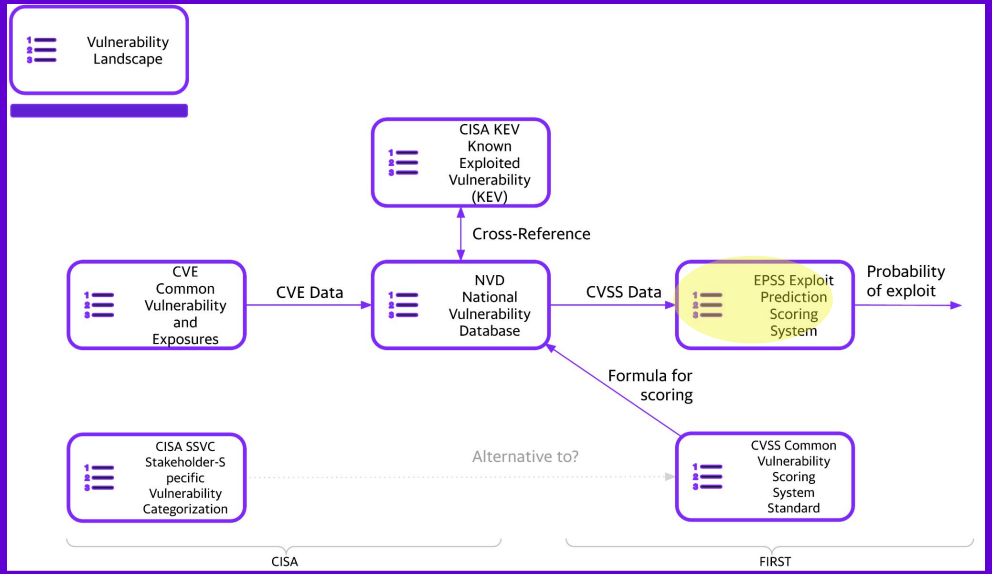
Vulnerability Name	Date Added	Due Date	Required Action
Microsoft Windows Common Log File System (CLFS) Driver Privilege Escalation Vulnerability	04/11/2023	05/02/2023	Apply updates per vendor instructions.

https://nvd.nist.gov/vuln/detail/CVE-2023-28252

CISA KEV and NIST NVD both link to each other

EPSS

Exploit Prediction Scoring System (EPSS)



Vulnerability Landscape

EPSS Exploit Prediction Scoring System

Why?

The Exploit Prediction Scoring System (EPSS) is an open, data-driven effort for **estimating the likelihood (probability) that a software vulnerability will be exploited in the wild**.

- **Its goal is to assist network defenders in better prioritizing vulnerability remediation efforts in conjunction with an existing CVSS score.**

EPSS uses current threat information from the CVE database combined with real-world exploit data for its predictions.

- EPSS then produces a probability score of between 0 and 1 (0 and 100%).
- The higher the score, the greater the probability that a vulnerability will be exploited in the next 30 days.

Covers all Published CVEs (not zero day vulnerabilities, or flaws that may never be assigned a CVE ID, or CVEs in Reserved or Rejected status).

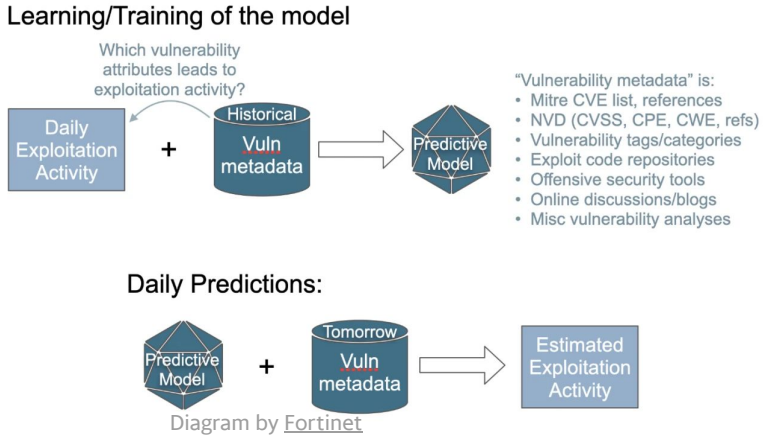
EPSS Data

https://api.first.org/data/v1/epss?cve=CVE-2021-44228

{"cve":"CVE-2021-44228","epss":"0.975780000","percentile":"0.999990000","date":"2023-04-17"}

https://www.first.org/epss/data_stats to download a snapshot and see other EPSS data reports

EPSS Model



Sources include Ground Truth: Daily observations of exploitation-in-the-wild activity. *EPSS collects and aggregates evidence of exploits from multiple sources: Fortiguard, Alienvault OTX, the Shadow Server Foundation and GreyNoise.*

Each of these data sources employ network- or host-layer intrusion detection/prevention systems (IDS/IPS), or honeypots, in order to identify attempted exploitation.

These systems are also predominantly signature-based (as opposed to anomaly-based) detection systems.

EPSS Probability Score: Probability of observing exploitation activity in the next 30 days

Vulnerability Landscape

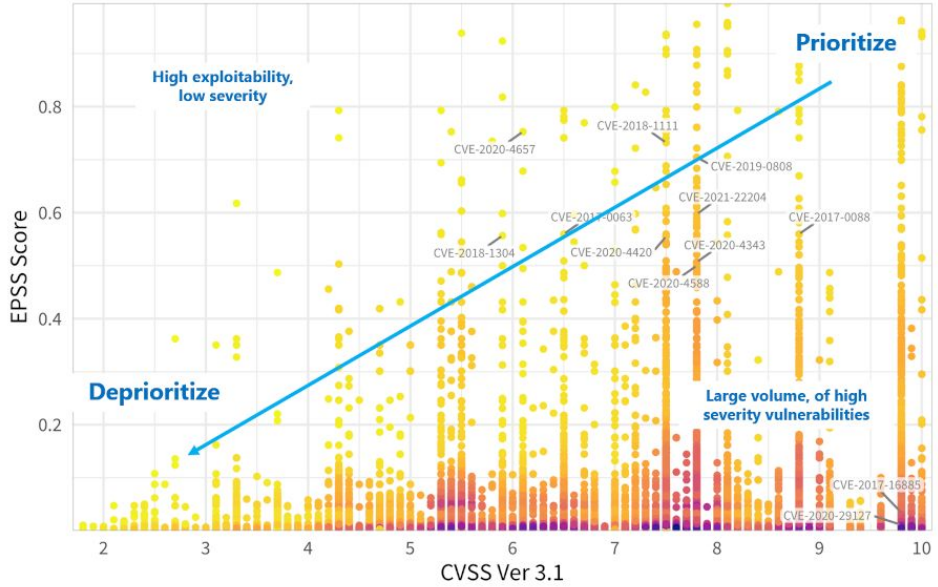
EPSS Exploit Prediction Scoring System

EPSS User Guide

Using EPSS Score

EPSS score compared to CVSS Base Score (NVD)

Point density is represented by color, yellow is less dense going through red to a deep purple for the most dense areas. Labeling a random sample of CVEs with higher values for reference.



Source: https://first.org/epss/data_stats, 2021-05-16

"If it's got a high EPSS score I should definitely be worried about it. If it's got a low EPSS score, I can't be certain whether I should be worried or not.

Most CVEs will have a low EPSS score near zero - whether there is a high or low probability of Exploit.

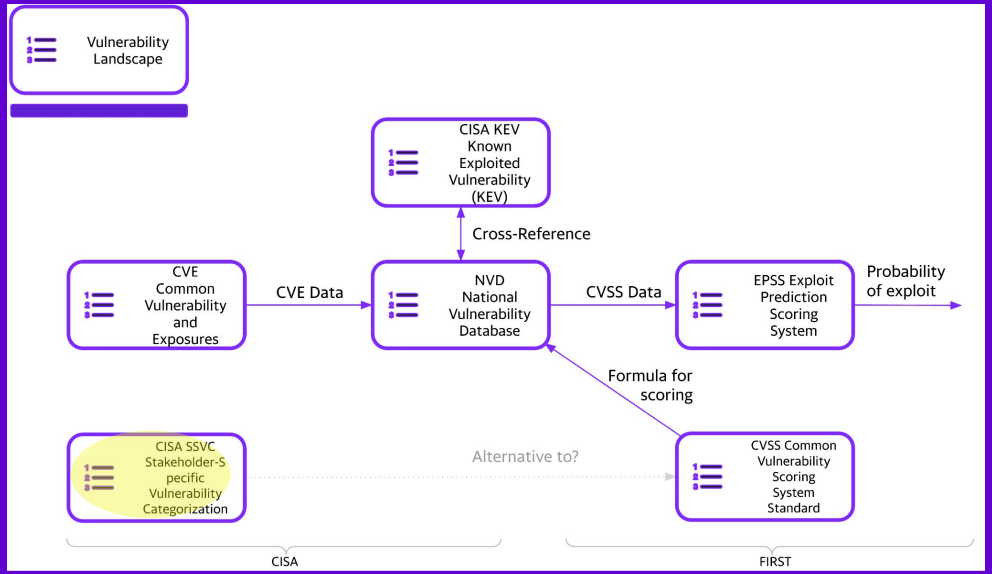
Neither the CVSS score, nor the EPSS score, are linear - so the straight line prioritization is for illustrative purposes only.

CISA KEV could also be used in conjunction with CVSS and EPSS.
See comments on CISA KEV in Enhancing Vulnerability Prioritization: Data-Driven Exploit Predictions with Community-Driven Insights, Feb 2023

If you are going to use CVSS Score for Prioritization, as a first step, EPSS can be used with CVSS (versus CVSS alone). Impact should also be assessed.

CISA SSVC

Cybersecurity and Infrastructure Security Agency (CISA) Stakeholder-Specific Vulnerability Categorization (SSVC)



Vulnerability Landscape

CISA SSVC Stakeholder Specific Vulnerability Categorization

"The goal of SSVC is to assist in prioritizing the remediation of a vulnerability based on the impact exploitation would have to the particular organization(s)."

"CISA encourages every organization to use a vulnerability management framework that considers a vulnerability's exploitation status, such as SSVC."

https://www.cisa.gov/ssvc-calculator#SSVCv2/E:A/A:Y/T:T/P:E/B:I/M:H/D:C/2023-04-18T18:10:41Z/

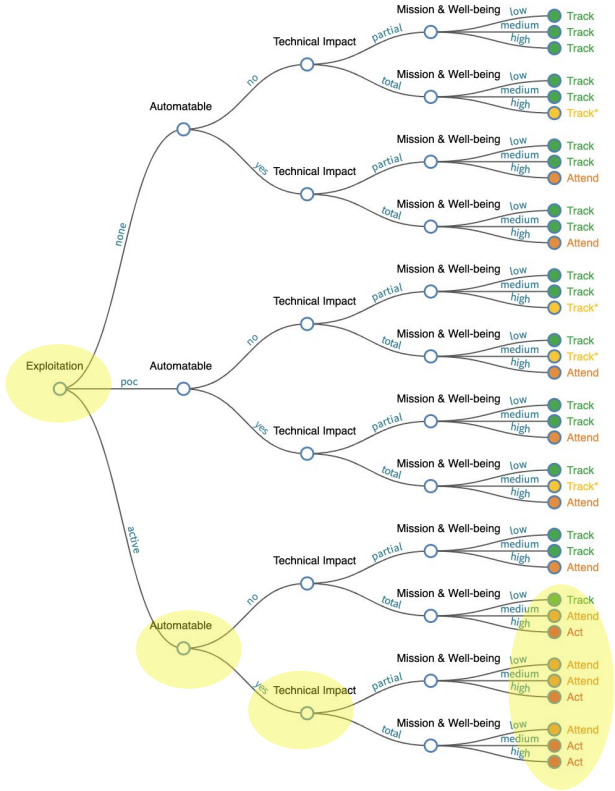
Decision

Track The vulnerability does not require attention outside of Vulnerability Management (VM) at this time. Continue to track the situation and reassess the severity of vulnerability if necessary.

Track* Track these closely, especially if mitigation is unavailable or difficult. Recommended that analyst discuss with other ana-lysts and get a second opinion.

Attend The vulnerability requires to be attended to by stakeholders outside VM. The action is a request to others for assistance / information / details, as well as a potential publication about the issue.

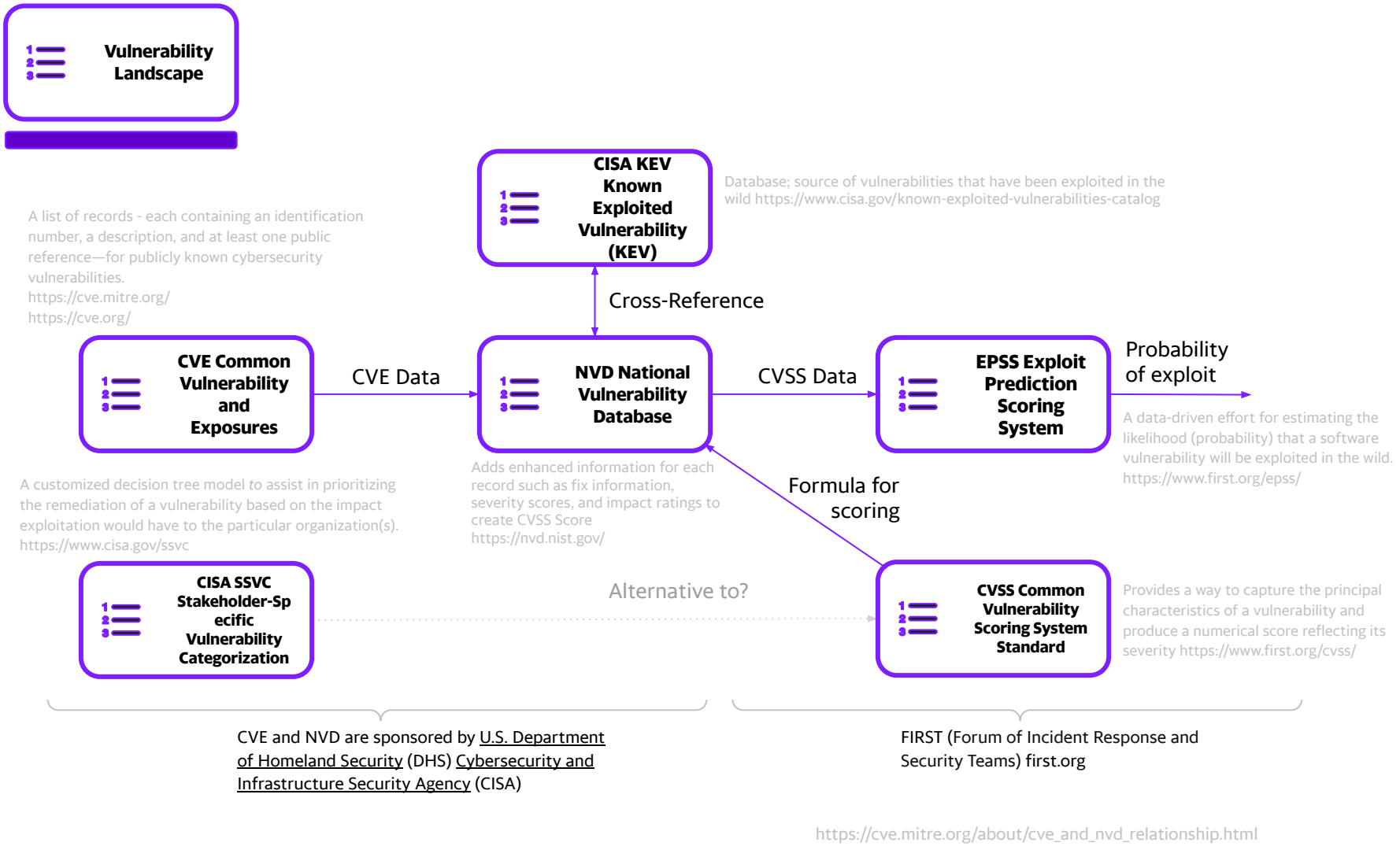
Act The vulnerability requires immediate action by the relevant leadership. The action is a high-priority meeting among the relevant supervisors to decide how to respond.



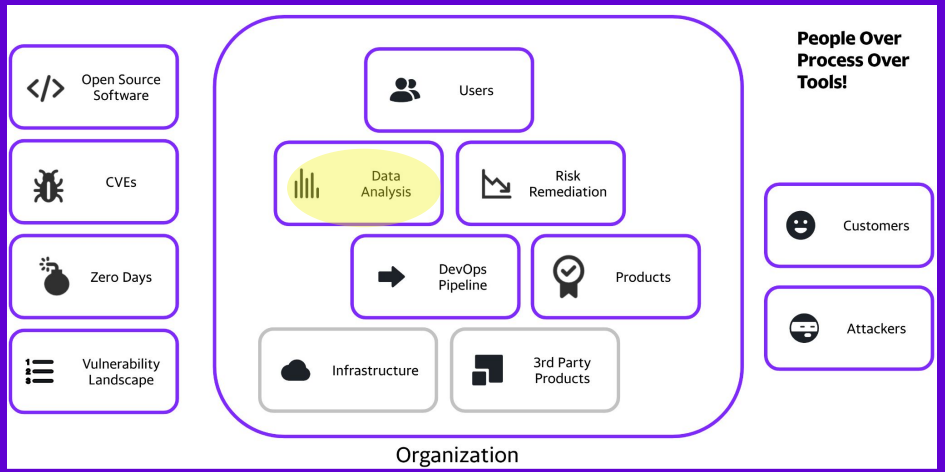
CISA SSVC is based on CMU SEI (Carnegie Mellon University Software Engineering Institute):

- "Prioritizing Vulnerability Response: A Stakeholder-Specific Vulnerability Categorization (Version 2.0)"
- Coordinated Vulnerability Disclosure User Stories

1. Use vulnerability exploitation status. 2. Prioritize based on impact to the organization


Vulnerability Landscape
A list of records - each containing an identification number, a description, and at least one public reference—for publicly known cybersecurity vulnerabilities.
https://cve.mitre.org/
https://cve.org/
CISA KEV Known Exploited Vulnerability (KEV)
Database; source of vulnerabilities that have been exploited in the wild https://www.cisa.gov/known-exploited-vulnerabilities-catalog
Cross-Reference
CVE Common Vulnerability and Exposures
CVE Data
NVD National Vulnerability Database
CVSS Data
EPSS Exploit Prediction Scoring System
Probability of exploit
A data-driven effort for estimating the likelihood (probability) that a software vulnerability will be exploited in the wild.
https://www.first.org/epss/
Adds enhanced information for each record such as fix information, severity scores, and impact ratings to create CVSS Score
https://nvd.nist.gov/
Formula for scoring
A customized decision tree model to assist in prioritizing the remediation of a vulnerability based on the impact exploitation would have to the particular organization(s).
https://www.cisa.gov/ssvc
CISA SSVC Stakeholder-Specific Vulnerability Categorization
Alternative to?
CVSS Common Vulnerability Scoring System Standard
Provides a way to capture the principal characteristics of a vulnerability and produce a numerical score reflecting its severity https://www.first.org/cvss/
CVE and NVD are sponsored by U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA)
FIRST (Forum of Incident Response and Security Teams) first.org
https://cve.mitre.org/about/cve_and_nvd_relationship.html

All CVEs Data Analysis



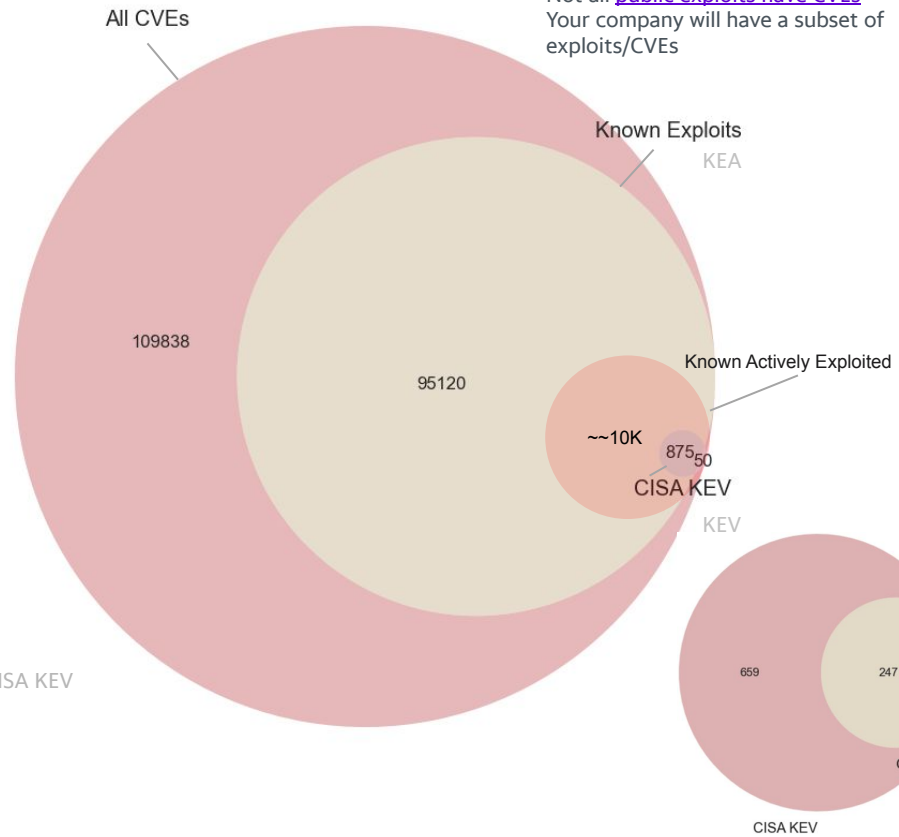
Data Analysis

CVEs Exploit %

~~50% (~93K) of all CVEs (~200K) have known exploits available (VendorDB)
~~5% (~10K) of all CVEs are actively exploited
~~10% of CVEs with Known Exploits Available (KEA) are known exploited

