CVE Services Workshop

November 2, 2022

Agenda

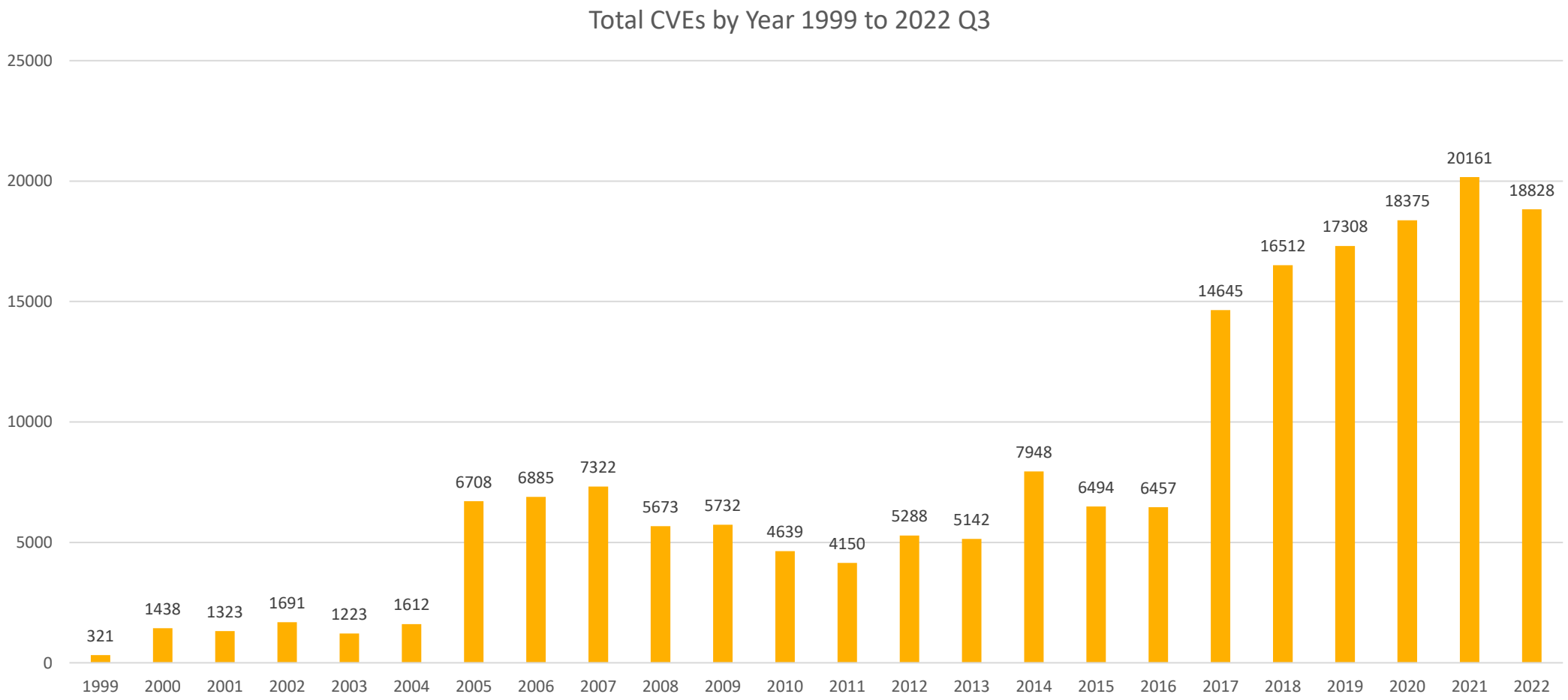
- 10:00-10:05—Welcome (Chris Levendis)
- 10:05-10:30—CVE Program Progression (Lisa Olson)
- 10:30-11:00—Introduction to new CVE Services (Kris Britton)
- 11:00-12:00—JSON 5.0: Introduction/Tips/Guidance (Chandan Nandakumaraiah)
- 12:00-1:00—BREAK
- 1:00-1:10—How to get a CVE Services Account (Dave Morse)
- 1:10-1:25—CVE Record Workflow Tutorial (Art Manion)
- 1:25-1:40—Vulnogram Tutorial (Art Manion)
- 1:40-1:55—CVEClient Tutorial (Art Manion)
- 1:55-2:00—Closing Remarks (Chris Levendis)

CVE Program Progression
(Lisa Olson)

CVE Program—What is it?

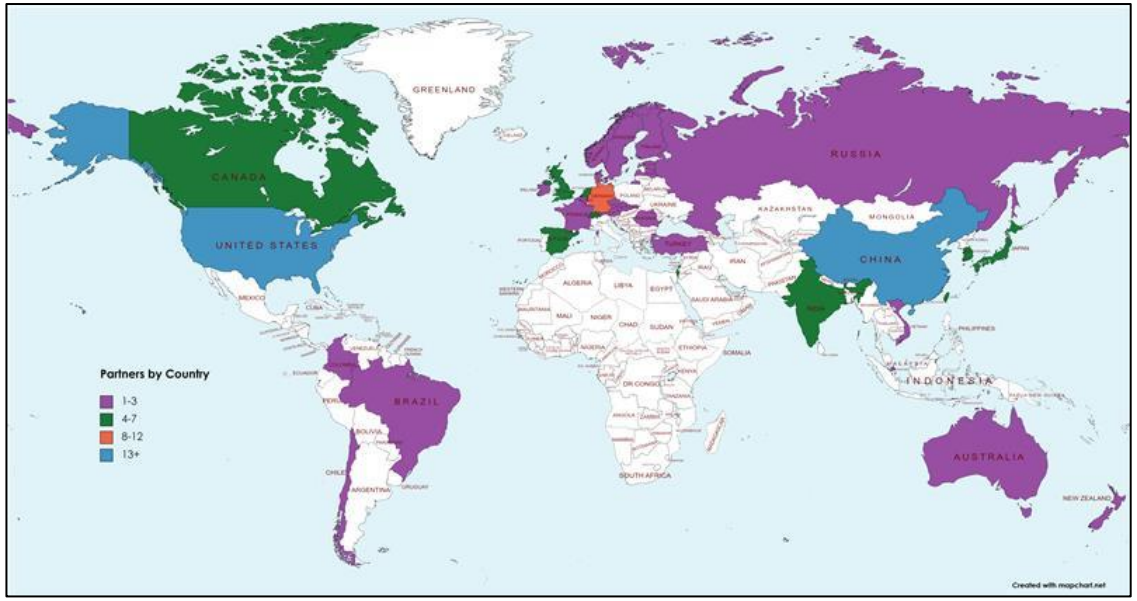
- CVE = Common Vulnerabilities and Exposures
- The mission of the CVE Program is to identify, define, and catalog publicly disclosed cybersecurity vulnerabilities
- The program was started in 1999, funded by the U.S. Federal Government
- The MITRE Corporation is paid by the U.S. Federal Government to administer the CVE Program
- The CVE Program is a global community effort managed by the CVE Board
- CNAs assign their own CVE IDs that they get from the CVE Program; CVE IDs are universally unique