~~0.5% (~1K) of all CVEs (~200K) are in CISA Known Exploited Vulnerability
~~5% (50) of all CISA KEV CVEs (~1K) are not listed in Vendor DBs

Not all exploits are public/known
Not all public exploits have CVEs
Your company will have a subset of exploits/CVEs



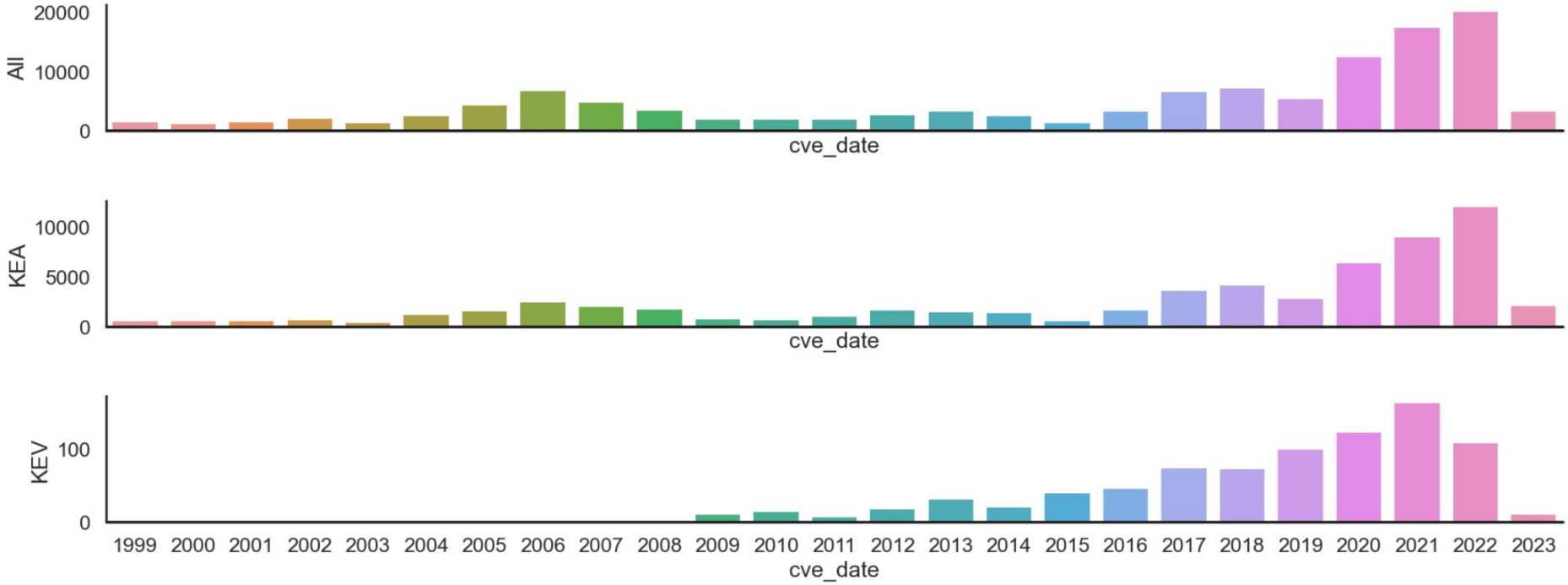
Various references for ~~5% actively exploited

- "Less than 3% of vulnerabilities have weaponized exploits or evidence of exploitation in the wild, two attributes posing the highest risk" Qualsys
- "less than 4% of the total number of CVEs have been publicly exploited", CISA KEV
- "we observe exploits in the wild for 5.5% of vulnerabilities in our dataset" "first.org EPSS.
- "Only 3 percent of critical vulnerabilities are worth prioritizing" https://www.datadoghq.com/state-of-application-security/

While Known Exploit Available is a good indicator of risk (better than CVSS score) - knowing that a CVE is being actively exploited is a whole lot better.

Google Project Zero cases of zero-day exploits that were detected "in the wild".

CVEs by Date



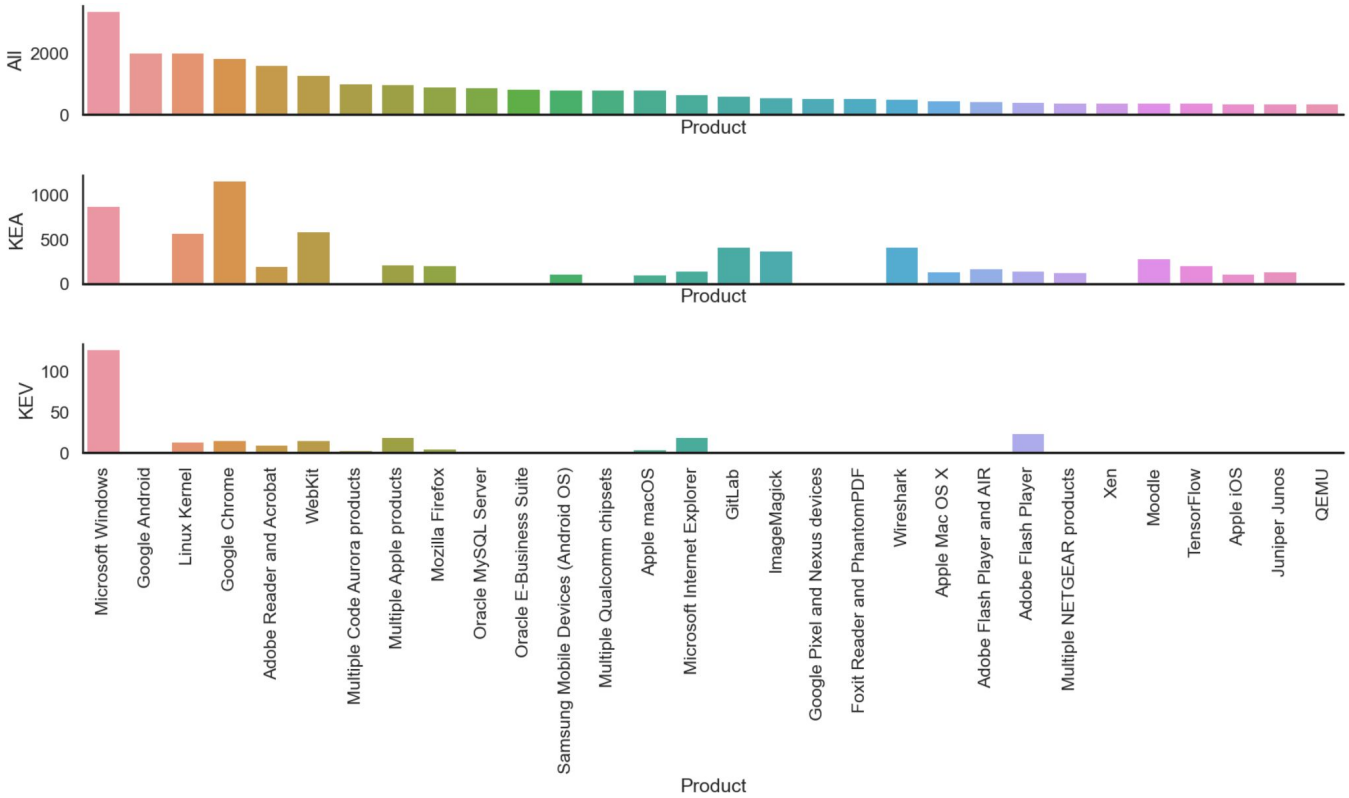
The count of CVEs per year is increasing - and the count of KEA and KEVs follows

ALL: All CVE IDs: ~200K. KEA: Known Exploit Available: ~90K. KEV: In CISA Known Exploited Vulnerability: ~1K

CVEs by Product

Note that if you have a CVE that is in CISA KEV, it does not mean you're using that Vendor product as listed in CISA KEV e.g. CVE-2015-4852 is attributed to Oracle WebLogic Server.

The vulnerability is in the associated open source library commons-collections-*.jar which you might be using in your apps.



Most CVEs are associated with OSs and Browsers
CVEs in CISA KEV may be in your apps/DevOps via an OSS dependency.

ALL: All CVE IDs: ~200K. KEA: Known Exploit Available: ~90K. KEV: In CISA Known Exploited Vulnerability: ~1K

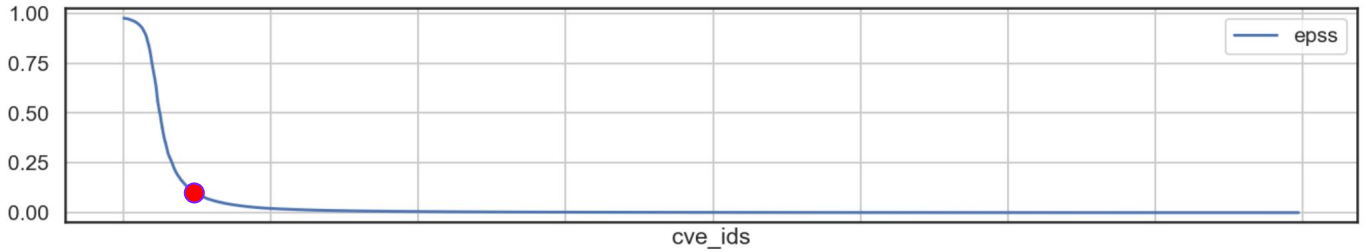
Data Analysis

CVEs EPSS Score Distributions

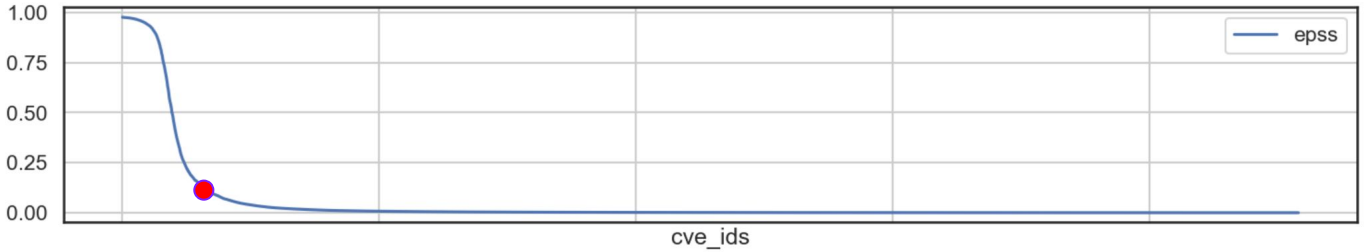
EPSS scores from high to low across CVEs in different datasets.

● EPSS = 0.1

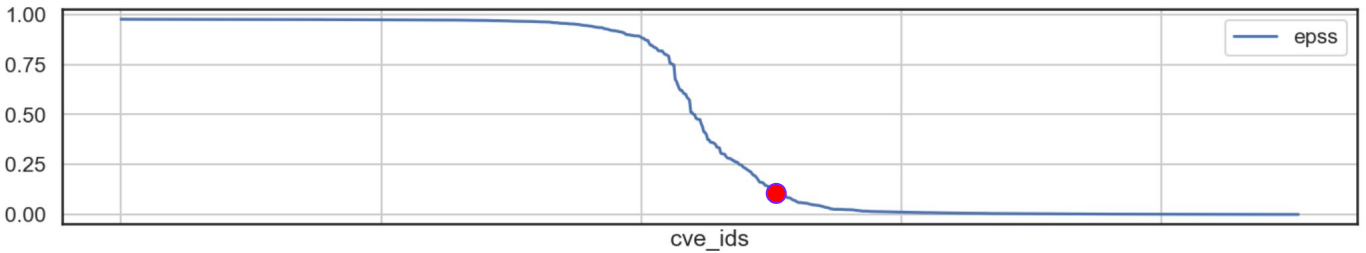
All



KEA



KEV



Most CVEs have low EPSS scores - EPSS is not telling us anything about these.

ALL: All CVE IDs: ~200K. KEA: Known Exploit Available: ~90K. KEV: In CISA Known Exploited Vulnerability: ~1K

Data Analysis

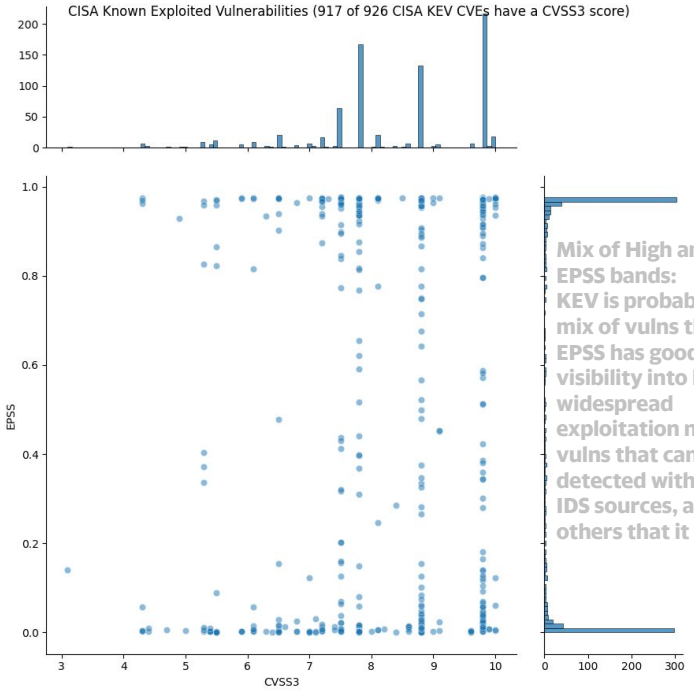
EPSS for CISA KEV, CISA KEV Top Routinely Exploited

All CVEs in the CISA KEV list, and CISA KEV "Top Routinely Exploited Cybersecurity Vulnerabilities" list per year, were known exploited (by definition).

Data Sources

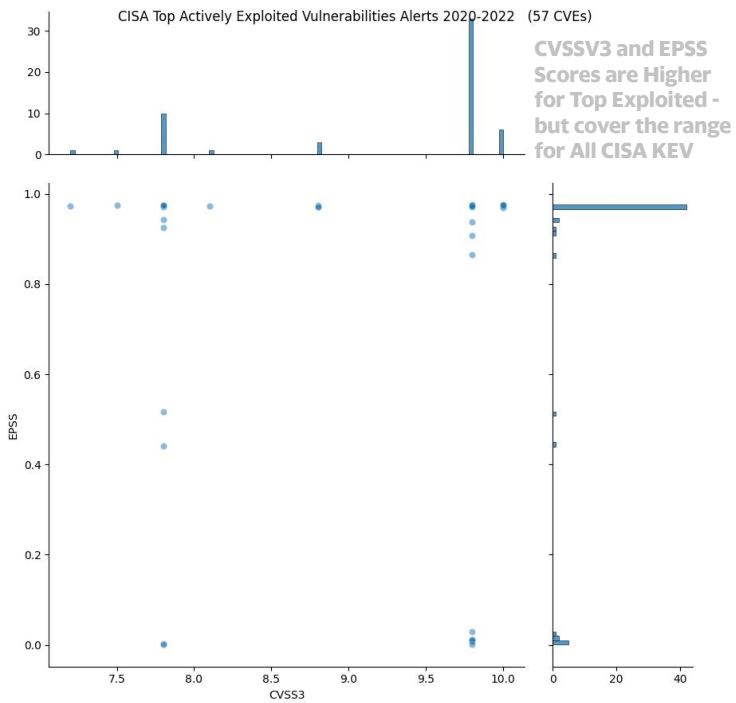
1. CISA Known Exploited Vulnerability catalog
2. CISA Top Routinely Exploited Vulnerabilities Alerts AA22-279A (2022), AA21-209A (2020-2021), AA22-117A (2021), AA20-133A (2016 to 2019). Some CVEs are duplicated across alerts.
3. EPSS

CISA KEV CVEs



Mix of High and Low EPSS bands: KEV is probably a mix of vulns that EPSS has good visibility into like widespread exploitation network vulns that can be detected with EPSS IDS sources, and others that it doesn't

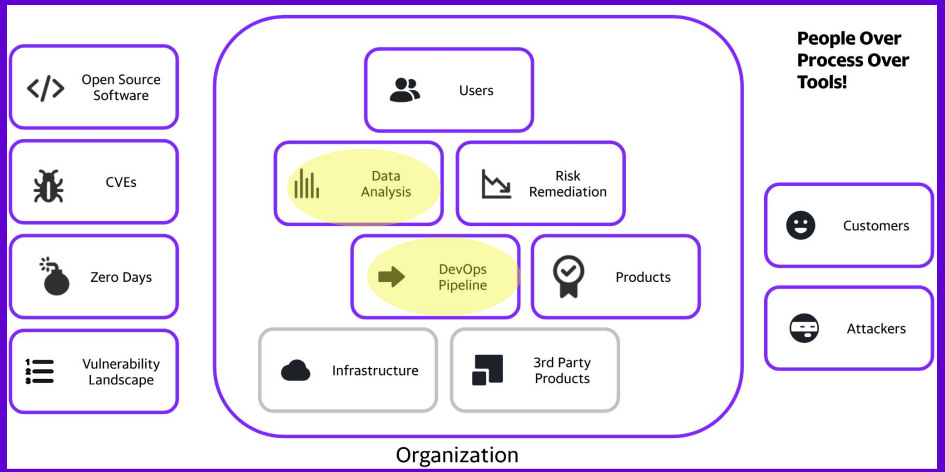
CISA KEV Top Routinely Exploited CVEs



CVSSV3 and EPSS Scores are Higher for Top Exploited - but cover the range for All CISA KEV

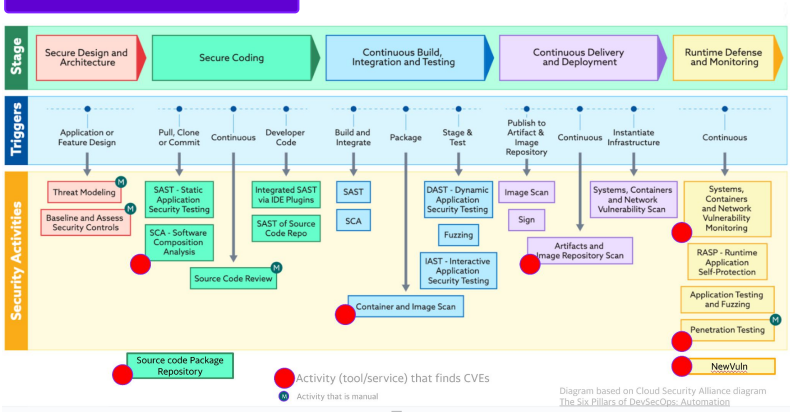
An EPSS score near zero should NOT be taken as a low probability of exploitation!
It could also be that EPSS has low information for that CVE so you can't rely on EPSS for that CVE!

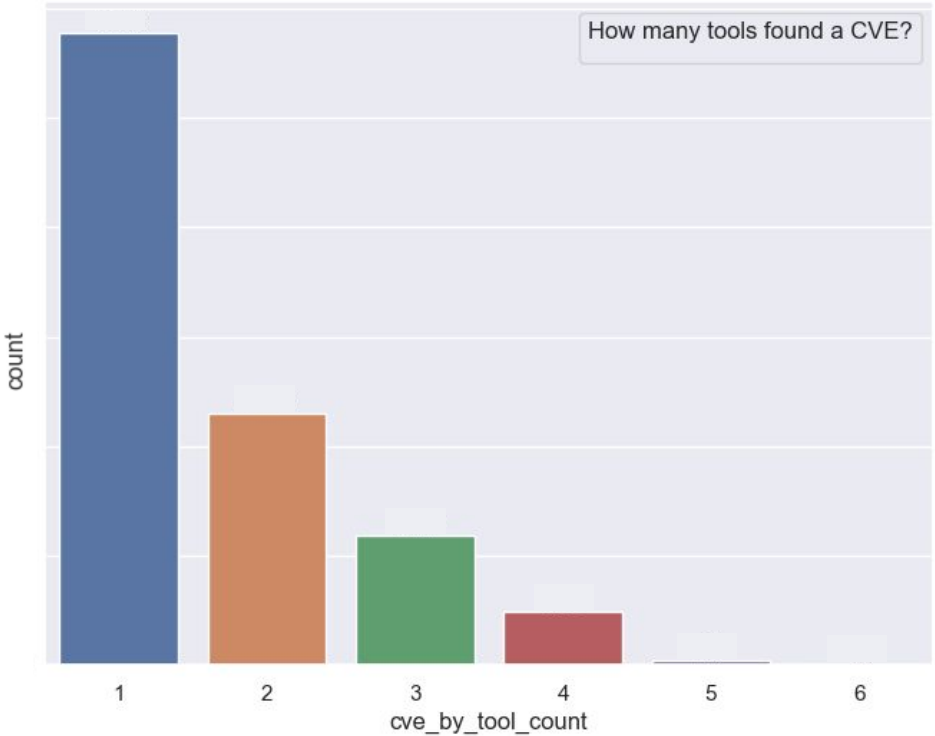
Yahoo CVEs Data Analysis



Data Analysis

Do the Tools find the same CVEs?