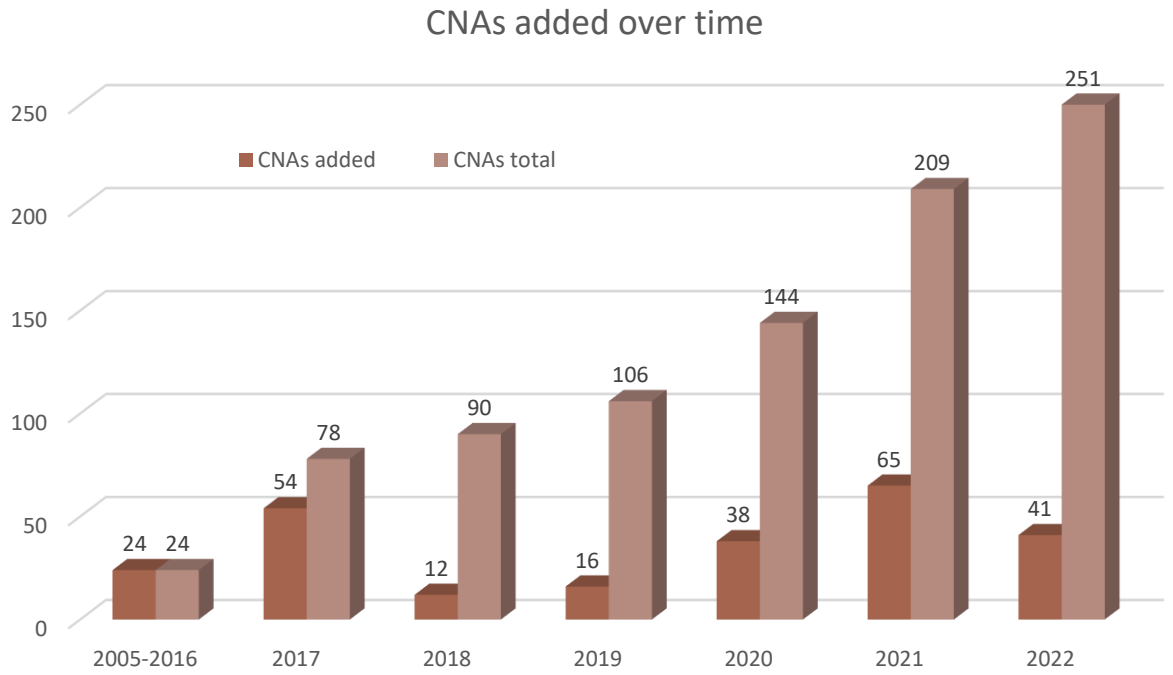
CVE Numbers Growth



CNA Growth

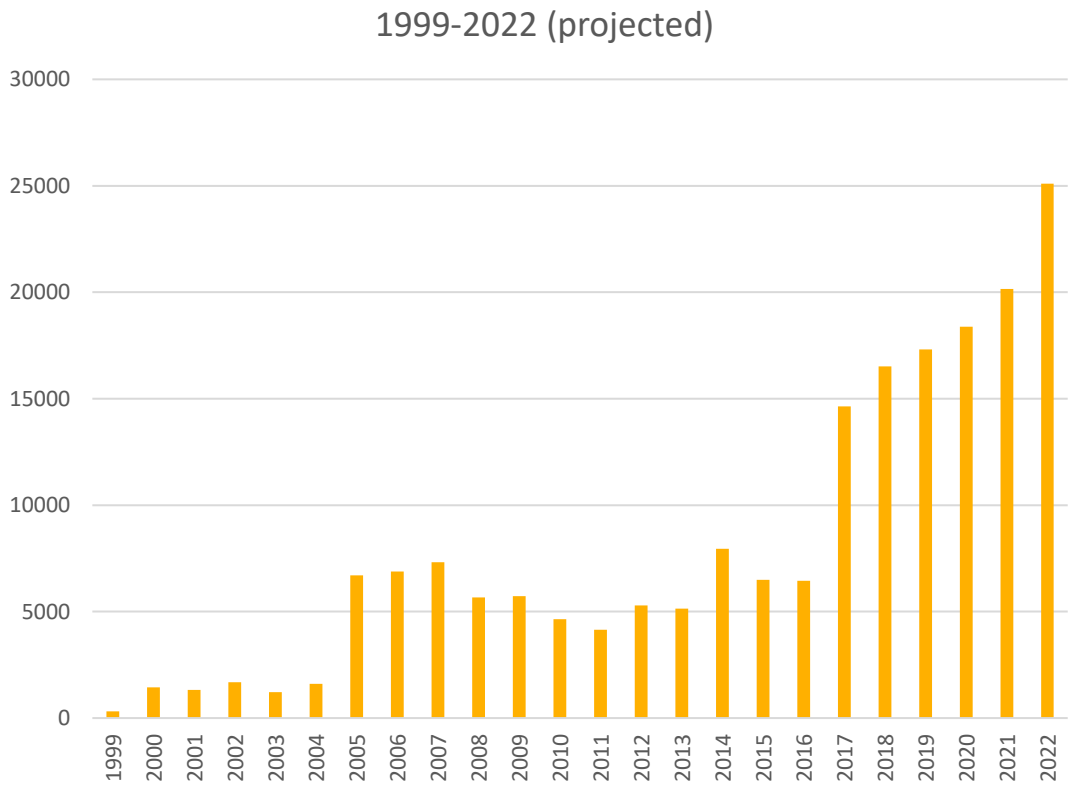
- 251 total partners
- 35 countries





Evolution of CNA Interaction with CVE Program

- 1999-2016, MITRE was the only team that doled out CVE IDs and instantiated CVEs
- In 2016, the CVE Board agreed on need to federate the CVE Program
- In 2016, CNAs were asked to instantiate their own CVEs
- In 2018, GitHub Pilot was stood up so that CNAs could package up CVE information and do a PR through GitHub to instantiate their CVEs
- In December 2020, ID Reservation Service (IDR) was launched so that CVE ID(s) are delivered via an API on-demand
- In 2021, www.cve.org was launched
- 2022 - CVE Record Submission and Upload Service (RSUS) just launched softly. CNAs can now submit JSON 5 records through an API.



What are the Next Steps?

- **Incorporating the additional service needed for fully implementing the programs federated model**
  - More formal User Registry Service allowing more control over access and permissions
  - Enhanced search on the website, allowing more advanced searching of the CVE Records on www.cve.org
- **Looking to enrich CVE data with authorized contributors**
  - Authorized Data Providers, allowing 3rd parties to provide rich data to existing CVEs in the corpus

Introduction to New CVE Services
(Kris Britton)

Introducing CVE Services

- What is CVE Services?
- What is the goal CVE Services Target Architecture?
- Where are we now (CVE Services Transitional Architecture)?
- Beginning the Transition to CVE Services/JSON 5.0

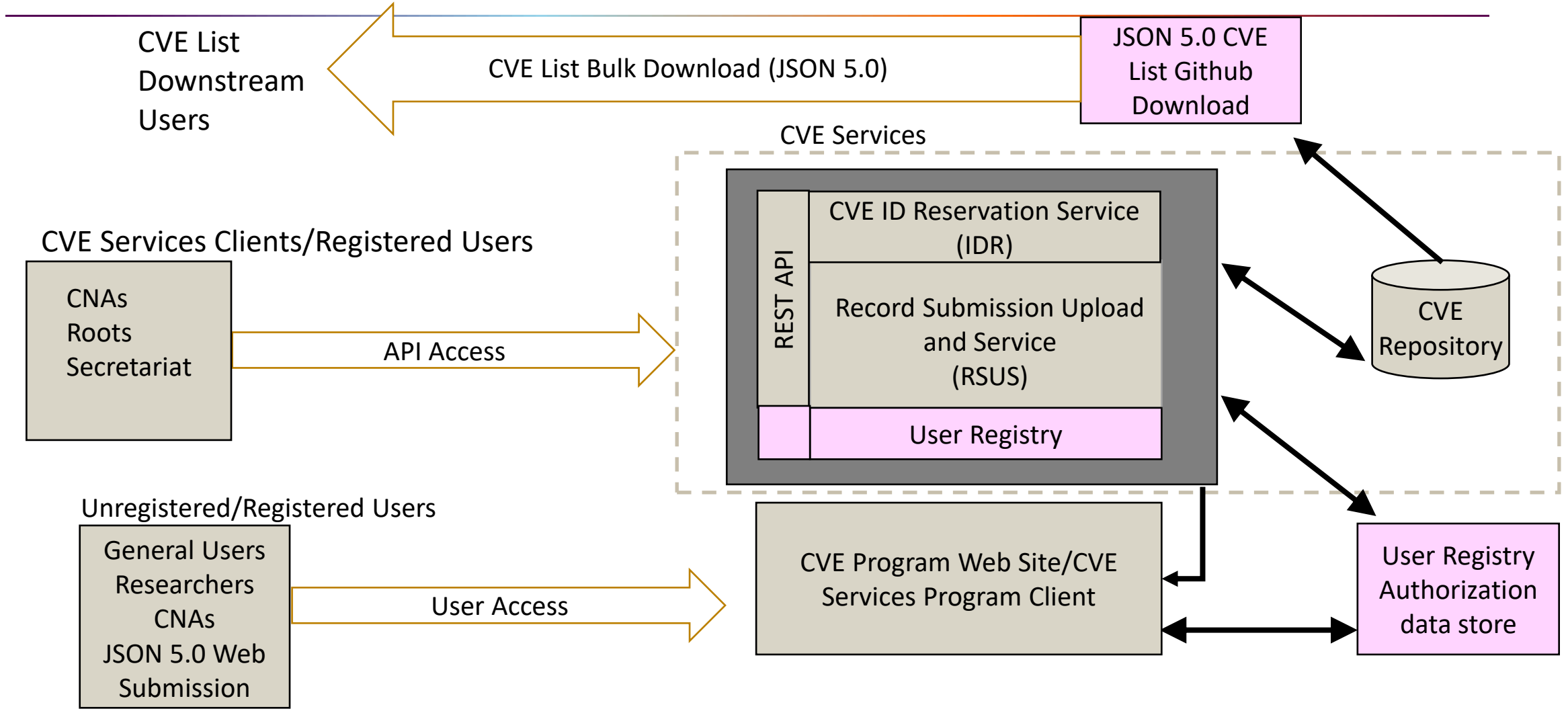
What is CVE Services? (1 of 2)