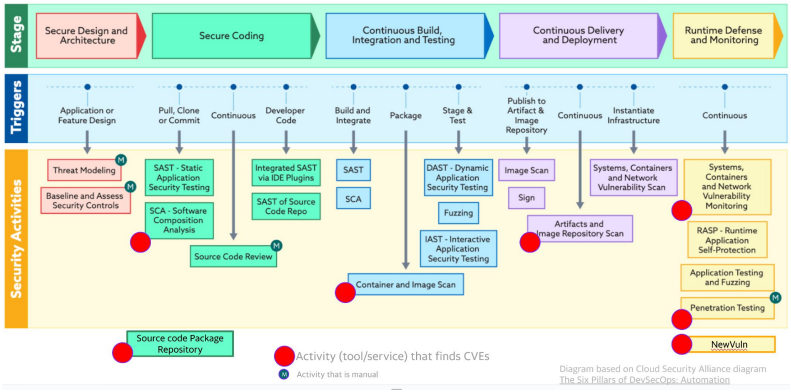


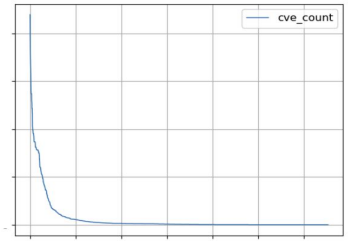


Most CVEs in our DevSecOps pipeline are found by 1 tool only

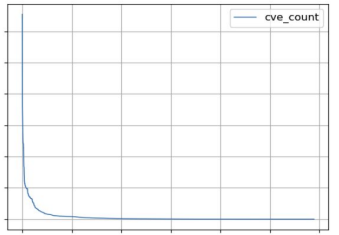
Data Analysis

DevOps CVEs Count Distribution

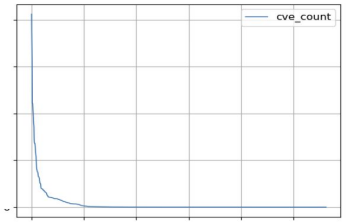




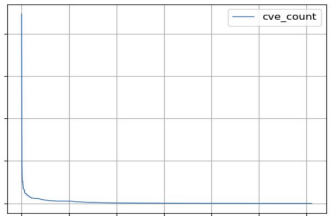
SCA



Container & Image Scan



Systems Containers and Network Vulnerability Monitoring

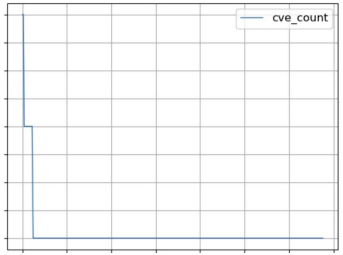


Artifacts and Image Repo Scan

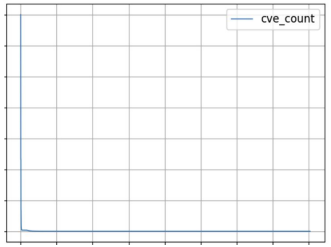
Observations

All Tools/Services follow a Pareto-type distribution.

- This suggests the Pareto effect can be applied i.e. addressing a relative small number of CVE IDs (with the most instances) will significantly reduce our total count of CVEs



NewVuln



Source Code Package Repository

The count of instances per CVEs for all Tools/Services follows a Pareto-type distribution. We can achieve a Pareto-effect as a result

For each plot: the count of instances of CVE IDs is the y-axis, where the CVE IDs are on the x-axis sorted by most instances of a CVE ID

Data Analysis

DevOps: SCA OSS Libraries CVEs

Industry averages

62%

Of Library Security Vulnerabilities are High Severity

7.5

Value	Frequency (%)
7.5	22.1%
9.8	15.6%
8.1	10.8%

Is the most common severity score (CVSS)

Paretos everywhere

X%

Of CVEs due to 1 specific library and associated versions which have multiple CVEs

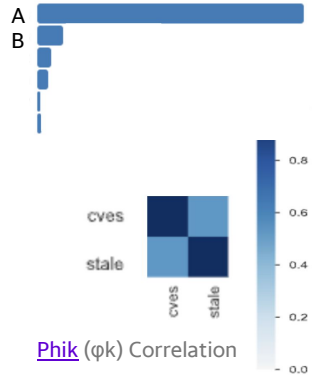
Y%

Of CVE counts due to the 0.Z% most common CVEs

Correlations

~2.5x more Language B libraries than A

Language A had the most CVEs (by far)



Phik (φk) Correlation

There is a high correlation between count of CVEs and stale libraries

Understanding Root Cause for YOUR CVEs is critical for YOUR Risk Remediation!

SCA = Software Composition Analysis OSS = Open Source Software

Data Analysis

HowTo: Exploratory Data Analysis

Tip

It is CRITICAL to understand your data before making decisions based on it!

- EDA is a minimal-effort high-value way to get that understanding.
- In other words, take your data as-is, and throw it at the tool, and see what comes back.

DevOps tools generally don't do a good job in going from data to intelligence.

- Export the data and EDA it.

ydata-profiling

```
ydata_profiling --title "Example Profiling Report"
--config_file default.yaml data.csv report.html
```

https://ydata-profiling.ydata.ai/docs/master/pages/getting_started/quickstart.html

PandasGUI

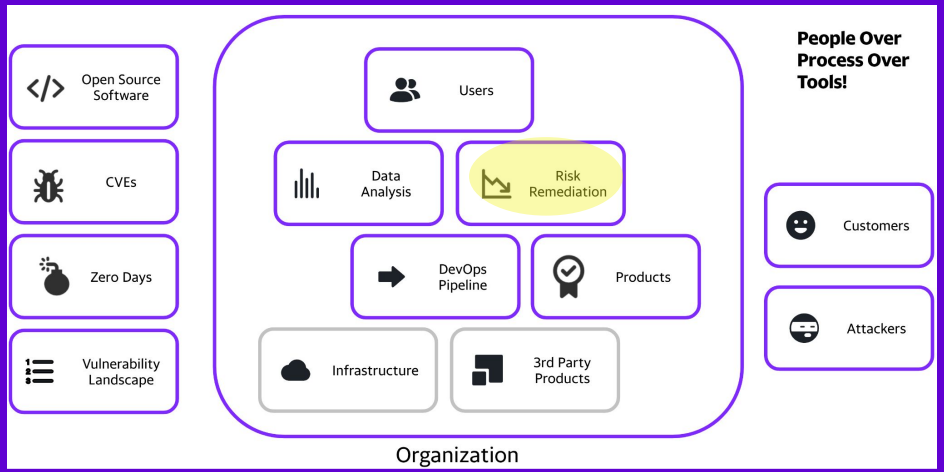
View, plot and analyze your data - via dragNdrop

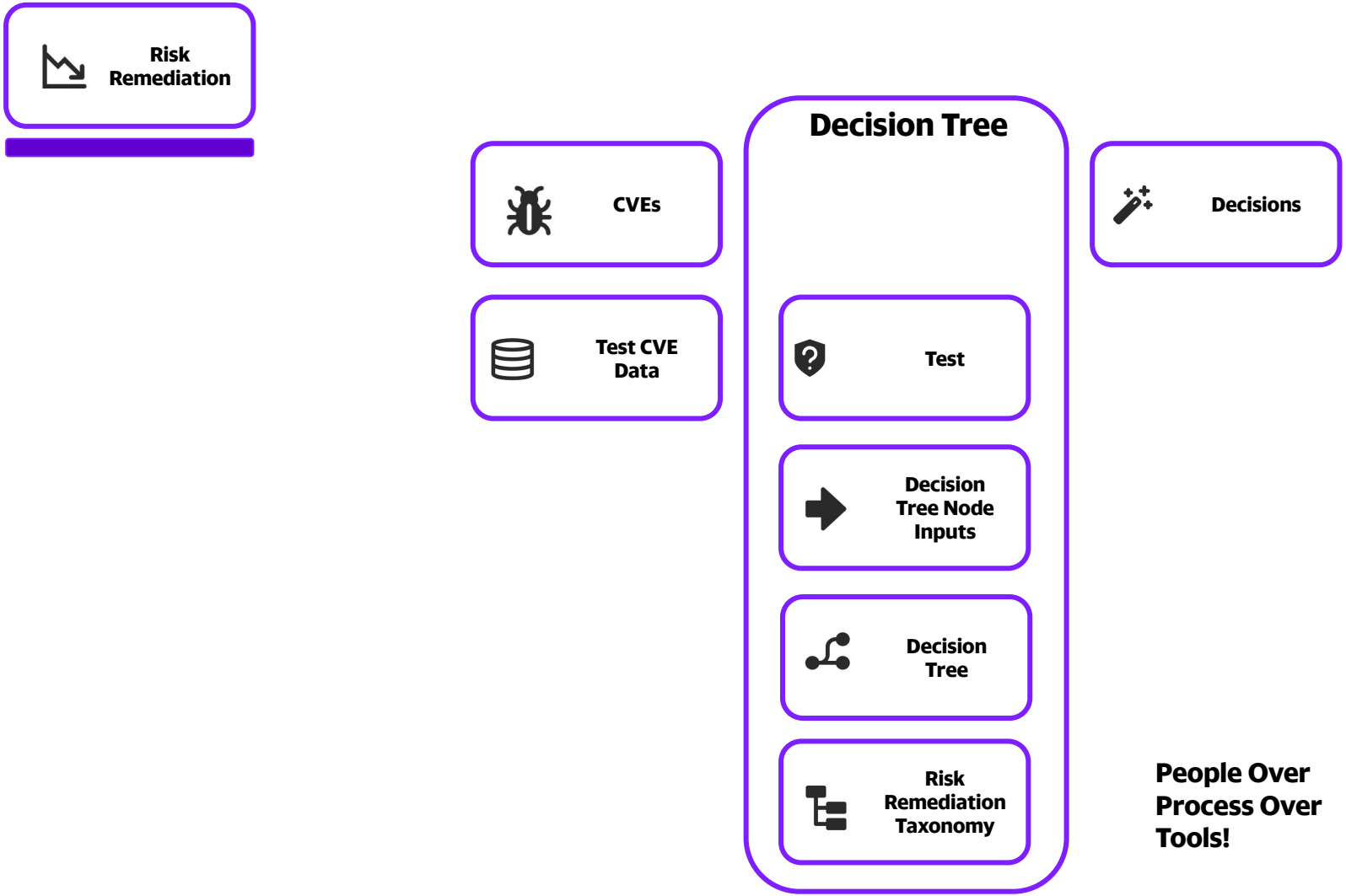
```
import pandas as pd
from pandas gui import show

df = pd.read_csv("./data.csv")
show (df)
```

https://github.com/adamerose/pandasgui

EDA is a quick, easy, first-step way to understand your data

Risk Remediation




Risk Remediation
CVEs
Test CVE Data
Decision Tree
Test
Decision Tree Node Inputs
Decision Tree
Risk Remediation Taxonomy
Decisions
People Over Process Over Tools!

Risk Remediation

Risk Remediation Taxonomy

Risk = Threat x Vulnerability x Impact

This isn't a mathematical formula or exact association - this is showing the different components of risk

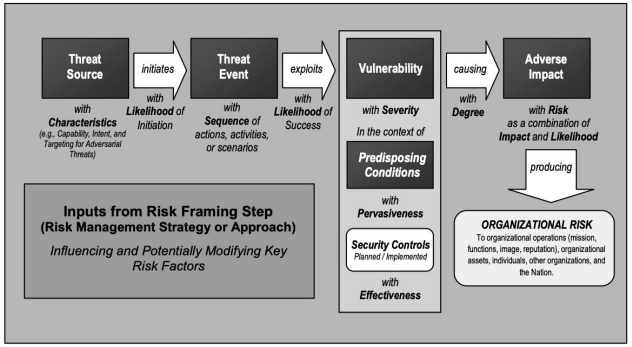


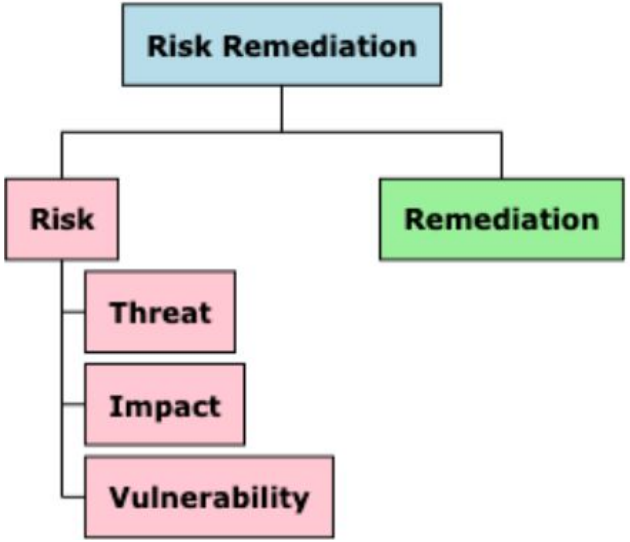
FIGURE 3: GENERIC RISK MODEL WITH KEY RISK FACTORS

- RISK A measure of the extent to which an entity is threatened by a potential circumstance or event, and typically is a function of:
  - (i) the adverse impact, or magnitude of harm, that would arise if the circumstance or event occurs; and
  - (ii) the likelihood of occurrence.
- Threat the potential for a threat-source to successfully exploit a particular information system vulnerability.
- Vulnerability Weakness in an information system, system security procedures, internal controls, or implementation that could be exploited by a threat source
- Impact The magnitude of harm that can be expected to result from the consequences of unauthorized disclosure of information, unauthorized modification of information, unauthorized destruction of information, or loss of information or information system availability.
- Asset The data, personnel, devices, systems, and facilities that enable the organization to achieve business purposes.

Risk is per Asset and depends on Impact of a Vulnerability being exploited by a Threat

NIST Special Publication 800-30 r1 Guide for Conducting Risk Assessments https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-30r1pdf

Risk Remediation Taxonomy



Understanding Risk is only half the picture. The full picture is Risk Remediation.

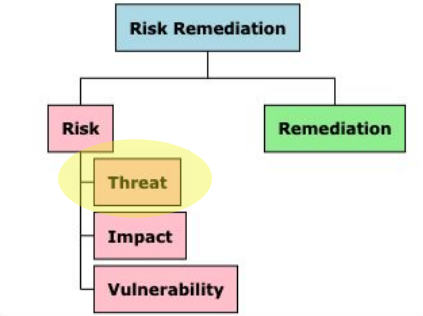
Risk Remediation

Risk Remediation Taxonomy

Threat

Threat the potential for a threat-source to successfully exploit a particular information system vulnerability.

Risk Remediation Taxonomy for CVEs



Icon	Description
	Internal
	Open and available to anyone
	Paid product
	Manual
	Relevant values defined in a standard

Counts of CVE-IDS order of magnitude

10^3 CISA KEV

10^4 EPSS > 0.1

10^5

- **Threat**
  - **Likelihood of Exploit**
    - **Known Actively Exploited in your Organization**
      - **Bug Bounty** has the CVE been exploited?
      - **Incident Response** has the CVE been exploited?
      - **Incident Response** has the Asset been attacked?
    - **Known Actively Exploited in the Wild**
      - **CISA Known Exploited Vulnerabilities (CISA KEV)**
      - **Vendor DBs** (typicaly include CISA KEV data)
      - **Cyber Threat Intelligence (CTI)** e.g. social media
    - **Probability of Exploitation Activity in the next 30 days %** (EPSS Exploit Prediction Scoring System v3)
      - **Exploitation activity in the wild (ground truth)** Fortinet, AlienVault, ShadowServer, GreyNoise
      - **Publicly available exploit code** ExploitDB, GitHub, MetaSploit
      - CVE listed on CISA KEV, Google Project Zero, Trend Micro's Zero Day Initiative
      - **Social Media** Mentions/discussion on Twitter
      - **Offensive security tools and scanners** Intrigue, sn1per, jaeles, nuclei
      - **References with labels** MITRE CVE List, NVD
      - **Keyword description of the vulnerability** Text description in MITRE CVE List
      - **CVSS metrics** National Vulnerability Database (NVD)
      - **CWE** National Vulnerability Database (NVD)
      - **Vendor labels** National Vulnerability Database (NVD)
      - **Age of the vulnerability** Days since CVE published in MITRE CVE list
    - **Known Exploit Available**
      - **Vendor DBs** which include data about (weaponized) vulnerabilities from tools like Metasploit
      - **Cyber Threat Intelligence (CTI)** e.g. social media
      - **Exploit Code Maturity**
        - **Vendor DBs**
          - Not Defined, Unproven, Proof-Of-Concept, Functional, High [CVSS Temporal]
          - e.g. public, poc_public, virus_malware, wormified, private, commercial, exploit_unknown, unknown
    - **Frequency of Exploit (VendorDB)**
    - **Average Time to Exploit Product (VendorDB)**
  - **System Exposure**
    - **Asset Inventory**
      - Small, Controlled, Open [CMU SEI SSVC]
  - **Report Confidence**
    - Not Defined, Unknown, Confirmed, Reasonable [CVSS Temporal]

"Likelihood of Exploit" branches listed in Order of Importance

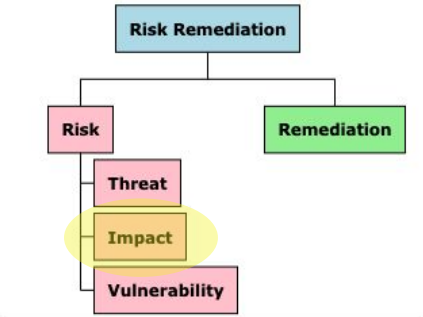
Risk Remediation

Risk Remediation Taxonomy

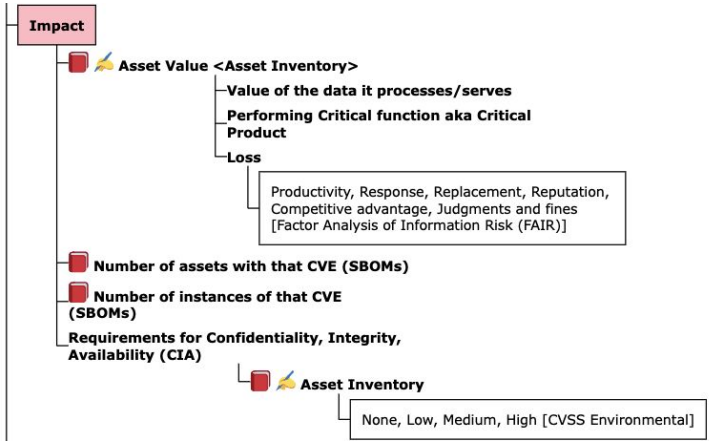
Impact

Impact The magnitude of harm that can be expected to result from the consequences of unauthorized disclosure of information, unauthorized modification of information, unauthorized destruction of information, or loss of information or information system availability.

Risk Remediation Taxonomy for CVEs



Icon	Description
	Internal
	Open and available to anyone
	Paid product
	Manual
	Relevant values defined in a standard



Impact Depends on Your Organization Context

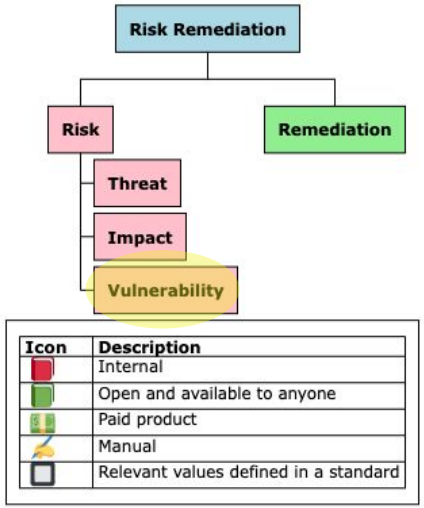
Risk Remediation

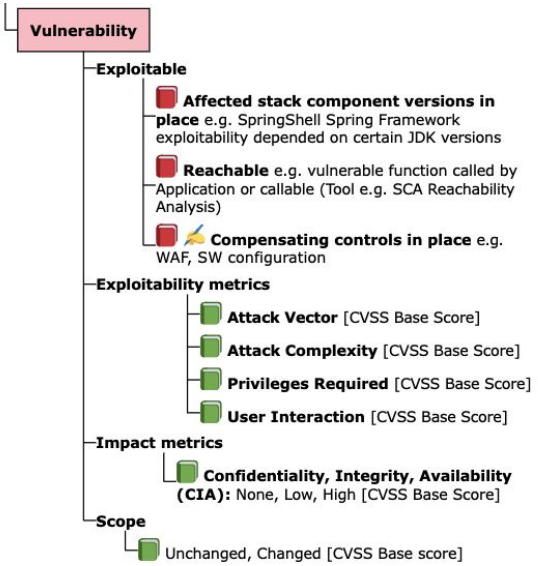
Risk Remediation Taxonomy

Vulnerability

Vulnerability Weakness in an information system, system security procedures, internal controls, or implementation that could be exploited by a threat source

Risk Remediation Taxonomy for CVEs

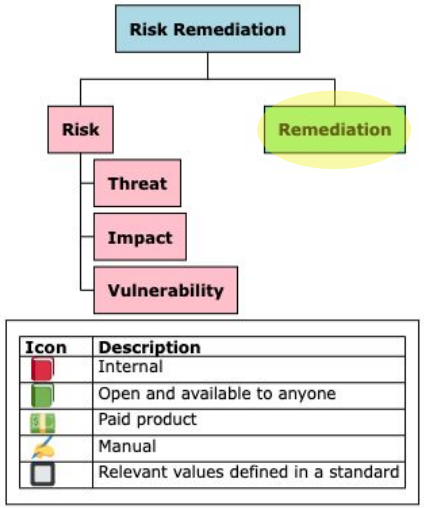


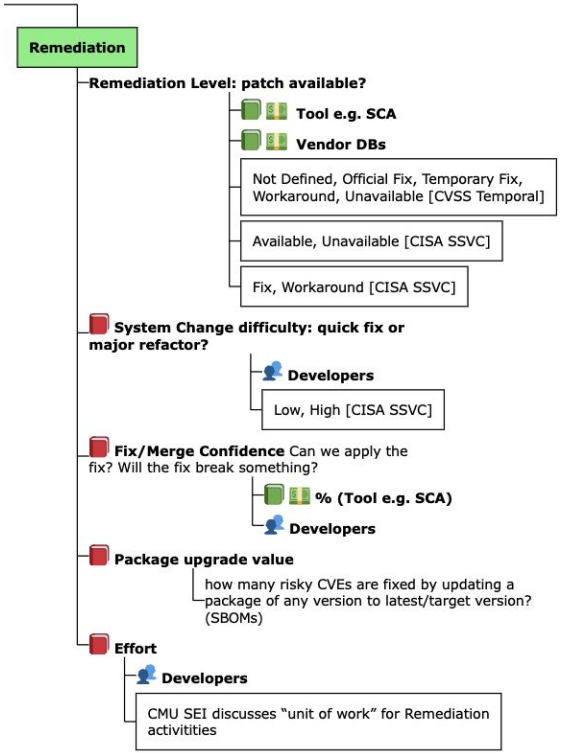


Exploitable Depends on Runtime Context e.g. not Called/Reachable

Remediation

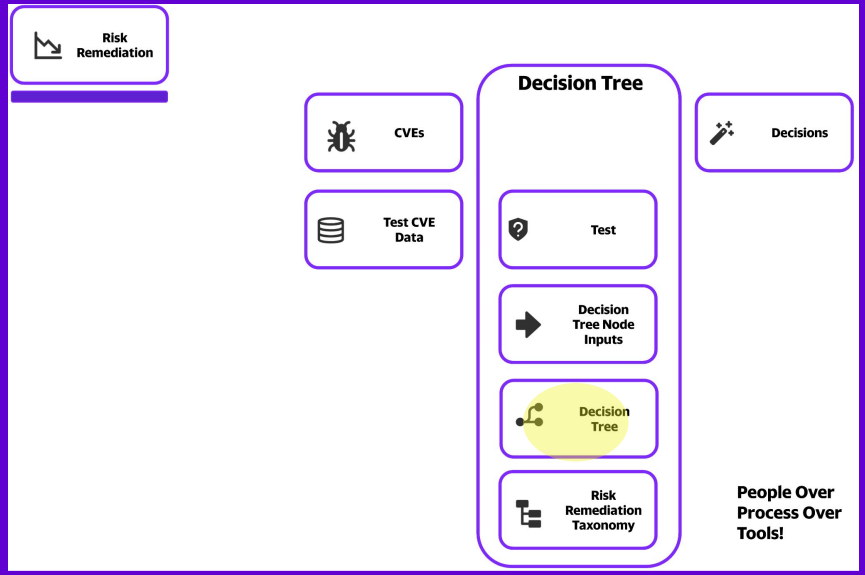
Risk Remediation Taxonomy for CVEs





Remediation Depends on Your Development Context

Decision Trees



Risk Remediation | **Decision Tree**

Why Decision Trees?

1. Focus on what matters: risk and its constituent components and what action needs to be taken when
2. Understandable.
3. Modular: e.g. allows change/customization of Mission & Well-being Decision Node for an organization. Loose coupling, high cohesion.
4. Decision Tree Analysis can be applied
5. Trees gives a very clear visual of all the parameters and decision nodes e.g. Attack Trees for Threat Modeling. Formulas are opaque, single output.

Creating Decision Trees

1. Good presentation on DTs for SSVC
   https://www.first.org/resources/papers/conf2022/121_04-PrioritizingVulnerability-Spring.pdf
2. https://democert.org/ssvc/ has different Decision Trees per Role (as defined by https://vuls.cert.org/confluence/display/CVD/3.+Roles+in+CVD
   a. *"Organizations using a DevOps approach to providing services might have a single group responsible for both the supplier and deployer roles"*

CISA Coordinator Decision Tree

Whether one uses Decision Trees or not, a LOT of the benefit is calling out the factors e.g. Exploitation, Automatable,... and the associated levels. It forces one to then define the parameters that contribute to those nodes (loose coupling, high cohesion).

This is in stark contrast to a formula e.g. "if CVSS >= 7 and Confidentiality Impact is High, then....".
While this is syntactically very easy to understand, semantically it's very difficult i.e.

- what was the author try to achieve in terms of risk?
- what is the set of permutations of CVSS base score parameters that make this very simple equation true?

https://phoenix.security/risk-based-priortiy-decision-tree/

Risk Remediation

Decision Tree

CISA SSVC Decision Tree

Exploitation

None
There is no evidence of active exploitation and no public proof of concept (PoC) of how to exploit the vulnerability.

Poc
One of the following cases is true: (1) private evidence of exploitation is attested but not shared; (2) widespread hearsay attests to exploitation; (3) typical public PoC in places such as Metasploit or ExploitDB; or (4) the vulnerability has a well-known method of exploitation. Some examples of condition (4) are open-source web proxies serve as the PoC code for how to exploit any vulnerability in the vein of improper validation of TLS certificates. As another example, Wireshark serves as a PoC for packet replay attacks on ethernet or WiFi networks.

Active
Shared, observable, reliable evidence that the exploit is being used in the wild by real attackers; there is credible public reporting.

Automatable

No
Steps 1-4 of the kill chain cannot be reliably automated for this vulnerability for some reason. These steps are reconnaissance, weaponization, delivery, and exploitation. Example reasons for why a step may not be reliably automatable include (1) the vulnerable component is not searchable or enumerable on the network, (2) weaponization may require human direction for each target, (3) delivery may require channels that widely deployed network security configurations block, and (4) exploitation may be frustrated by adequate exploit-prevention techniques enabled by default; ASLR is an example of an exploit-prevention tool.

Yes
Steps 1-4 of the of the kill chain can be reliably automated. If the vulnerability allows unauthenticated remote code execution (RCE) or command injection, the response is likely yes.

Technical Impact

Partial
The exploit gives the adversary limited control over, or information exposure about, the behavior of the software that contains the vulnerability. Or the exploit gives the adversary an importantly low stochastic opportunity for total control. In this context, "low" means that the attacker cannot reasona-bly make enough attempts to overcome the low chance of each attempt not working. Denial of service is a form of limited control over the behavior of the vulnerable component.

Total
The exploit gives the adversary total control over the behavior of the software, or it gives total disclosure of all information on the system that contains the vulnerability.

Mission & Well-being
(Complex Decision)

Low
Mission Prevalence is Low and Public well-being impact is Minimal

Medium
Mission Prevalence is Medium and Public well-being impact is in Material

High
Mission Prevalence is Essential and Public well-being impact is Irreversible

Depends on 1

Mission Prevalence

Minimal
Neither support nor essential apply. The vulnerable component may be used within the entities, but it is not used as a mission-essential component nor does it support (enough) mission essential functions.

Support
The operation of the vulnerable component merely supports mission essential functions for two or more entities.

Essential
The vulnerable component directly provides capabilities that constitute at least one MEF for at least one entity, and failure may (but need not) lead to overall mission failure.

Depends on 2

Public Well-being Impact

Minimal
Type of harm is "All" (Physical, Environmental,Financial,Psychological). The effect is below the threshold for all aspects described in material.

Material
Any one or more of the conditions (Physical, Environmental,Financial,Psychological) hold. "Physical harm" means "Physical distress or injuries for users of the system OR introduces occupational safety hazards OR reduction and/or failure of cyber-physical system's safety margins." "Environment" means "Major externalities (property damage, environmental damage, etc.) imposed on other parties." "Financial" means "Financial losses that likely lead to bankruptcy of multiple persons." "Psychological" means "Widespread emotional or psychological harm, sufficient to be cause for counselling or therapy, to populations of people."

Irreversible
Any one or more of the following conditions hold. "Physical harm" means "Multiple fatalities likely OR loss or destruction of cyber-physical system of which the vulnerable component is a part." "Environment" means "Extreme or serious externalities (immediate public health threat, environmental damage leading to small ecosystem collapse, etc.) imposed on other parties." "Financial" means "Social systems (elections, financial grid, etc.) supported by the software are destabilized and potentially collapse."

Decision

Track The vulnerability does not require attention outside of Vulnerability Management (VM) at this time. Continue to track the situation and reassess the severity of vulnerability if necessary.

Track* Track these closely, especially if mitigation is unavailable or difficult. Recommended that analyst discuss with other ana-lysts and get a second opinion.

Attend The vulnerability requires to be attended to by stakeholders outside VM. The action is a request to others for assistance / information / details, as well as a potential publication about the issue.

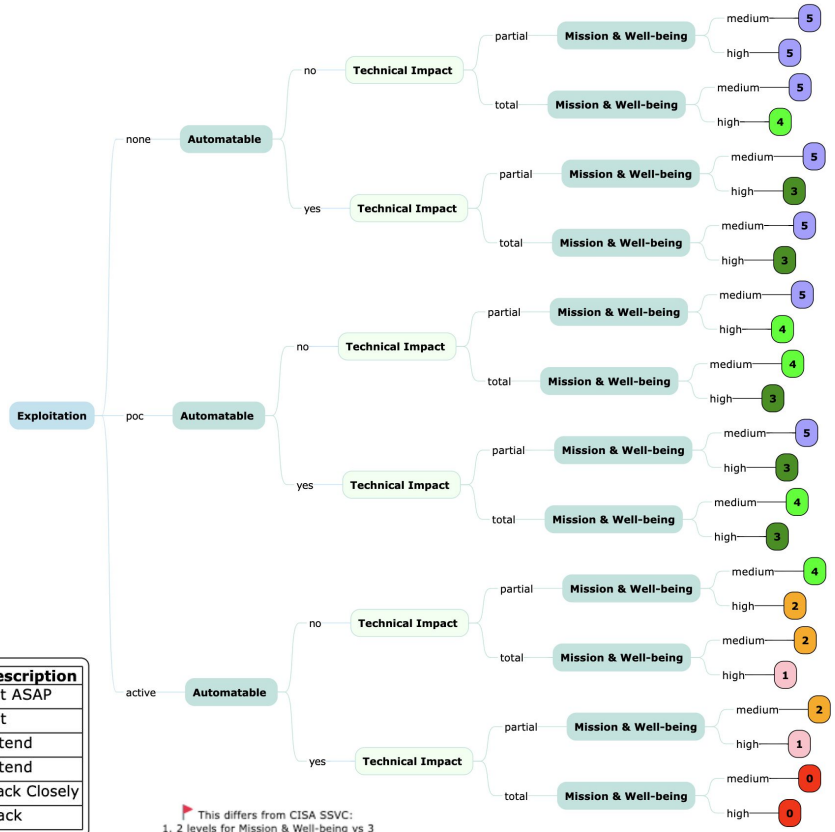
Act The vulnerability requires immediate action by the relevant leadership. The action is a high-priority meeting among the relevant supervisors to decide how to respond.

Cybersecurity and Infrastructure Security Agency (CISA) Stakeholder-Specific Vulnerability Categorization (SSVC)

The value of mitigation does not change the priority of the SSVC decision

Risk Remediation | **Decision Tree**

Risk-Based Prioritization Decision Tree



Number	Description
0	Act ASAP
1	Act
2	Attend
3	Attend
4	Track Closely
5	Track

This differs from CISA SSVC:
1. 2 levels for Mission & Well-being vs 3
2. Additional Decision for highest priority

Decision

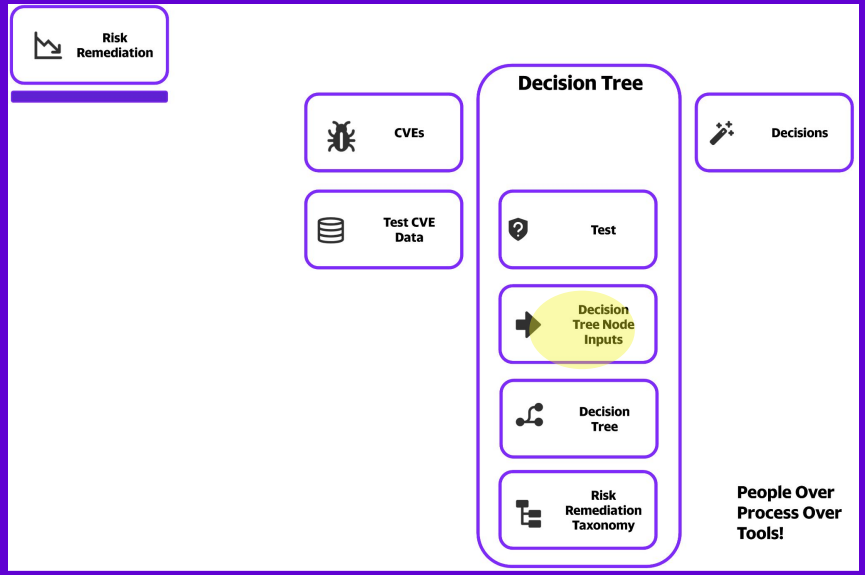
Track The vulnerability does not require attention outside of Vulnerability Management (VM) at this time. Continue to track the situation and reassess the severity of vulnerability if necessary.

Track* Track these closely, especially if mitigation is unavailable or difficult. Recommended that analyst discuss with other ana-lysts and get a second opinion.

Attend The vulnerability requires to be attended to by stakeholders outside VM. The action is a request to others for assistance / information / details, as well as a potential publication about the issue.

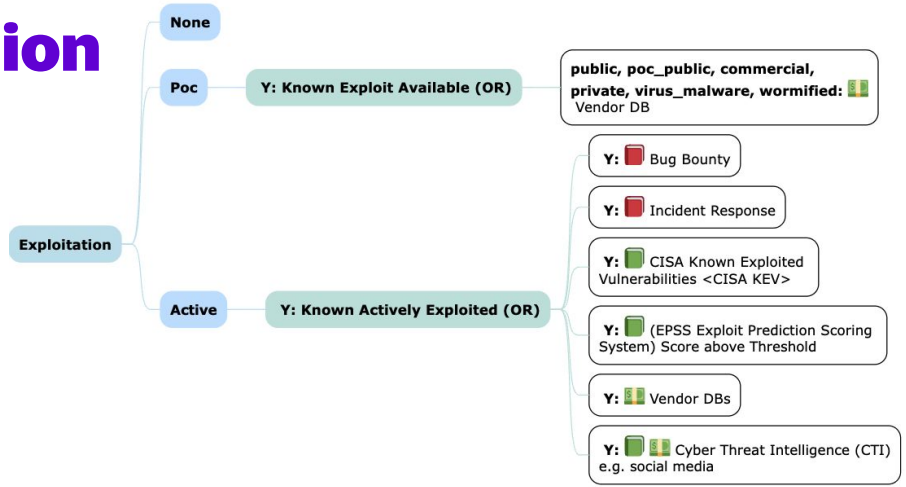
Act The vulnerability requires immediate action by the relevant leadership. The action is a high-priority meeting among the relevant supervisors to decide how to respond.

Decision Tree Node Inputs



Risk Remediation | **Decision Tree** | **Decision Tree Node Inputs**

Exploitation



Icon	Description
	Internal
	Open and available to anyone
	Paid product
	Manual
[ ]	Standard
< >	Data Source

Exploitation

None

There is no evidence of active exploitation and no public proof of concept (PoC) of how to exploit the vulnerability.

Poc

One of the following cases is true: (1) private evidence of exploitation is attested but not shared; (2) widespread hearsay attests to exploitation; (3) typical public PoC in places such as Metasploit or ExploitDB; or (4) the vulnerability has a well-known method of exploitation. Some examples of condition (4) are open-source web proxies serve as the PoC code for how to exploit any vulnerability in the vein of improper validation of TLS certificates. As another example, Wireshark serves as a PoC for packet replay attacks on ethernet or WiFi networks.

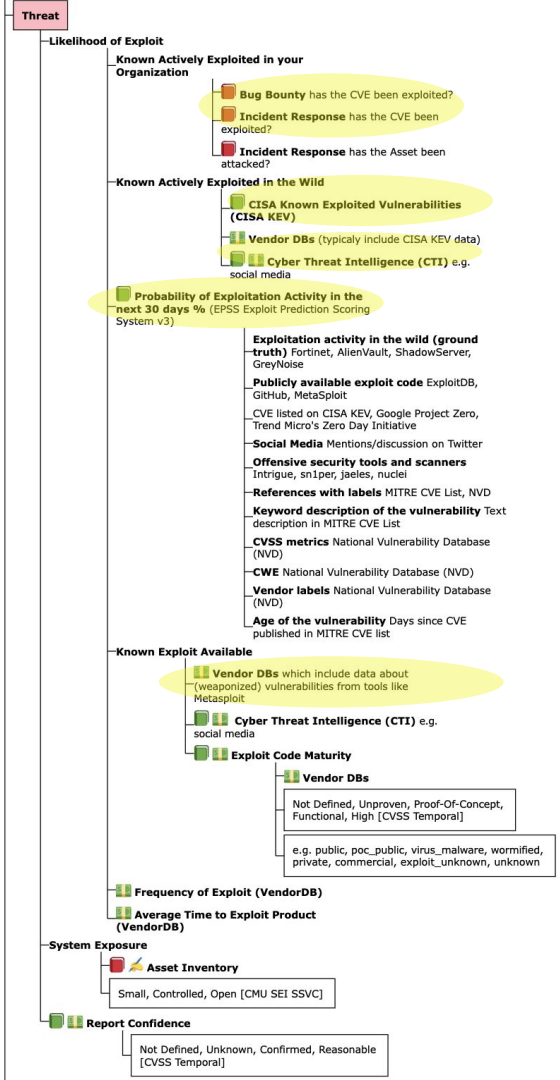
Active

Shared, observable, reliable evidence that the exploit is being used in the wild by real attackers; there is credible public reporting.

"EPSS could be used to inform the Exploitation decision point. Currently, Exploitation focuses on the observable state of the world at the time of the SSVC decision. EPSS is about predicting if a transition will occur from the SSVC state of none to active. A sufficiently high EPSS score could therefore be used as an additional criterion for scoring a vulnerability as active even when there is no observed active exploitation."

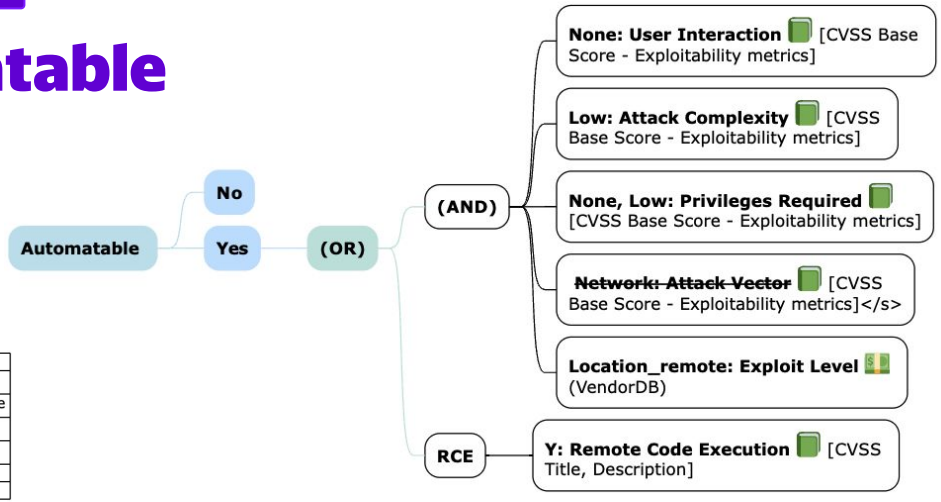
If a CVE is Known Actively Exploited, it should be prioritized even if it has a low EPSS

SEI CMU Prioritizing Vulnerability Response: A Stakeholder-Specific Vulnerability Categorization (Version 2.0)

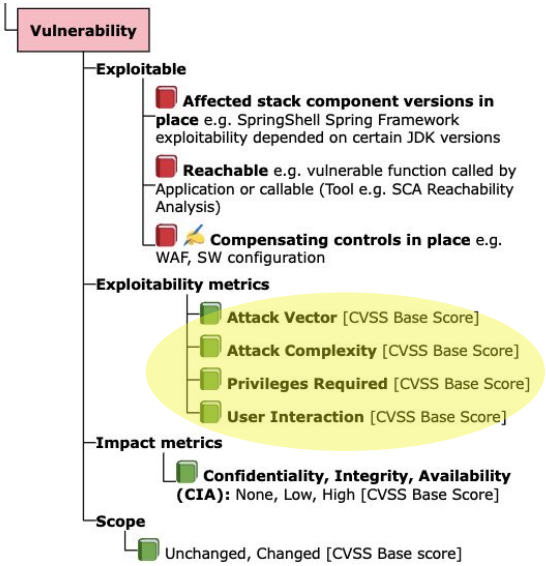


Risk Remediation | Decision Tree | Decision Tree Node Inputs

Automatable



Icon	Description
	Internal
	Open and available to anyone
	Paid product
	Manual
[ ]	Standard
< >	Data Source



Automatable

No

Steps 1-4 of the kill chain cannot be reliably automated for this vulnerability for some reason. These steps are reconnaissance, weaponization, delivery, and exploitation. Example reasons for why a step may not be reliably automatable include (1) the vulnerable component is not searchable or enumerable on the network, (2) weaponization may require human direction for each target, (3) delivery may require channels that widely deployed network security configurations block, and (4) exploitation may be frustrated by adequate exploit-prevention techniques enabled by default; ASLR is an example of an exploit-prevention tool.