The CNA Superhighway to CVE Reservation, Submission, Update and Maintenance

What is CVE Services? (2 of 2)

- **A web addressable application for CVE Numbering Authorities (CNAs)**
- **Comprises a series of RESTful endpoints / Application Programming Interfaces (APIs) to which CNA "clients" will connect to perform common functions such as:**
  - Reserving a CVE ID
  - Submitting/updating a CVE Record for publication
  - Managing the CVE Services users at your CNA (i.e., create accounts, change password)
- **Instantaneous, automatic response**
  - No human intervention, no lag time

CVE Program Automation Target Architecture



CVE Services Subsystems (1 of 4)

- **CVE ID Reservation (IDR) Service**
  - A service that comprises APIs/functions that allow CNAs to do "just in time" CVE ID Reservation
  - Automated, immediate response to the requestor
    - No human in the loop
    - No longer a need to reserve CVE IDs that are not going to be immediately used
    - Deployed December 2020, CVE Services 1.0.0
    - Upgraded in June 2021, and October 2022

CVE Services Subsystems (2 of 4)

- **CVE Record Submission and Upload Service (RSUS)**
  - A service comprising APIs/functions that provide a CNA with the ability to:
    - Submit a CVE Record
    - Update a CVE Record
  - "Soft Deploy" October 24, 2022
    - A deployment with basic submission/update functions available for JSON 5.0 using CVE Services

CVE Services Subsystems (3 of 4)

- **CVE Repository**
  - A structured document, NoSQL database built around a CVE Program defined schema
  - Hosted by the Amazon Web Service DocumentDB technology
  - CVE Schema JSON 5.0 deployed
    - **Now the "format of record" for the CVE Program**
    - JSON 4.0 submission/download continues to be supported during a "transition period"
      - JSON 4.0 will ultimately be deprecated (Sunset Date not set)

CVE Services Subsystems (4 of 4)

- **User Registry Subsystem and User Registry Authorization Data Store**
  - A Subsystem that will comprise APIs/Functions/Datastore that will provide finely grained access control and authorization to various program services
  - Will support CVE Services as well as a future CNA Service Portal
  - Design/implementation to be part of a future CVE Service deployment (not part of CVE Services 2.1)

CVE Services Clients (1 of 2)

- **CVE Services are available to CNAs only**
  - General public/researchers will use the CVE Program website
  - Individuals authorized by CNAs are required to have a CVE Services Account to use CVE Services and will be required to authenticate to CVE Services before use
- **CNAs will adopt (or develop) a CVE Services Client that will interface with the CVE Services API**