Yes

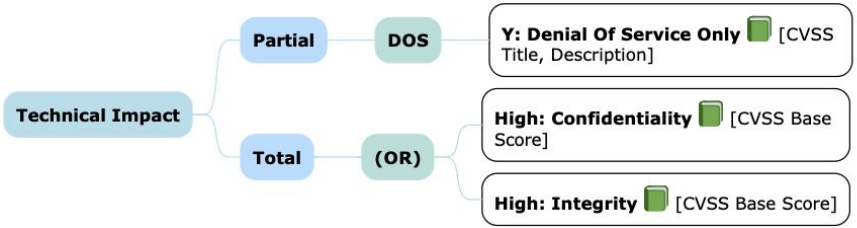
Steps 1-4 of the of the kill chain can be reliably automated. If the vulnerability allows unauthenticated remote code execution (RCE) or command injection, the response is likely yes.

"If the vulnerability allows unauthenticated remote code execution (RCE) or command injection, the response is likely yes."

SEI CMU Prioritizing Vulnerability Response: A Stakeholder-Specific Vulnerability Categorization (Version 2.0)

Risk Remediation | Decision Tree | Decision Tree Node Inputs

Technical Impact



Icon	Description
	Internal
	Open and available to anyone
	Paid product
	Manual
[ ]	Standard
< >	Data Source

Technical Impact

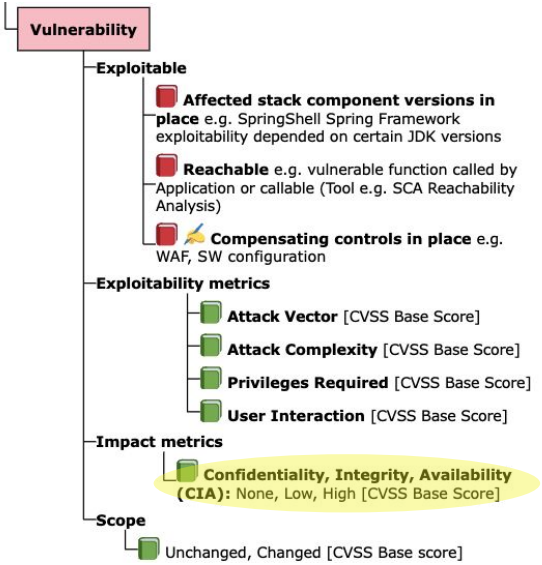
Partial

The exploit gives the adversary limited control over, or information exposure about, the behavior of the software that contains the vulnerability. Or the exploit gives the adversary an importantly low stochastic opportunity for total control. In this context, "low" means that the attacker cannot reasona-bly make enough attempts to overcome the low chance of each attempt not working. Denial of service is a form of limited control over the behavior of the vulnerable component.

Total

The exploit gives the adversary total control over the behavior of the software, or it gives total disclosure of all information on the system that contains the vulnerability.

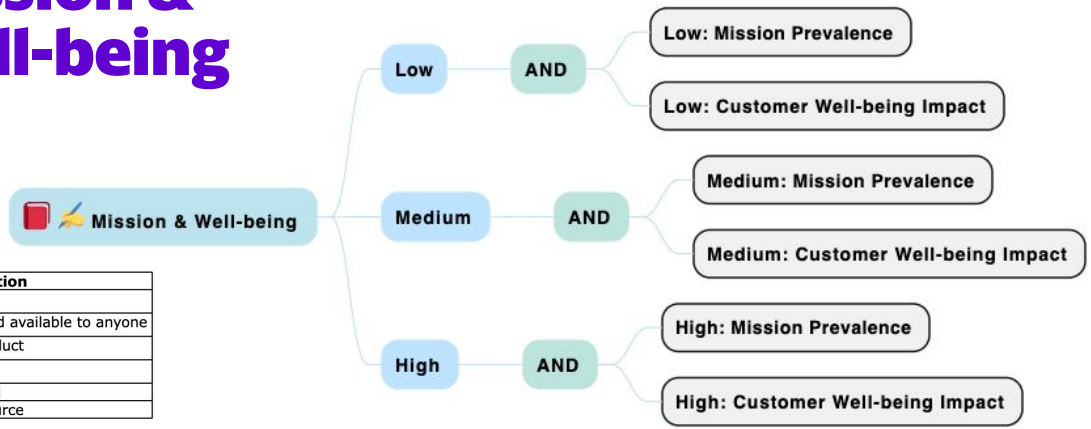
Technical impact: partial/total is decided regarding the system scope definition, which considers a database or a web server program as the "whole" system. Furthermore, "total" also includes any technical impact that exposes authentication credentials to the adversary, if those credentials are to the whole system.



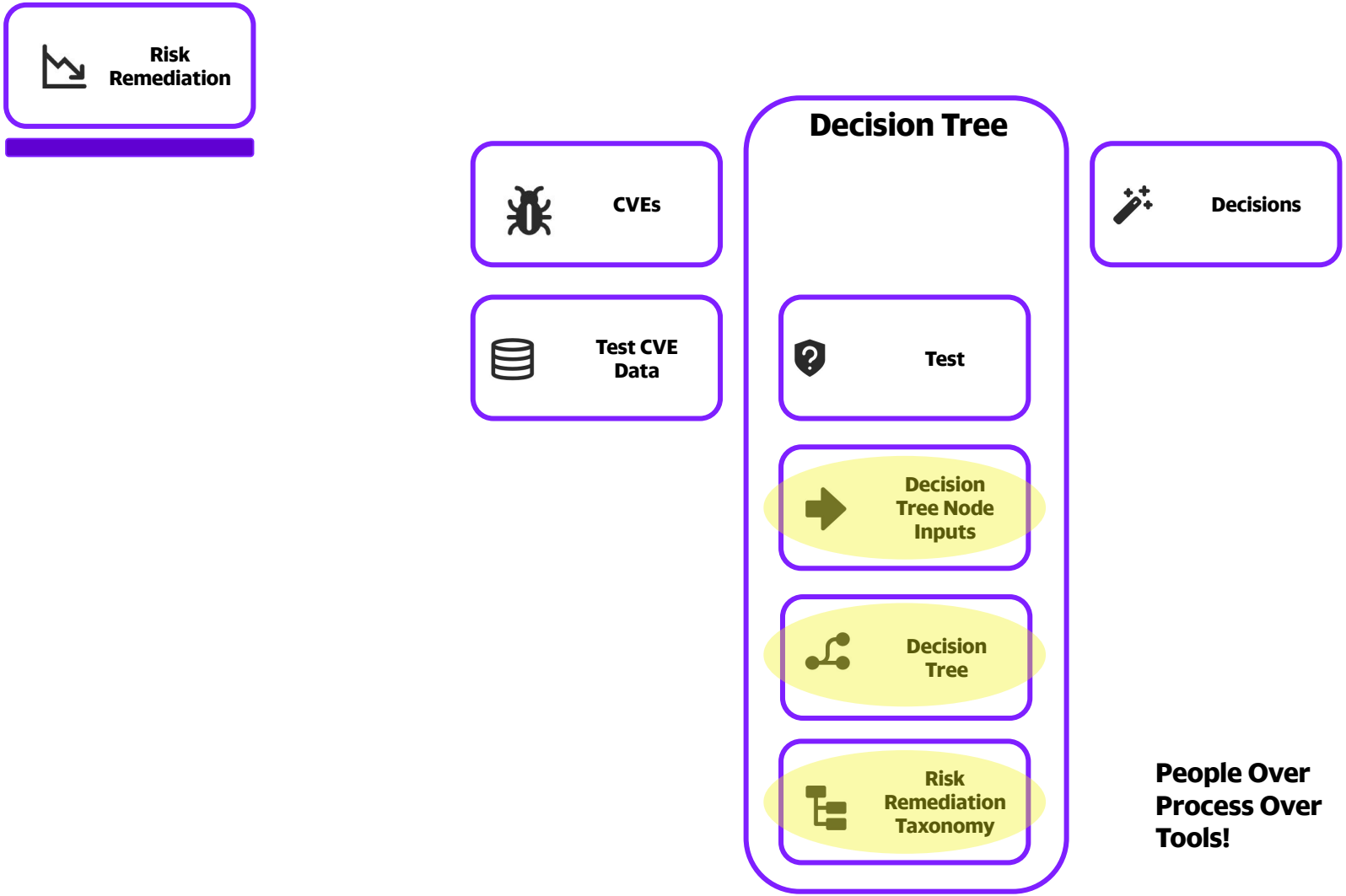
SEI CMU Prioritizing Vulnerability Response: A Stakeholder-Specific Vulnerability Categorization (Version 2.0)

Risk Remediation | Decision Tree | Decision Tree Node Inputs

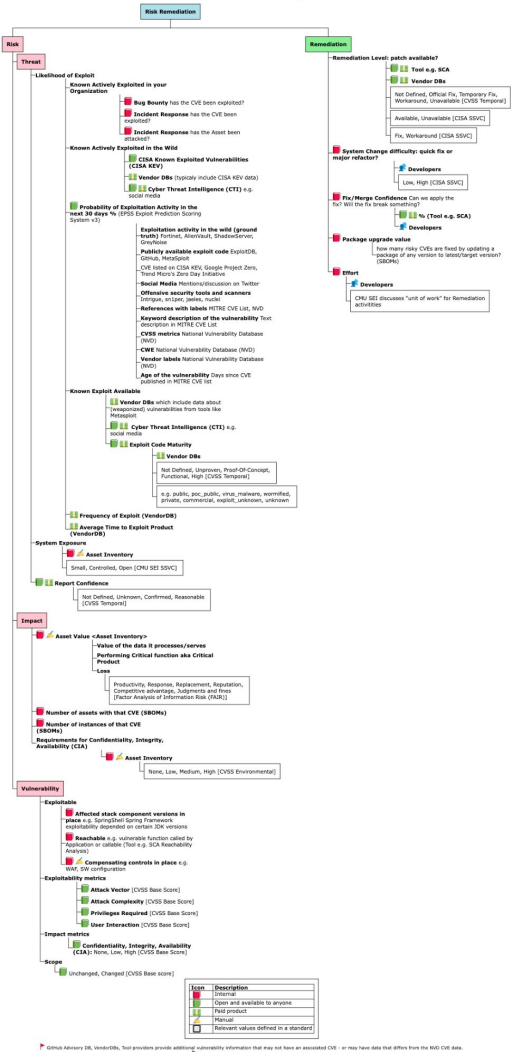
Mission & Well-being



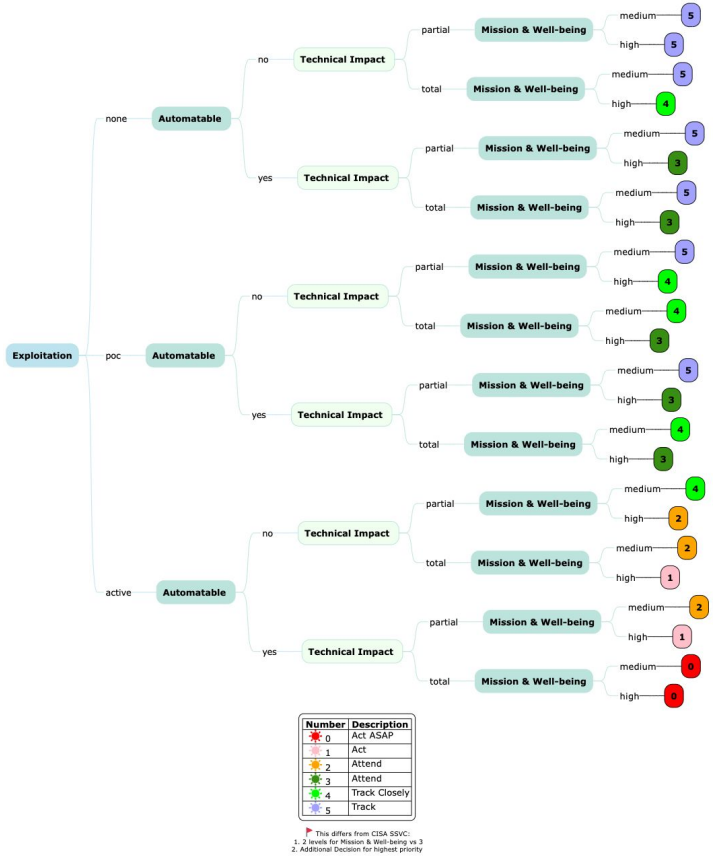
Icon	Description
	Internal
	Open and available to anyone
	Paid product
	Manual
[ ]	Standard
< >	Data Source


Risk Remediation
CVEs
Test CVE Data
Decision Tree
Test
Decision Tree Node Inputs
Decision Tree
Risk Remediation Taxonomy
Decisions
People Over Process Over Tools!

Risk Remediation Taxonomy for CVEs



Risk-Based Prioritization Decision Tree



Number	Description
0	Act ASAP
1	Act
2	Attend
3	Attend
4	Track Closely
5	Track

This differs from CISA SSVC:
1. 2 levels for Mission & Well-being vs 3
2. Additional Decision for highest priority

Decision Tree Decision Nodes Inputs

Diagrams & PlantUML source here as Open Source Software:
https://github.com/theparanoids/PrioritizedRiskRemediation

This is not intended to be a detailed flow-chart

So we just built a Risk-Based Prioritization Decision Tree...

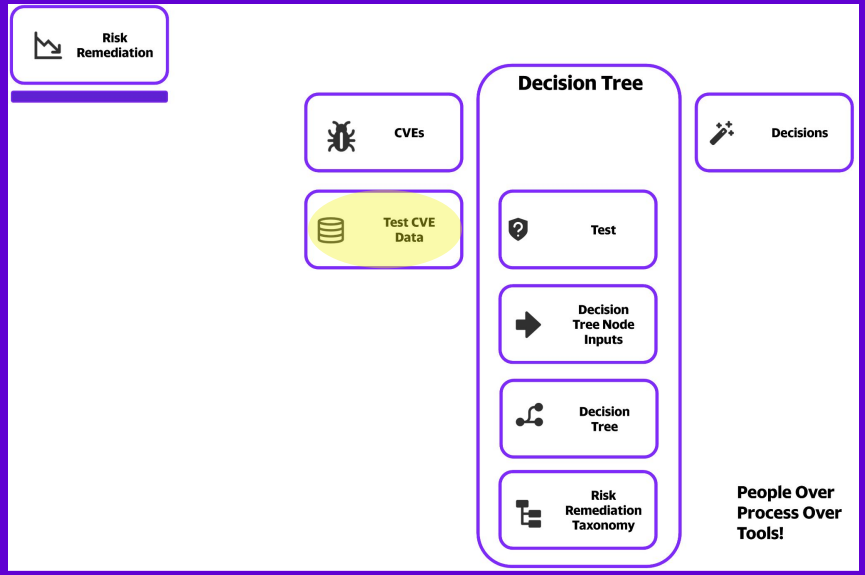
“Time to test our talents in the real world, d’you reckon?” Fred Weasley

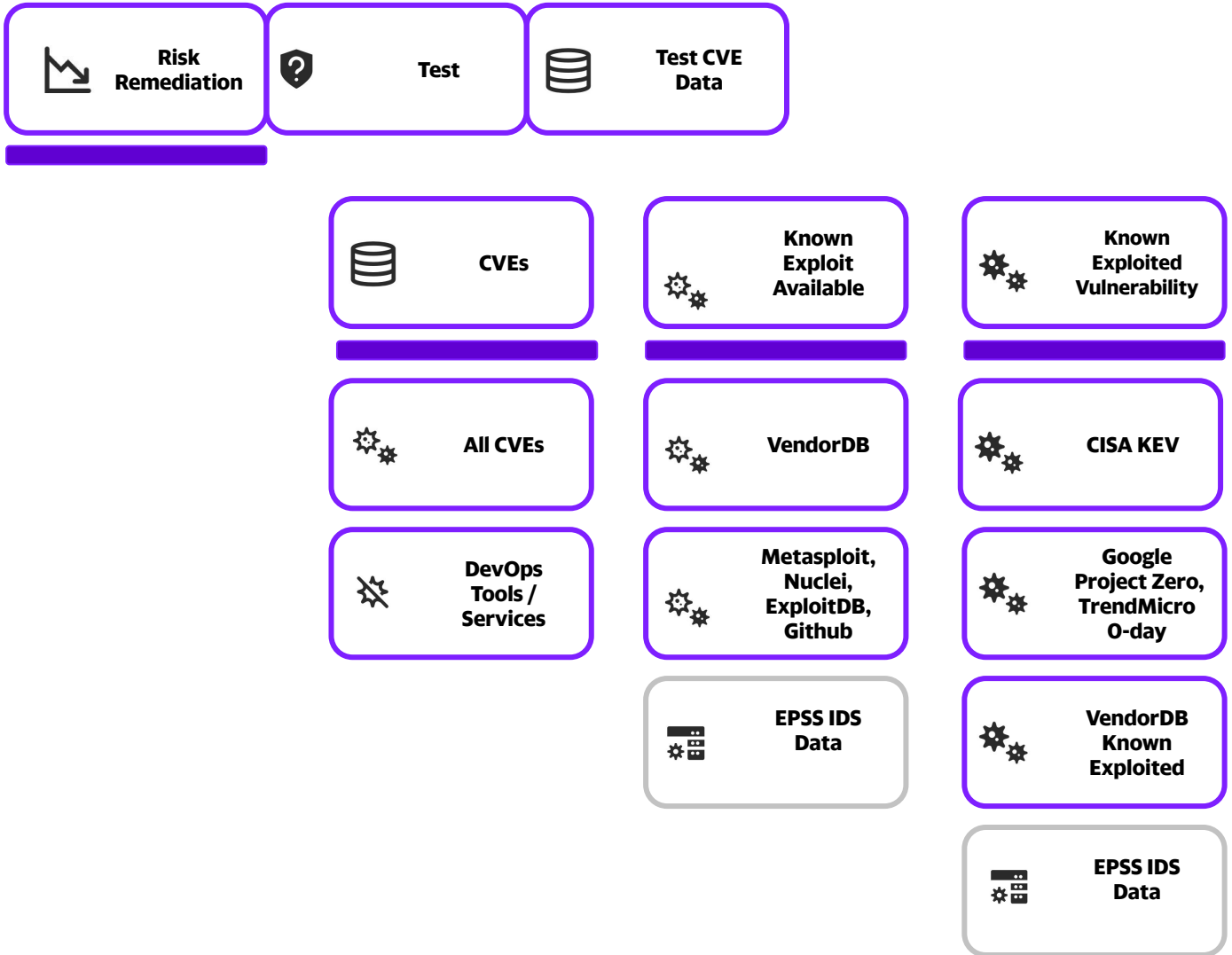
Image from https://www.funkidslive.com/activities/make-your-own-magic-wand/

For analysis purposes, assume that Mission & Well-being is “high” for all CVEs - and System Exposure is “Open”.

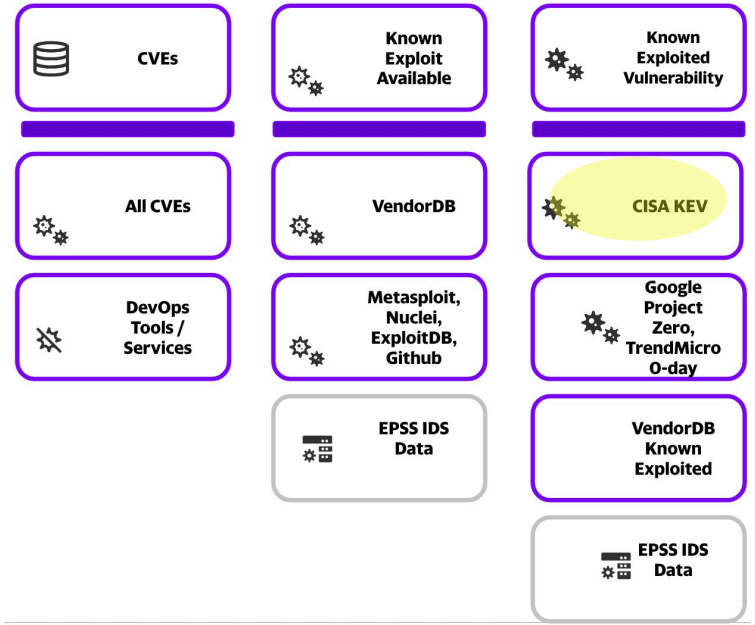
Test Decision Tree

Decision Trees with All Risk Parameters




Risk Remediation
Test
Test CVE Data
CVEs
Known Exploit Available
Known Exploited Vulnerability
All CVEs
VendorDB
CISA KEV
DevOps Tools / Services
Metasploit, Nuclei, ExploitDB, Github
Google Project Zero, TrendMicro 0-day
EPSS IDS Data
VendorDB Known Exploited
EPSS IDS Data

CISA KEV

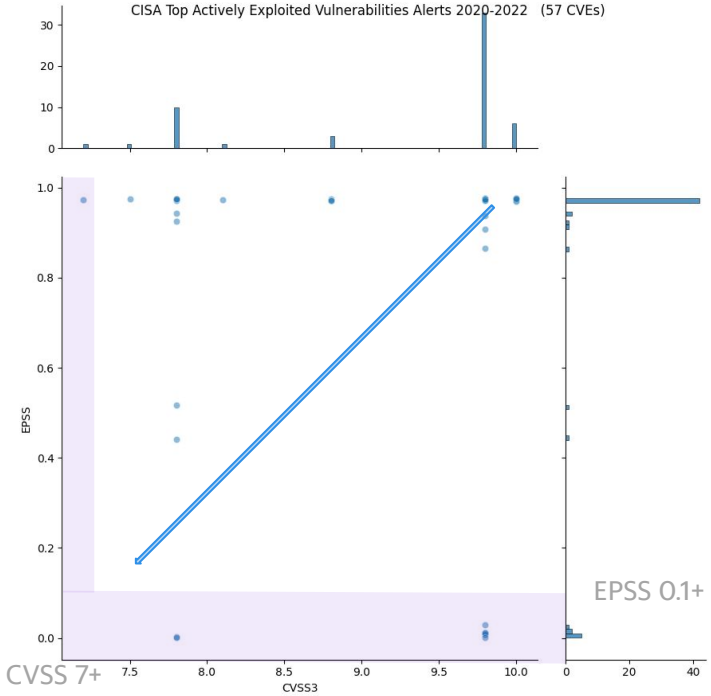

CVEs
Known Exploit Available
Known Exploited Vulnerability
All CVEs
VendorDB
CISA KEV
DevOps Tools / Services
Metasploit, Nuclei, ExploitDB, Github
Google Project Zero, TrendMicro 0-day
EPSS IDS Data
VendorDB Known Exploited
EPSS IDS Data

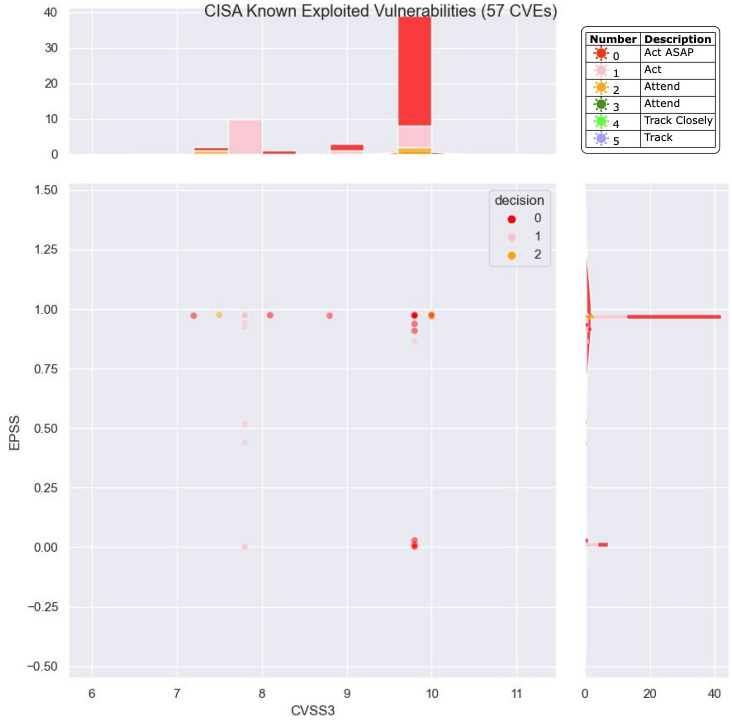
Risk Remediation | Test | Test CVE Data | Public Known Exploited CVE Data | CISA KEV Top Alerts

CVSS 7+ AND EPSS >= 0.1

Risk-Based Prioritization DT Applied

All CAVEs in the CISA KEV list, and CISA KEV "Top Routinely Exploited Cybersecurity Vulnerabilities" list per year, were known exploited (by definition).





Data Sources

1. CISA Top Routinely Exploited Vulnerabilities Alerts AA22-279A (2022), AA21-209A (2020-2021), AA22-117A (2021), AA20-133A (2016 to 2019). Some CVEs are duplicated across alerts.
2. EPSS

DT covers all CVEs (including those with low EPSS) - and prioritizes them via Decisions

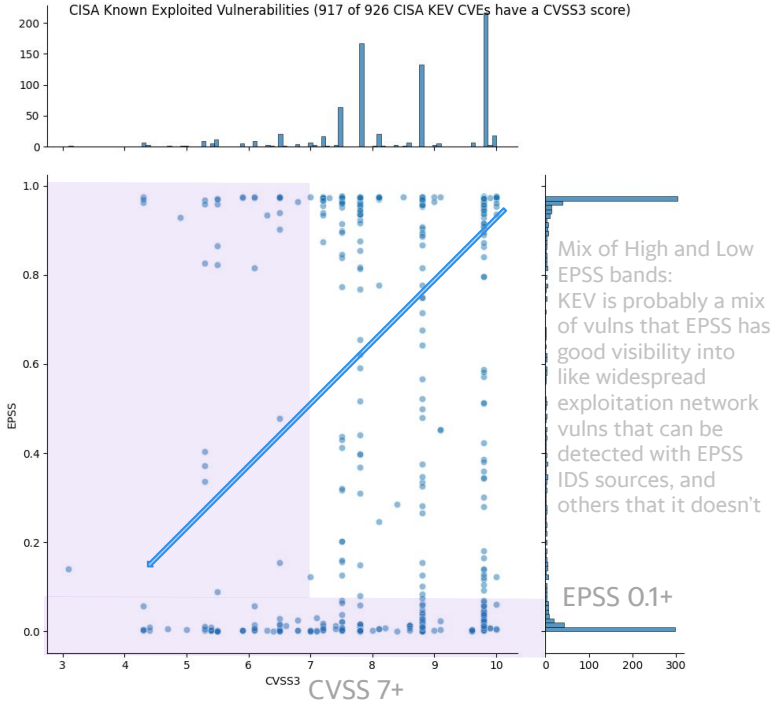
Risk Remediation | Test | Test CVE Data | Public Known Exploited CVE Data | CISA KEV

All CAVEs in the CISA KEV list, and CISA KEV "Top Routinely Exploited Cybersecurity Vulnerabilities" list per year, were known exploited (by definition).

Data Sources

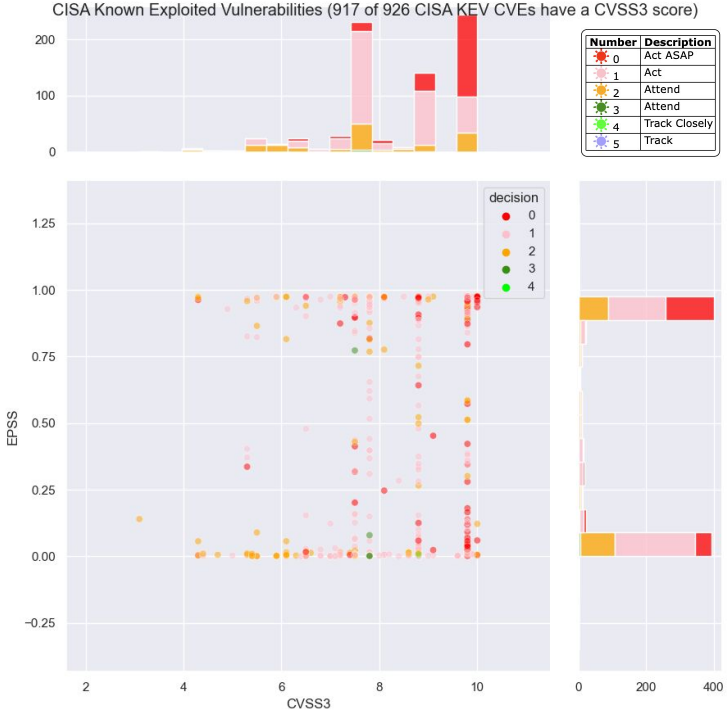
1. CISA Known Exploited Vulnerability catalog
2. EPSS

CVSS 7+ AND EPSS >= 0.1



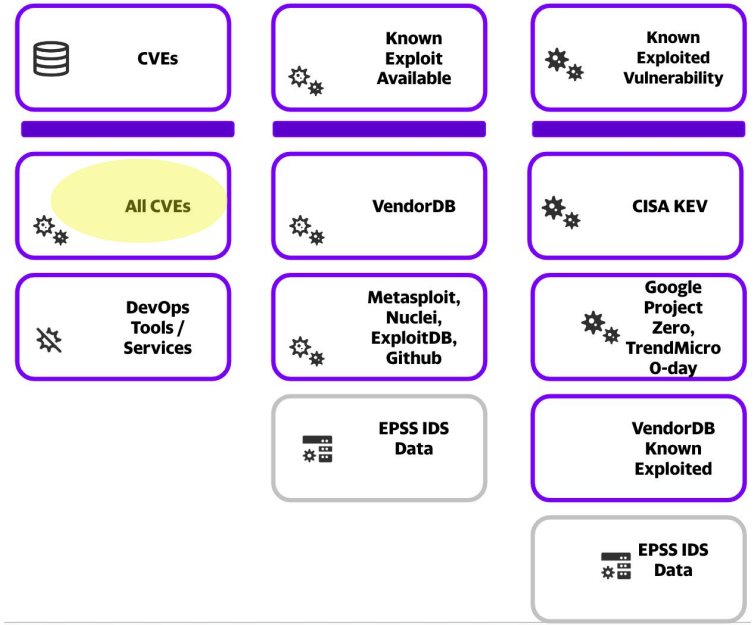
Mix of High and Low EPSS bands:
KEV is probably a mix of vulns that EPSS has good visibility into like widespread exploitation network vulns that can be detected with EPSS IDS sources, and others that it doesn't

Risk-Based Prioritization DT Applied



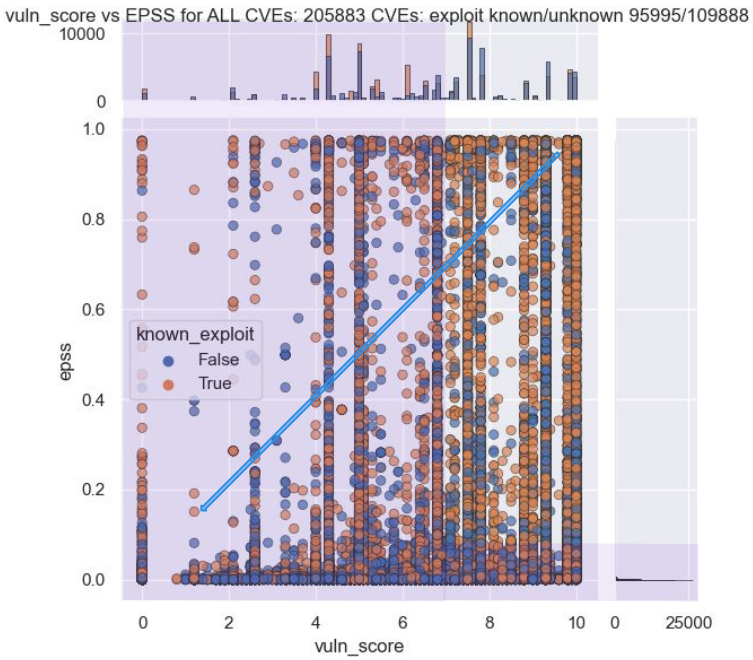
Our DT targets CVEs by highest risk vs prioritizing diagonally downwards

All CVEs

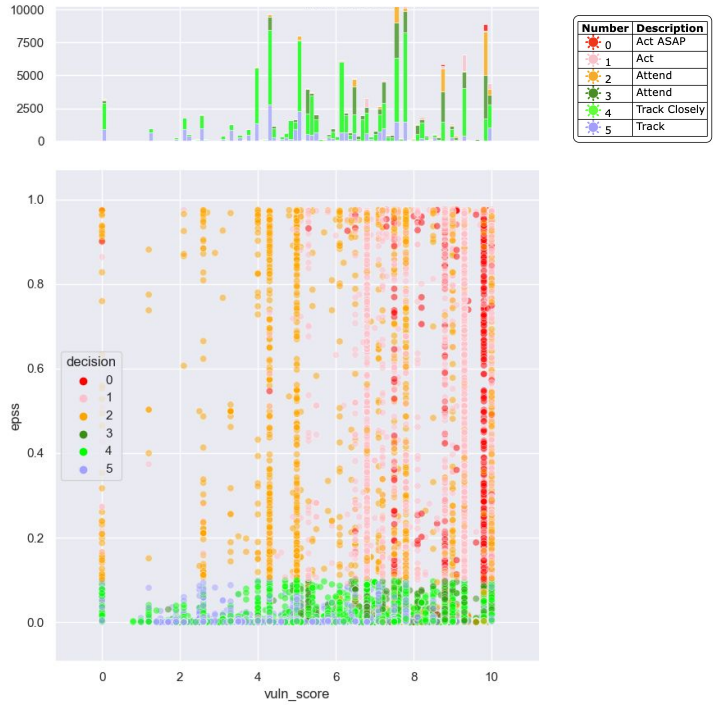

CVEs
Known Exploit Available
Known Exploited Vulnerability
All CVEs
VendorDB
CISA KEV
DevOps Tools / Services
Metasploit, Nuclei, ExploitDB, Github
Google Project Zero, TrendMicro 0-day
EPSS IDS Data
VendorDB Known Exploited
EPSS IDS Data

Risk Remediation | Test | Test CVE Data | Private Exploit CVE Data | VendorDB Known Exploit Available

CVSS 7+ AND EPSS >= 0.1



Risk-Based Prioritization DT Applied



Data Sources

1. NVD
2. EPSS
3. VendorDB for Known Exploit Available

Our DT targets CVEs by highest risk vs prioritizing diagonally downwards

vuln_score is CVSS V3 if it exists, else CVSS V2

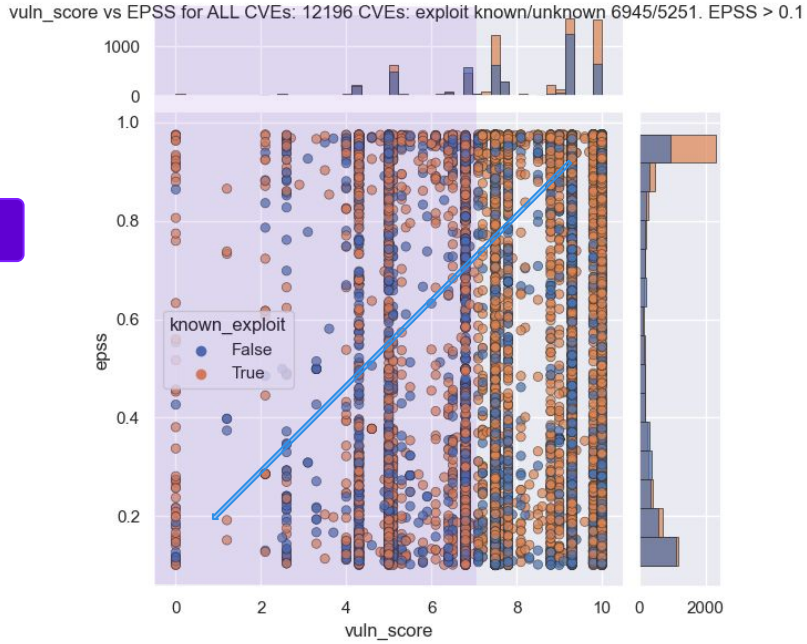
Risk Remediation | Test | Test CVE Data | Private Exploit CVE Data | VendorDB Known Exploit Available

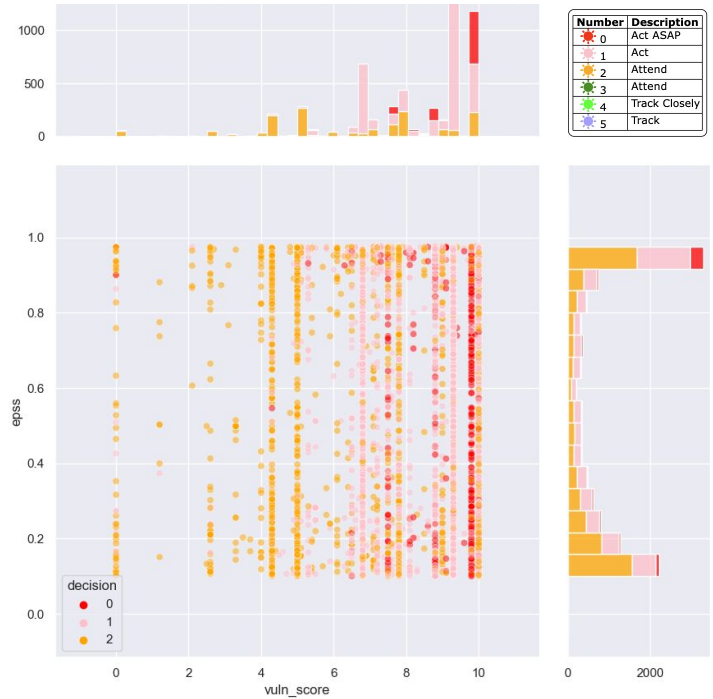
EPSS >= 0.1

to see upper detail

CVSS 7+ AND EPSS >= 0.1

Risk-Based Prioritization DT Applied





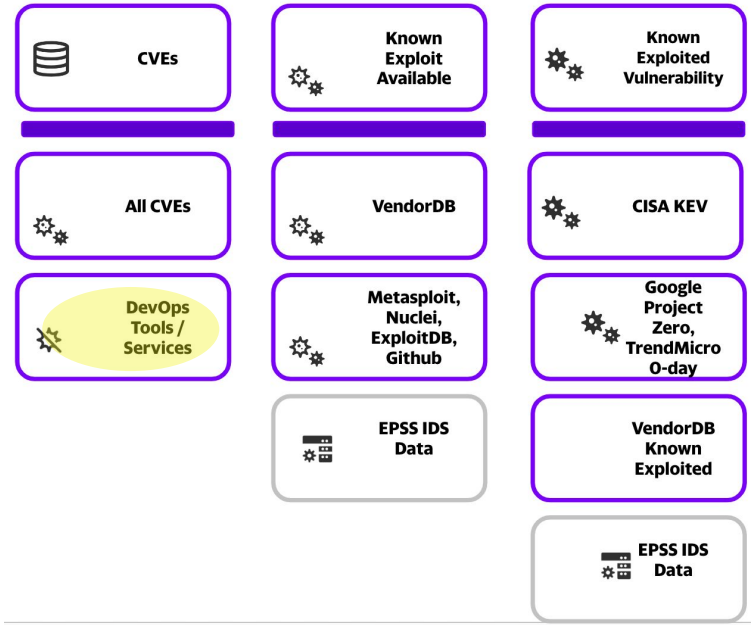
Data Sources

1. NVD
2. EPSS
3. VendorDB for Known Exploit Available

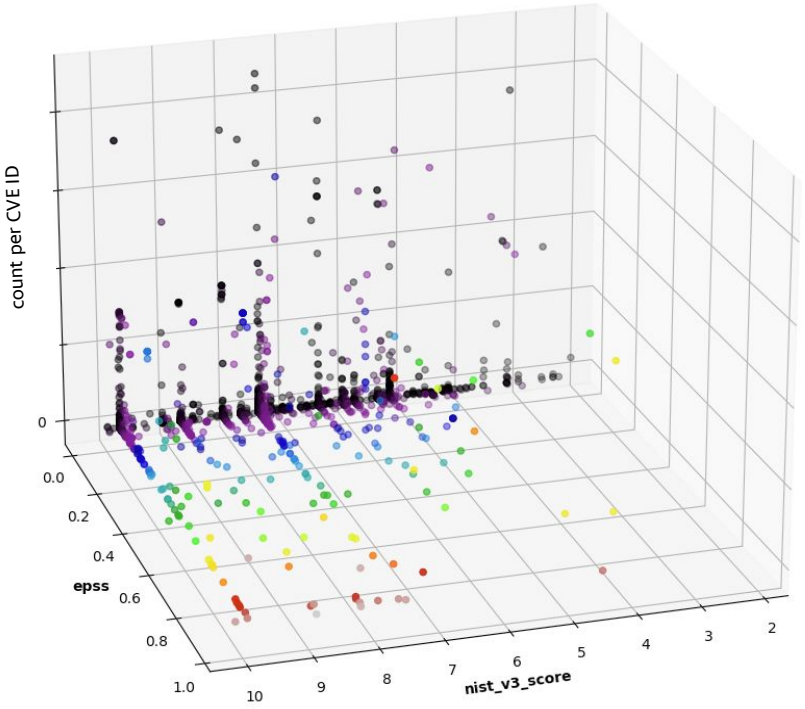
Our DT targets CVEs by highest risk vs prioritizing diagonally downwards

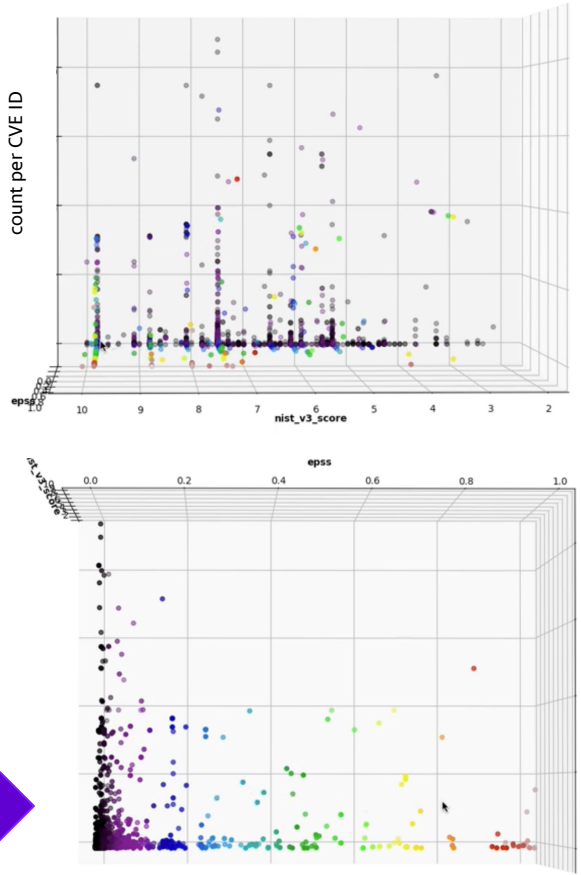
vuln_score is CVSS V3 if it exists, else CVSS V2

Internal DevOps



Risk Remediation | Test | Test CVE Data | Internal CVE Data | DevOps Tools / Services