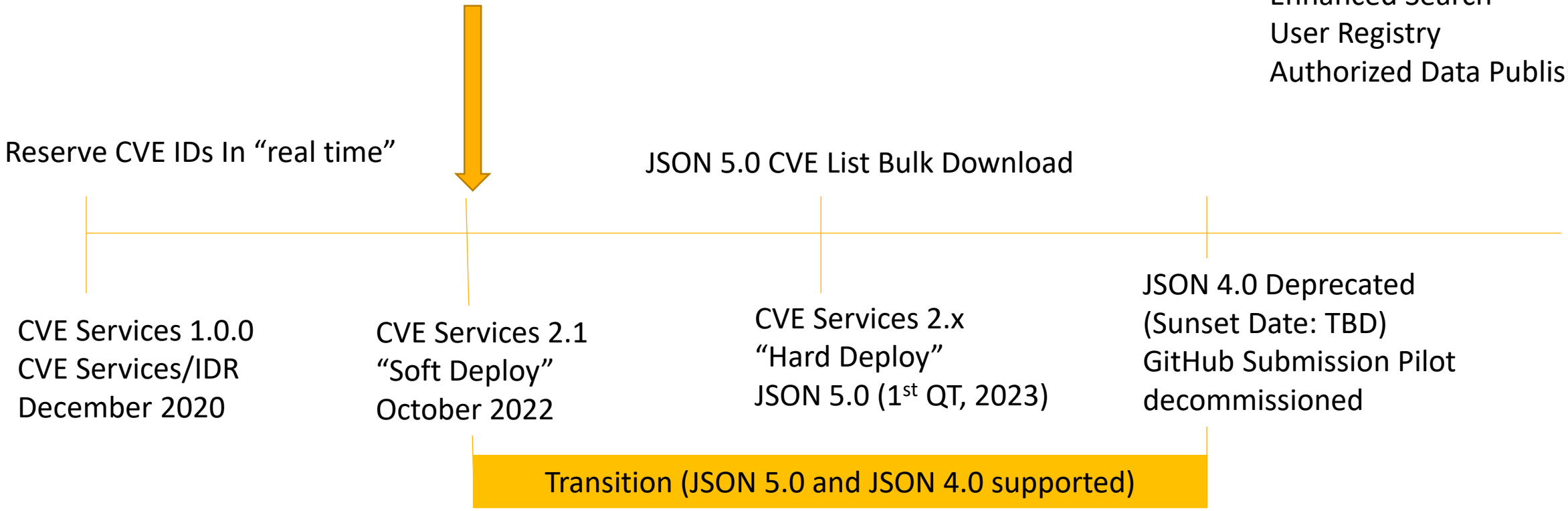
CVE Services Clients (2 of 2)

- **CVE Service Clients available for adoption/use**
  - Vulnogram, CVEClient, cvelib
  - Each client enables CNAs to:
    - Reserve a CVE ID using IDR
    - Submit/update a JSON 5.0 record using RSUS
    - Manage their CNA CVE Services users
  - All clients are open-source projects available on GitHub

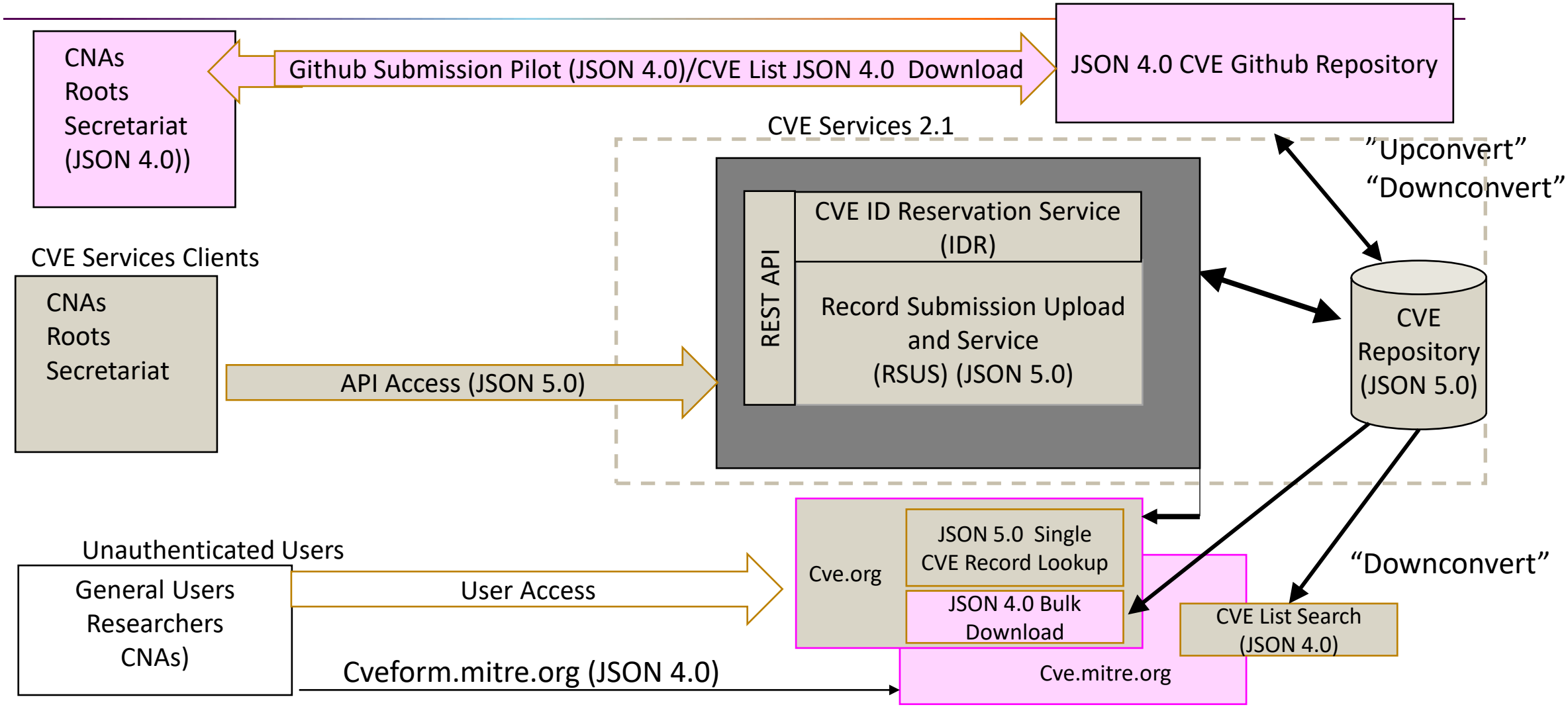
We are Here…

CVE Program JSON 5.0 Schema of Record
Submit/Review/Update JSON 5.0 Records available

Future:
Enhanced Search
User Registry
Authorized Data Publisher



CVE Transitional Architecture



What Should CNAs Begin Doing? (1 of 2)

- **Get Informed**
  - Become familiar with JSON 5.0
  - Become familiar with CVE Services
    - Determine whether to adopt or build a CVE Services Client
    - Get a CVE Services Account
    - Get comfortable using CVE Services by using the CVE Services Testing Instance
      - CVE Services Testing Instance available (see Getting Started with CVE Services) on the github.io page
    - Review the Transition Frequently Asked Question list

What Should CNAs Begin Doing? (2 of 2)

- **Review/update CNA historical CVE Services Records**
  - https://github.com/CVEProject/cvelistV5
    - Not a maintained list
    - A view of "upconverted" records as of October 4, 2022
    - Make updates as appropriate (after Prioritized Issues are addressed)
- **Report any observed issues over the course of your transition to**
  - CVE Services Slack Channel (email rbritton@mitre.org for an invite)
  - AWG (awg@cve-cwe-programs.groups.io)
  - CVE Program Request web forms (use "other" form)

Where to Start…

- **This Workshop….**
- **The CVE Program Automation website**
  - See all the bulletins that have been sent thus far
  - Check out the "self help" pages and the Frequently Asked Questions list
- **Where to get help:**
  - CVE Services Slack Channel (rbritton@mitre.org for invitation)
  - CVE Automation Working Group (meet every Tuesday, 4:00 PM EDT, send request to rbritton@miter.org for invite)

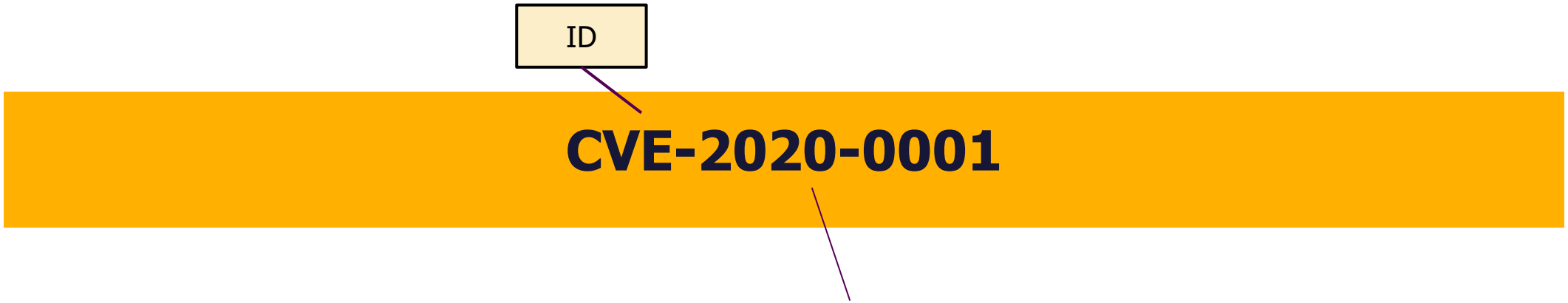
JSON 5.0: Introduction/Tips/Guidance
(Chandan Nandakumara)

What is a CVE Record?