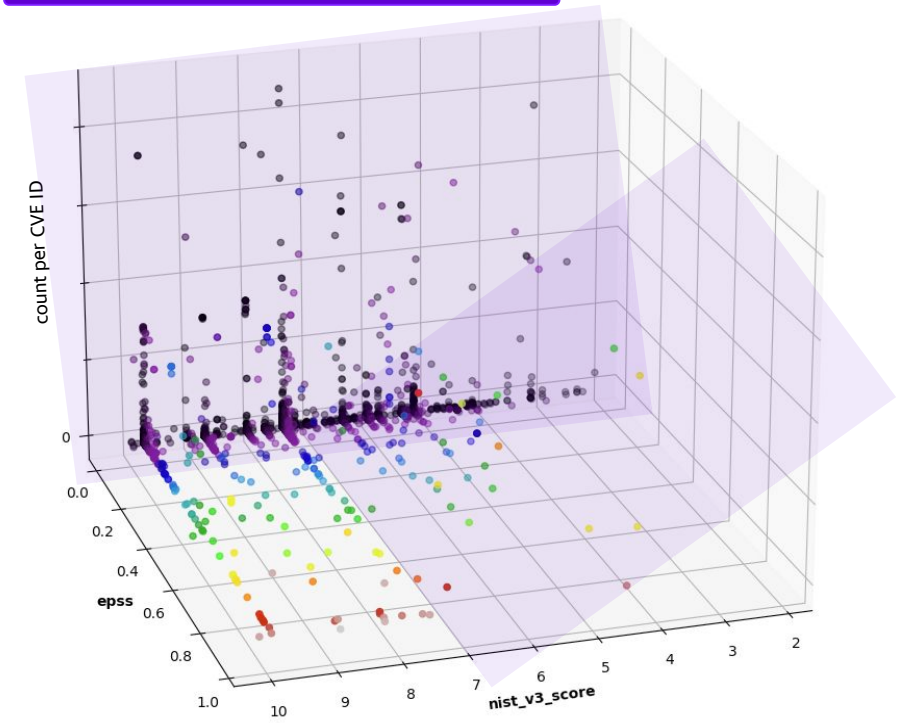


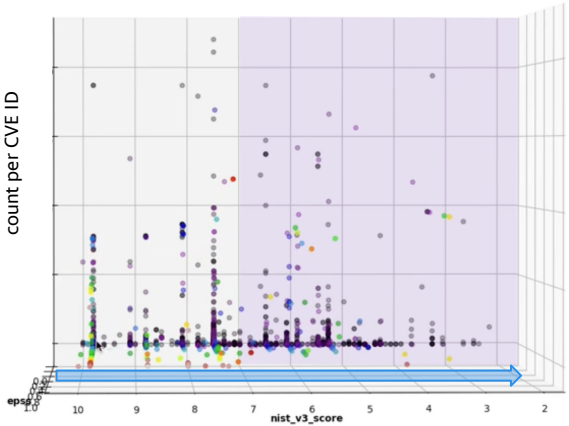
Many CVEs (across CVSS score range) have a low EPSS scores

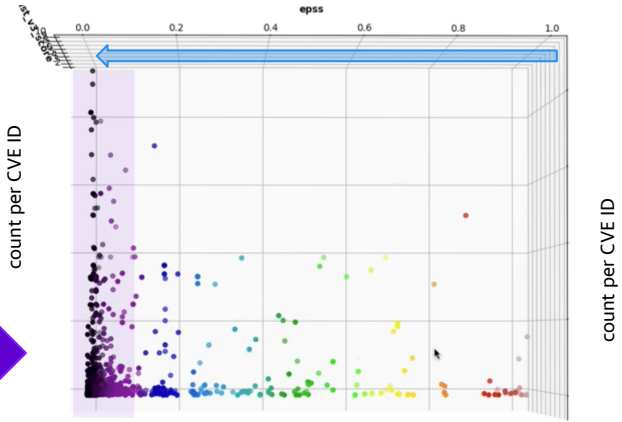
nist_v3_score is CVSS V3 score

Risk Remediation | Test | Test CVE Data | Internal CVE Data | DevOps Tools / Services

CVSS 7+ AND EPSS >= 0.1





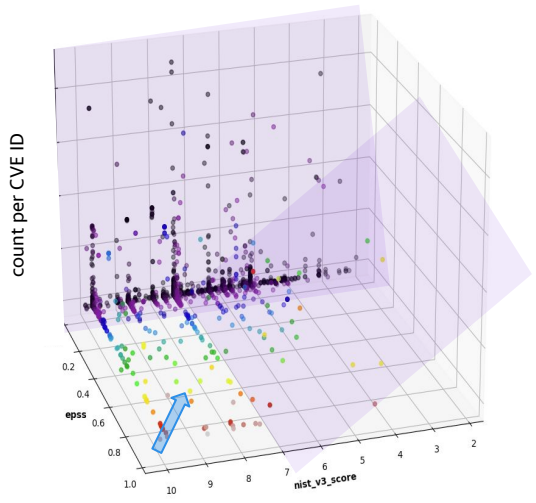


Many CVEs (across CVSS score range) are deprioritized due to low EPSS scores

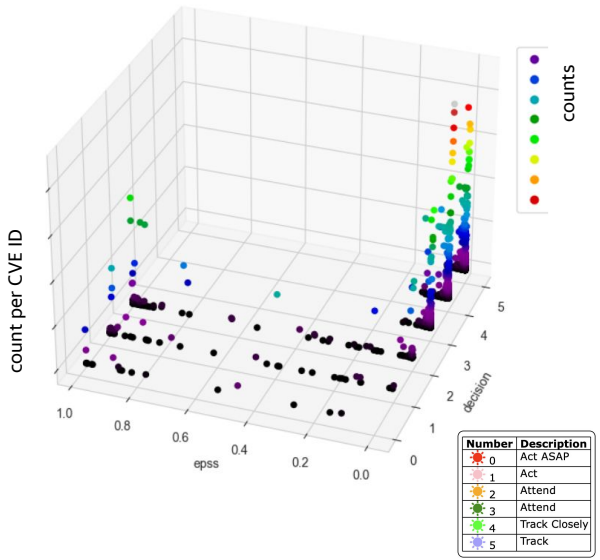
nist_v3_score is CVSS V3 score

Risk Remediation | Test | Test CVE Data | Internal CVE Data | DevOps Tools / Services

CVSS 7+ AND EPSS >= 0.1

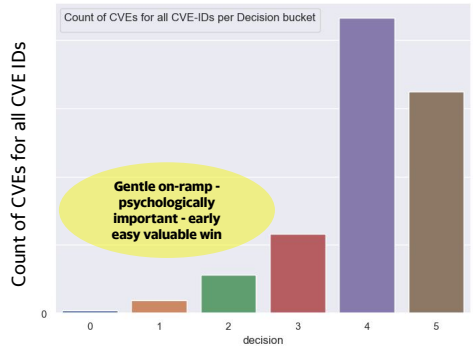


Risk-Based Prioritization DT Applied

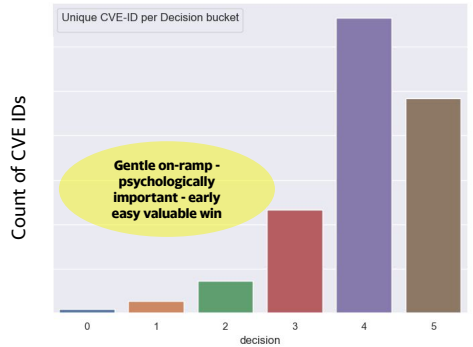


Counts

CVE Count



Unique CVE-ID Count



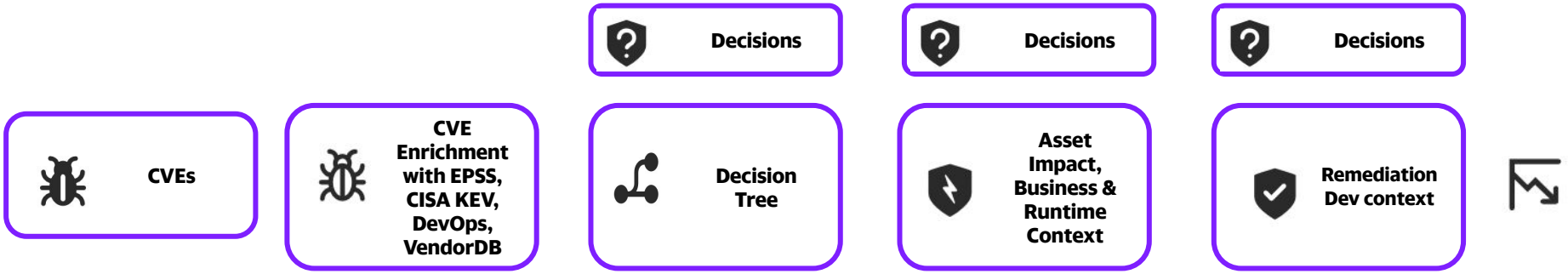
A CVE-ID may have 1 or more instances e.g. same CVE-ID found in multiple repos. Shown is

1. CVE Count: count of all CVE instances per CVE-ID per decision bucket
2. Count of unique CVE-IDs per decision bucket

Our Decision Tree based on our Risk Taxonomy gives additional (more targeted) prioritization over CVSS score or EPSS score. We get the best of both worlds by retaining EPSS so we can prioritize by EPSS across Decision band(s).

nist_v3_score is CVSS V3 score

Risk Based Prioritization Stages



Decision Tree

- Exploitation
- Automatable
- Technical Impact

Decision Tree?

- Mission & Well-being

Risk Remediation Taxonomy

- Threat
- Vulnerability

Risk Remediation Taxonomy

- Impact

Risk Remediation Taxonomy

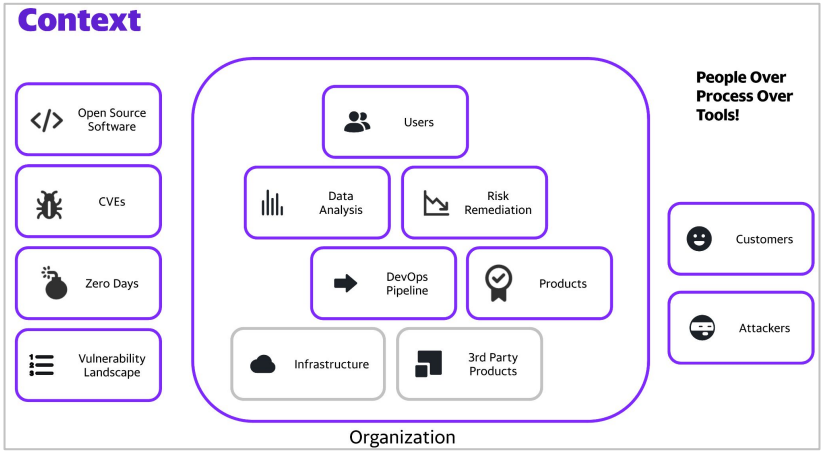
- Remediation

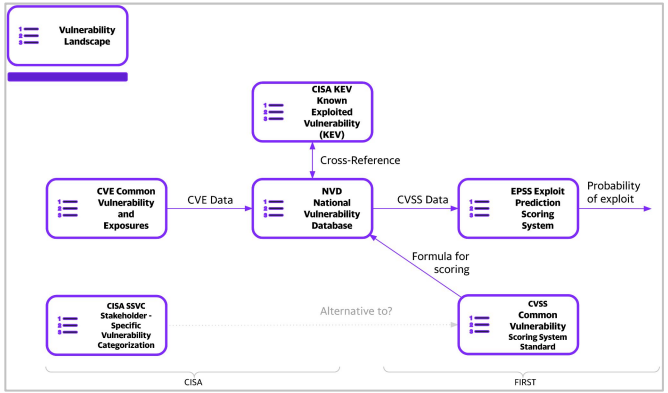
Risk

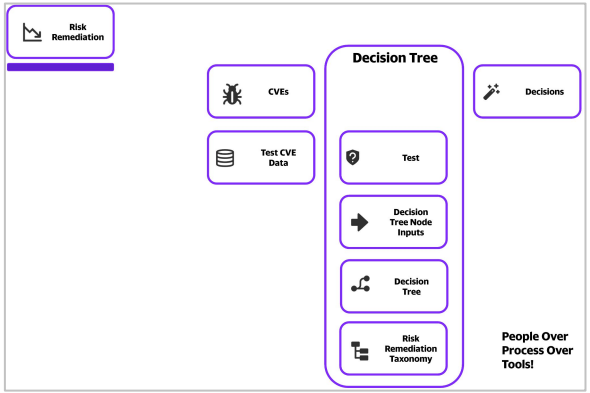
Run **automatically** daily with fresh data (**Generic**)

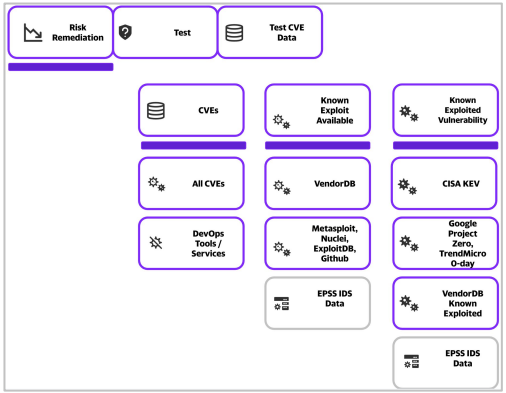
Context-specific Input from Product teams and Developers

Conclusion


Context
Open Source Software
CVEs
Zero Days
Vulnerability Landscape
Users
Data Analysis
Risk Remediation
DevOps Pipeline
Products
Infrastructure
3rd Party Products
Organization
People Over Process Over Tools!
Customers
Attackers


Vulnerability Landscape
CISA KEV Known Exploited Vulnerability (KEV)
Cross-Reference
CVE Common Vulnerability and Exposures
CVE Data
NVD National Vulnerability Database
CVSS Data
EPSS Exploit Prediction Scoring System
Probability of exploit
Formula for scoring
CISA SSVC Stakeholder - Specific Vulnerability Categorization
Alternative to?
CVSS Common Vulnerability Scoring System Standard
CISA
FIRST


Risk Remediation
CVEs
Test CVE Data
Decision Tree
Test
Decision Tree Node Inputs
Decision Tree
Risk Remediation Taxonomy
Decisions
People Over Process Over Tools!


Risk Remediation
Test
Test CVE Data
CVEs
Known Exploit Available
Known Exploited Vulnerability
All CVEs
VendorDB
CISA KEV
DevOps Tools / Services
Metasploit, Nuclei, ExploitDB, Github
Google Project Zero, TrendMicro O-day
EPSS IDS Data
VendorDB Known Exploited
EPSS IDS Data

Know

- What matters most to you in your DevOps pipeline
- Your tool(s) sweetspots and blindspots
- The root cause for your CVEs: EDA!
- Where your Paretos are
- Your Risk Taxonomy

Decision Trees

- **Our Decision Tree gives more targeted prioritization over**
  - > CVSS score, by using the CVSS Base Score parameters instead applied to our environment
  - > EPSS score, by covering the (majority of CVEs) case where EPSS score is low (when we can't tell from EPSS score if we should be worried or not).
- **We get the best of both worlds by retaining EPSS with our Decisions so we can**
  - > prioritize by EPSS across Decision band(s) where EPSS is not Low
  - > retain Temporal Data (EPSS scores are Temporal)
- **We get a Risk based SLA (Service Level Agreement) with sufficiently granular and understandable Decisions**

Users

security person

CISO

Now we have a unified prioritized personalized achievable view (across tools and teams) of what to fix first.

We can optimize our flow of value / software vs Risk!

developer

developer / leader

Customers

Organization

THANK YOU!

- ★ **Lisa** for the expert input, keeping all this real, and tolerating more dumb questions than any human should endure in one lifetime
- ★ **Nate** for his wisdom and gathering the data sources
- ★ **DJ** for his wealth of experience and feedback
- ★ **Yahoo** for cultivating such a rich environment for people to thrive, and putting People first
- ★ **EPSS SIG** for feedback & being receptive and responsive to my inputs
- ★ **Multiple vendors** for feedback
- ★ **Plantuml** Arnaud for a great tool!
- ★ **Denali** for Icons
- ★ **BSidesDub** Paul and crew

- ★ **You** for sharing 40 minutes of your lives with me.

yahoo!

Annex

Abstract

Understanding your vulnerability data to optimize your DevOps pipeline flow

DevOps pipelines typically contain several tools and services that detect publicly known security vulnerabilities (CVEs). Prioritizing the remediation of these vulnerabilities at scale is a hard problem.

What if we did some Data Analysis on these vulnerabilities at a system level, and use what we learn to prioritize by risk so we optimize efficiency versus coverage in what we fix?

In this talk, we'll set the stage for the Data Analysis by walking through:

- A real DevOps pipeline and what tools and services detect CVEs (versus those that don't)
- The properties we want to achieve with that DevOps pipeline
- The components of risk - and the data sources for these components
- The recent initiatives for vulnerability management and risk based prioritisation including EPSS (Exploit Prediction Scoring System) and CISA SSVC (Cybersecurity and Infrastructure Security Agency Stakeholder-Specific Vulnerability Categorization)

We'll then review the Data Analysis - what was done and how it was done and what we learnt.

Based on that Data Analysis, we'll examine the recipe developed for risk based prioritization at scale.

In this talk, you'll learn to understand risk based prioritization at scale to optimize flow of software through your DevOps pipeline versus security risk.

Data Analysis

Data Sources

1. CISA Known Exploited Vulnerability catalog
2. CISA Top (10) Routinely Exploited Vulnerabilities Alerts AA21-209A (2020-2021), AA22-117A (2021), AA20-133A (2016 to 2019)
3. EPSS
4. All CVE IDs from NVD
5. Vendor DB for exploit availability and other data -
    a. a commercial paid for product that we use that gives additional context
6. The 7 DevOps tools described that detect CVEs

The data analysed is from May 2023

Tools Used

1. Python Pandas to process the input data and create the output data for EDA and plots.
2. Python pandas_profiling for EDA (Exploratory Data Analysis) and PandasGUI
3. Python Seaborn, Plotly, matplotlib-venn to create the plots
4. PlantUML for tree diagrams

DIY. Most of the data sources used are open. Python is great for analysis and plots.

Zero Days

A zero-day vulnerability is a flaw in software or hardware that is unknown to a vendor prior to its public disclosure, or has been publicly disclosed prior to a patch being made available. As soon as a zero day is disclosed and a patch is made available it, of course, joins the pantheon of known vulnerabilities. Tenable 2022 Threat Landscape Report

EPSS scores won't be available for Zero Days (because EPSS depends on the CVE being published)

Tenable

- *Don't go chasing zero days, patch your known vulnerabilities instead....*
- ***Vulnerabilities increase risk, whether or not they start as zero days. We advise organizations to operate with a defensive posture by applying available patches for known, exploited vulnerabilities sooner rather than later.***

Tenable 2022 Threat Landscape Report

FIRST EPSS

"published exploit code is the biggest predictor of exploitation activity hands down" *FIRST EPSS, April 2023*

Gartner

- *Zero day vulnerabilities made up only approximately 0.4% of vulnerabilities during the past decade.*
- *The amount spent on trying to detect them is out of kilter with the actual risks they pose. This is compared with the massive numbers of breaches and infections that come from a small number of known vulnerabilities that are being repeatedly exploited.*
- ***As a top priority, focus your efforts on patching the vulnerabilities that are being exploited in the wild or have competent compensating control(s) that can. This is an effective approach to risk mitigation and prevention, yet very few organization do this.***

Focus on the Biggest Security Threats, Not the Most Publicized, Gartner, Nov 2017

Prioritize fixing known exploited vulnerabilities, then vulnerabilities with known exploit code

CVSS

Vulnerability Landscape

NVD National Vulnerability Database

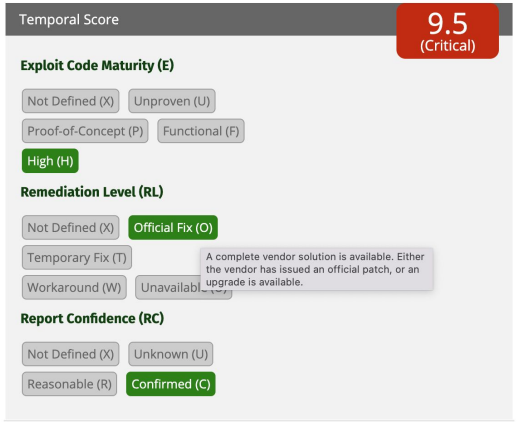
CVE CVSS Temporal and Environmental Score

CVSS Base scores relate to **Severity**

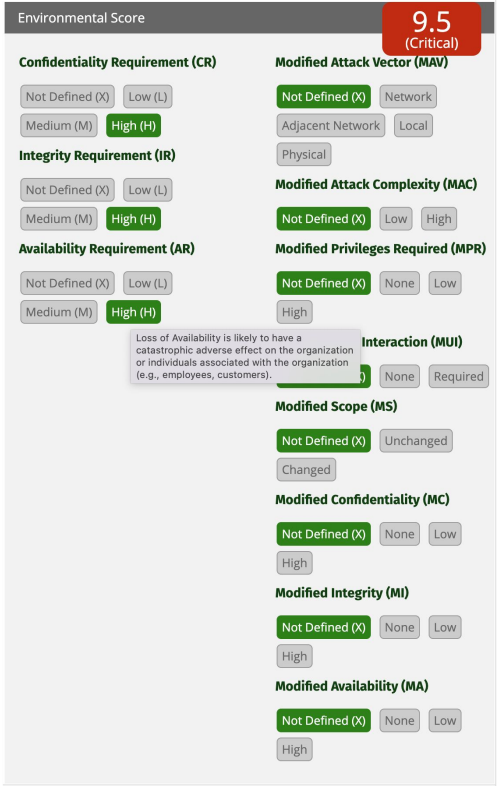
https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

Base Score did not change

https://nvd.nist.gov/vuln/detail/CVE-2021-44228#VulnChangeHistorySection



CVSS Temporal scores relate to **Threat**



CVSS Environmental scores relate to **Impact**

Metrics/score specified by:

- Base: NIST NVD
- Temporal: vulnerability product or information vendors or **you**
- Environmental: **you** as only you know your environment

CVE CVSS supports characteristics of a vulnerability that change over time, and that are unique to a user's environment. But these are rarely used.

https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H/**E:H/RL:O/RC:C/CR:H/IR:H/AR:H**

Upcoming CVSS 4.0 - What's New?

- Finer granularity in Base Metrics
  - Attack Requirements (AR) added as Base Metric
  - Enhanced User Interaction Granularity (None/Active/Passive)
- Removal of downstream scoring ambiguity (read: Scope)
  - C/I/A expanded into separate Vulnerable System C/I/A and Subsequent System C/I/A
- Simplification of Threat metrics and improved scoring impact
  - Remediation Level, Report Confidence, and Exploit Code Maturity simplified to Exploit Maturity
- Supplemental attributes for vulnerability response
  - Supplemental Metric: Automatable
  - Supplemental Metric: Recovery
  - Supplemental Metric: Value Density
  - Supplemental Metric: Vulnerability Response Effort
  - Supplemental Metric: Provider Urgency
- Additional applicability to OT/ICS/IoT
  - Safety Metric Values added to Environmental Metrics

FirstCon 2022

Value Density: Concentrated (Diffuse): The system that contains the vulnerable component is rich in resources. ... Examples of concentrated value are database systems, Kerberos servers, web servers hosting login pages, and cloud service providers. However, usefulness and uniqueness of the resources on the vulnerable system also inform value density

CVESS 4.0 is coming with improvements.

https://csrc.nist.gov/csrc/media/Presentations/2023/update-on-cvss-4-0/jan-25-2023-ssca-dugal-rich.pdf

CVSS 4.0 Calculator https://bit.ly/cvssv4-calculator

Vulnerability Landscape

NVD National Vulnerability Database

CVE CVSS Summary

Pros +

1. Common way to score vulnerabilities
2. CVSS Base Score commonly used
3. All Published CVEs have a CVSS Base score

Cons -

1. Base Values and associated scores are static
2. Most CVEs are scored High or Critical (in CVSS 3.1)
3. The origins and validation of the weightings in the formulas used to calculate the CVSS score are opaque
4. CVSS Temporal and Environmental scores not commonly used
5. CVSS is designed to be accurate only within +/- 0.5. In practice it's scored with errors of 2-4 points (Allodi et al. 2018) via Towards Improving CVSS CMU SEI

Guidance

"A comprehensive risk assessment system should be employed that considers more factors than simply the CVSS Base Score. ***Such systems typically also consider factors outside the scope of CVSS such as exposure and threat."***
CVSS User Guide from FIRST

PCI DSS 4.0 11.3.2.1 "External vulnerability scans are performed after any significant change as follows: Vulnerabilities that are scored 4.0 or higher by the CVSS are resolved."

CVSS should be used with other factors to assess Risk
CVSS scores have multiple issues that affect their use

KEV

CISA KEV - Active Exploitation

The main criteria for KEV catalog inclusion, is whether the vulnerability has been exploited or is under active exploitation*. These two terms refer to the use of malicious code by an individual to take advantage of a vulnerability. In reference to the KEV catalog, active exploitation and exploited are synonymous.*

A vulnerability under active exploitation is one for which there is reliable evidence that execution of malicious code was performed by an actor on a system without permission of the system owner.

Active exploitation, about the KEV catalog, includes attempted and successful exploitation.

- *Attempted exploitation occurs when an attacker executes code on a target system. Still, the code does not execute due to the system not being vulnerable or the system being a honeypot, etc. A honeypot is a computer security mechanism set to detect, deflect, or, in some manner, counteract attempts at unauthorized use of information systems. Successful malicious code execution on a honeypot is considered attempted exploitation because the attacker does not obtain target information.*
- *Successful exploitation occurs when attackers exploit vulnerable code on a target system, allowing them to perform additional, unauthorized actions on that system or network.*

The two key takeaways for active exploitation are: ***the intent of the actor is to succeed in exploitation and the attack(s) occurred in real-time, or "in the wild."***

Events that do not constitute as active exploitation, in relation to the KEV catalog, include:

- *Scanning*
- *Security research of an exploit*
- *Proof of Concept (PoC)*

CISA KEV criteria for Active Exploitation are different than EPSS

CISA KEV Summary

Pros +

1. Free (one of the few free sources of vulnerability exploitation activity)
2. Puts exploitability first over e.g. severity of vulnerability per CVSS
3. Vendor Vulnerability DBs and tools use it

Entry Criteria for CISA KEV

1. The vulnerability has an assigned Common Vulnerabilities and Exposures (CVE) ID.
2. There is reliable evidence that the vulnerability has been actively exploited in the wild.
3. There is a clear remediation action for the vulnerability, such as a vendor-provided update.

Cons -

1. It contains a small number (~1K) of actively exploited vulnerabilities (~10K 200K CVEs of which ~5% are exploited)
   a. Other vulnerability intelligence sources required to identify broader set of exploited vulnerabilities
   b. New (Nov 2021) but likely to grow (more CVEs added) significantly based on recent growth
2. It's opaque i.e. the details behind why a CVE is, or is not, in CISA KEV are not clear, and who's exploiting it. No context given on the Threat aspect - only the Vulnerability
   a. Some CVEs included in the KEV list have no public proof of concept or reporting of exploitation in the wild
   b. *"42 vulnerabilities assigned CVEs in 2022, which were publicly reported to be exploited in the wild. Yet, none of these vulnerabilities are in the CISA KEV Catalog."* https://vulncheck.com/blog/2022-missing-kev-report

CISA KEV is a useful reference for known exploitation. It's likely to grow over time.

EPSS

EPSS Variable Contribution

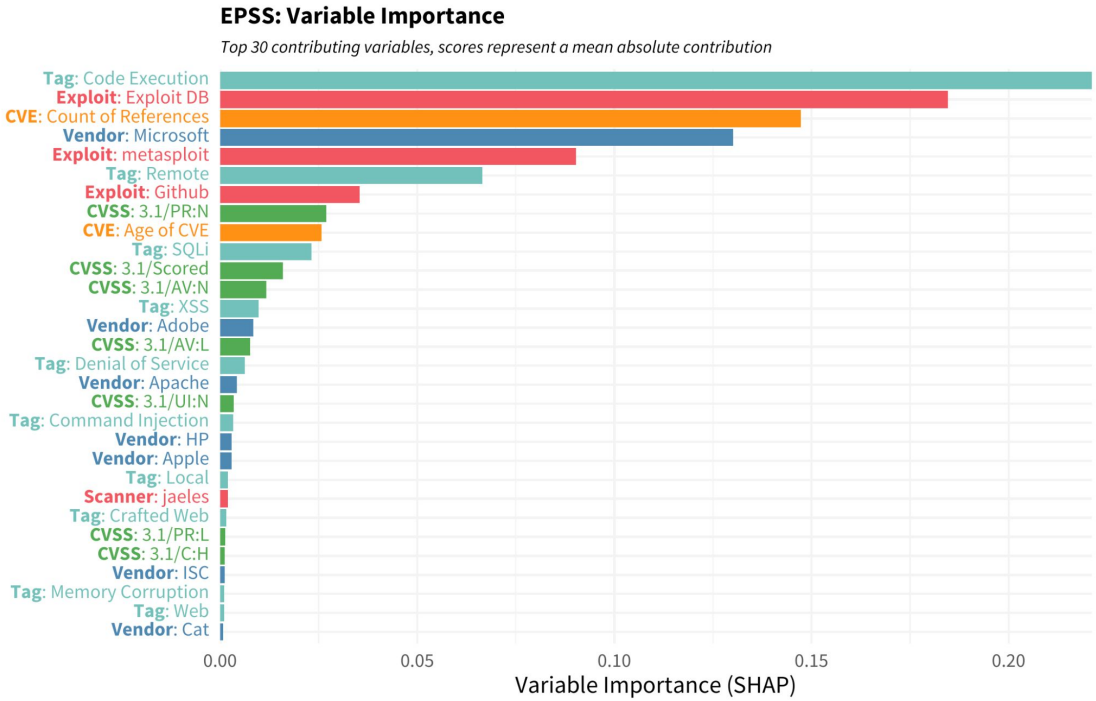
For the EPSS ML Model, first.org did a SHAP values analysis on the variables in the model.

The figure shows the top variables sorted by their contribution.

Takeaway

These variables and ordering could also be applied to a traditional rule-based risk prioritization algorithm.

Having exploit code published and easily available for a remote code execution vulnerability with no privilege required on a Microsoft product would probably see exploitation activity.



https://www.first.org/epss/model

EPSS V3

Improved Precision

EPSS V3 launched Mar 2023, offers improved precision at identifying vulnerabilities likely to be exploited in the wild.

- Expand the sources of exploit data by partnering with multiple organizations willing to share data for model development, and engineer more complex and informative features.
- Allowed the proposed v3 model to achieve **an overall 82% improvement in classifier performance over v2**
- This boost in prediction performance allows organizations to substantially improve their prioritization practices and design data-driven patching strategies.

Data Sources Used to Feed the EPSS V3 Model

Description	# of variables	Sources
Exploitation activity in the wild (ground truth)	1 (with dates)	Fortinet, AlienVault, ShadowServer, GreyNoise
Publicly available exploit code	3	Exploit-DB, GitHub, MetaSploit
CVE is listed/discussed on a list or website ("site")	3	CISA KEV, Google Project Zero, Trend Micro's Zero Day Initiative (ZDI)
Social media	3	Mentions/discussion on Twitter
Offensive security tools and scanners	4	Intrigue, sn1per, jaeles, nuclei
References with labels	17	MITRE CVE List, NVD
Keyword description of the vulnerability	147	Text description in MITRE CVE List
CVSS metrics	15	National Vulnerability Database (NVD)
CWE	188	National Vulnerability Database (NVD)
Vendor labels	1,096	National Vulnerability Database (NVD)
Age of the vulnerability	1	Days since CVE published in MITRE CVE list

"The exploit data used in this research paper covers activity from July 1, 2016 to December 31st, 2022 (2,374 days / 78 months / 6.5 years), over which we collected 6.4 million exploitation observations (date and CVE combinations), targeting 12,243 unique vulnerabilities. Based on this data, we find that 6.4% (12,243 of 192,035) of all published vulnerabilities were observed to be exploited during this period"

EPSS v3 allows organizations to substantially improve their prioritization practices

Enhancing Vulnerability Prioritization: Data-Driven Exploit Predictions with Community-Driven Insights, Feb 2023 https://arxiv.org/pdf/2302.14172.pdf

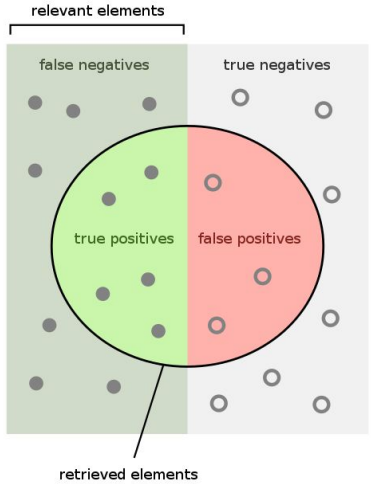
EPSS V3

Precision (efficiency) measures how well resources are being allocated, (where low efficiency represents wasted effort), and

- calculated as the **true positives divided by the sum of the true and false positives.**
- In the vulnerability management context, efficiency addresses the question, “out of all the vulnerabilities remediated, how many were actually exploited?”
- If a remediation strategy suggests patching 100 vulnerabilities, 60 of which were exploited, the efficiency would be 60%.

Recall (coverage), on the other hand, considers how well a remediation strategy actually addresses those vulnerabilities that should be patched (e.g., that have observed exploitation activity),

- calculated as the **true positives divided by the sum of the true positives and false negatives**.
- In the vulnerability management context, coverage addresses the question, “out of all the vulnerabilities that are being exploited, how many were actually remediated?”
- If 100 vulnerabilities are exploited, 40 of which are patched, the coverage would be 40%.



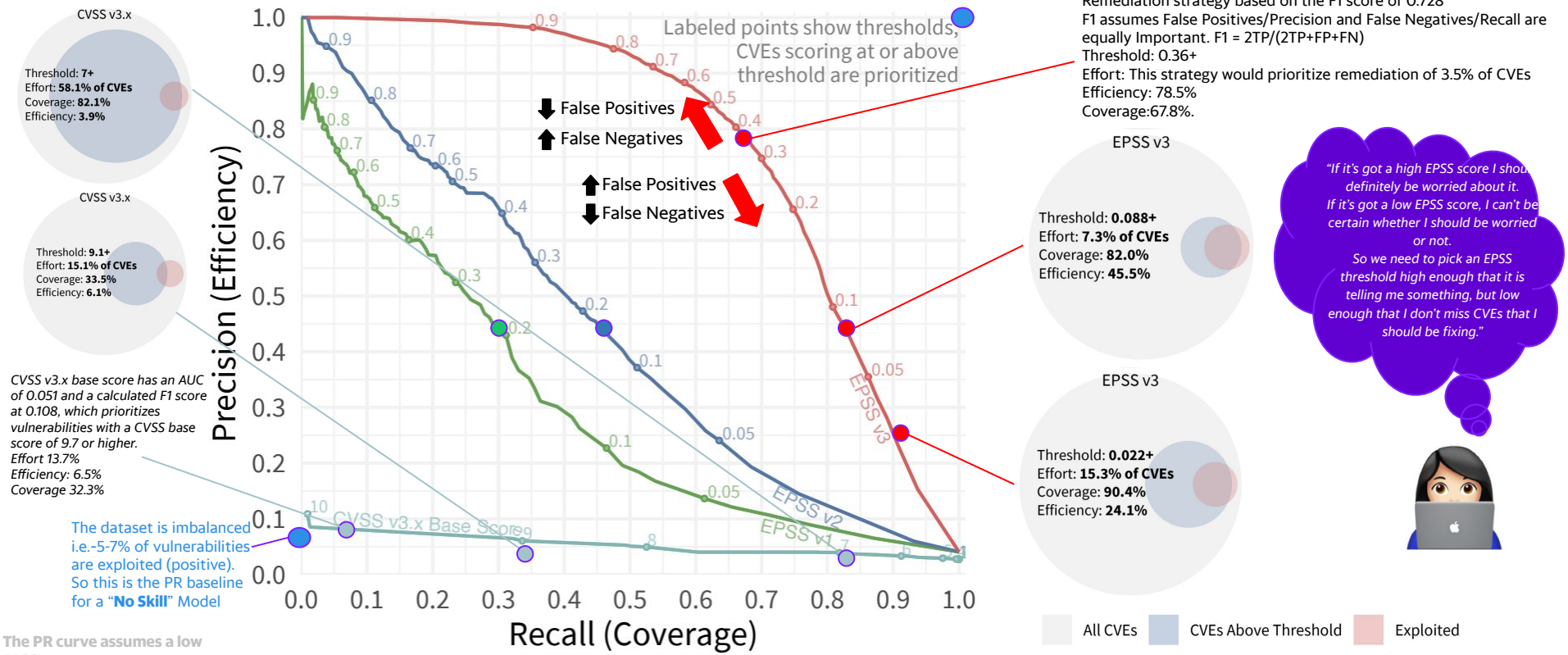
https://en.wikipedia.org/wiki/Precision_and_recall

“Relevant elements” is Exploited CVEs in our case.

A PR curve is drawn by picking Threshold values, then working out the PR values.

Enhancing Vulnerability Prioritization: Data-Driven Exploit Predictions with Community-Driven Insights, Feb 2023 https://arxiv.org/pdf/2302.14172.pdf

What EPSS Threshold to use?



The PR curve assumes a low EPSS score means not an exploit - which is not the case.

Pick EPSS Threshold per above. Start Conservative. Adjust based on YOUR CVE data.

Enhancing Vulnerability Prioritization: Data-Driven Exploit Predictions with Community-Driven Insights, Feb 2023 https://arxiv.org/pdf/2302.14172.pdf

EPSS Predictability & Percentile Scores

https://api.first.org/data/v1/epss?cve=CVE-2021-44228

{"cve":"CVE-2021-44228","epss":"0.975780000","percentile":"0.999990000","date":"2023-04-17"}

1. EPSS provides 2 scores:
   a. **a probability of observing exploitation activity in the next 30 days**
   b. a **percentile** (a rank ordering of probabilities from highest to lowest).
2. **Probability** is the "the most objective way of presenting EPSS scores"
3. **Percentiles** are a direct transformation from probabilities and provide a measure of an EPSS probability *relative to all other scores*.
   a. A CVE EPSS Percentile score of N% means that the CVE EPSS Probability score is greater than N% of CVE EPSS Probability scores in the population (population is all CVEs (~200K) that have an EPSS score)
   b. A Percentile score based on the population of all your CVEs is more relevant - and easily calculated.

Which one to use?

It is the official guidance and recommendation of EPSS that: When communicating a single "EPSS score," that value should be the probability score (not the percentile). It can be expressed as either a decimal value (0.153) or a percent (15.3%), though the prefered method is a percent. As often as possible, the percentile should be communicated with the probability and should include the appropriate suffix (i.e. "st", "nd", "rd", "th") for display. For example, "15.3% (92nd)" implies that the vulnerability has a 15.3% probability, and is ranked in the 92nd percentile.

Which Percentile?

The Percentile score is relative to all ~~200K published CVE IDs that have an EPSS score.
A fraction of those CVE IDs will apply to a typical organization e.g. ~~20K.
A user is likely more interested in the EPSS Percentile for their organization - than for all CVE IDs.
E.g. A CVE's EPSS percentile could be e.g. 60% - but in the 90% percentile for the CVEs in the organization (if the organization has few CVEs with high EPSS score).
The EPSS Percentile is easily calculated for their organization (subset of CVEs applicable to their organization).

EPSS Probability: Probability of observing exploitation activity in the next 30 days

Vulnerability Landscape

EPSS Exploit Prediction Scoring System

EPSS Summary

Pros +

1. Gives a measure of exploit predictability that is unique (useful in the absence of exploitation evidence)
2. Open (but opaque: the model and data inputs, weights, are not available)
3. Coverage is good i.e. all Published CVEs have an EPSS score

Usage

1. For a CVE:
   a. **“If it’s got a high EPSS score I should definitely be worried about it”**
   b. **“If it’s got a low EPSS score, I can’t be certain whether I should be worried or not”**
2. EPSS scores change (as expected i.e. Temporal)

Cons -

1. **Most CVEs have lower EPSS scores, and it’s not clear if this is because of**
   a. **low information/confidence**
   b. **high information/confidence in low probability**
2. There’s a significant lag (up to 10d +) between a critical vulnerability being known and associated EPSS scores being published due to relying on CVE publication.
3. Your environment may be different than the environment for the EPSS Model e.g. IOT, Medical.
4. EPSS model does not differentiate between 1 detection vs exploitation at scale *FIRST EPSS, April 2023*

EPSS is a useful tool (when you understand what it can and can’t do... as with any tool)

SSVC

CISA SSVC CMU SEI Insights

Carnegie Mellon University Software Engineering Institute (CMU SEI) developed the SSVC. Their document(s) provide a lot of insights into the rationale - including criticisms of CVSS. CISA SSVC has many but not all of the features proposed.

Why?

The context of the vulnerability, and the systems it impacts, are inextricably linked to managing it. Temporal and environmental considerations should be primary, not optional as they are in CVSS.

Goals

The following are our design goals for a vulnerability management process:

- *Outputs are decisions.*
- *Pluralistic recommendations are made among a manageable number of stakeholder groups.*
- *Inputs are qualitative.*
- *Outputs are qualitative, and there are no (unjustified) shifts to quantitative calculations.*
- *Process justification is transparent.*
- *Results are explainable.*

These goals prevent the use of:

- **Scores (*Outputs are qualitative*)**
- **ML (*Results are explainable*)**

The CMU SEI document gives some good insights into CVSS, EPSS and the landscape in general

SEI CMU Prioritizing Vulnerability Response: A Stakeholder-Specific Vulnerability Categorization (Version 2.0)

CISA SSVC Summary

Pros +

1. **Focuses on what matters: risk (starting with active exploitation or exploitation Proof Of Concept), impact to the organisation, and what action needs to be taken when**
2. The Decision Tree for Criteria gives a very clear visual of all the parameters and risk remediation/mitigation. This also facilitates DT Classification analysis.
3. Public Well-Being Impact: should we have similar customer-focused parameter for our customers (though the "types of harm" would be very different)?

Cons -

1. The Mission & Well-being - especially the Public Well-being Impact criteria are not portable to organizations (though they can and probably should be customized).
2. It's not obvious what risk parameters should be used to inform each decision node (though some worked examples are available).
3. "standard update timelines" not defined - though part of the vulnerability scoring decision
4. CISA SSVC does not include "System Exposure" "The Accessible Attack Surface of the Affected System or Service" per original SEI CMU paper
5. Limited integration with other systems as of now.

CISA SSVC is a great initiative and reference - taking a pragmatic approach to vulnerability management. The SEI CMU document and Decision Trees behind it has a lot of insights that can be applied.

SEI CMU Prioritizing Vulnerability Response: A Stakeholder-Specific Vulnerability Categorization (Version 2.0)