- **A CVE Record:**
  - Is an entry in the CVE List
  - Helps identify a distinct vulnerability

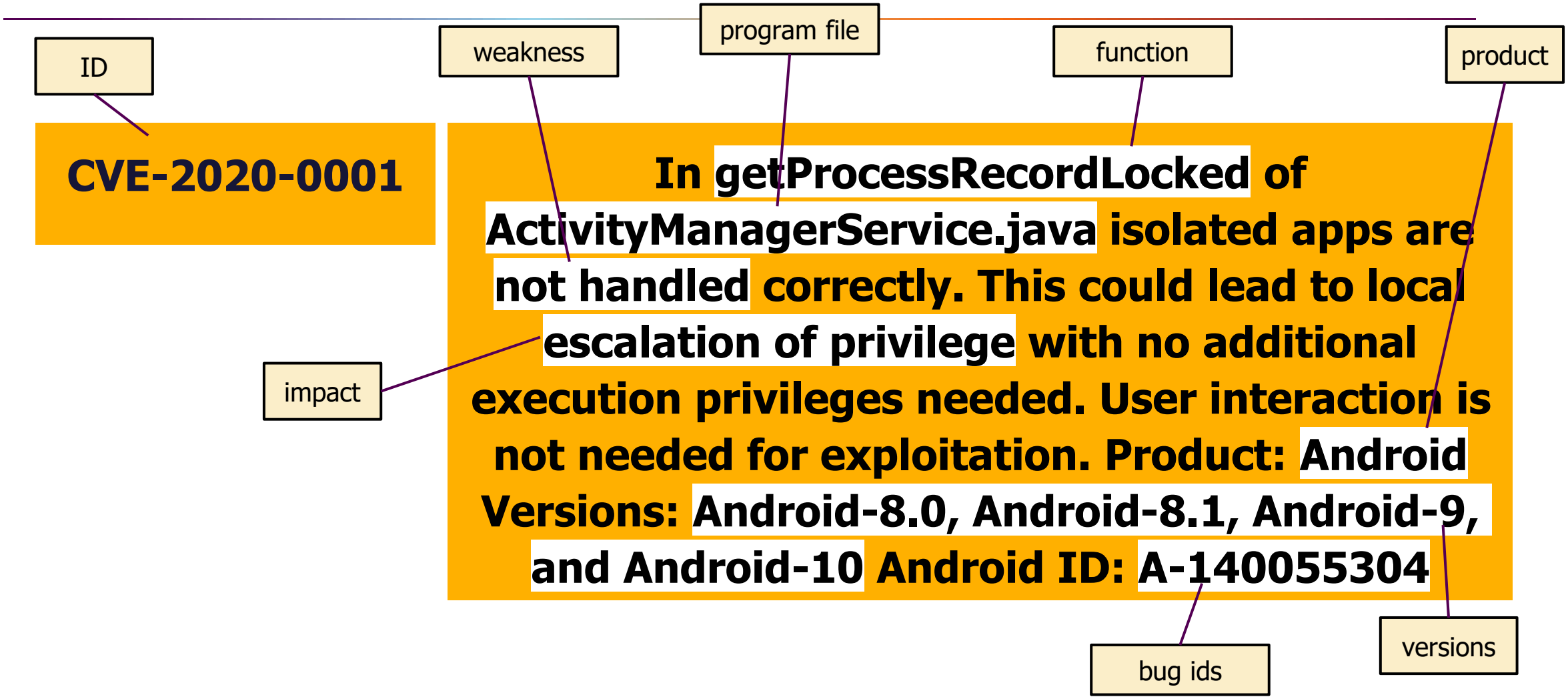
ID	Description	Links	Meta
CVE-2020-0001	In getProcessRecordLocked of ActivityManagerService.java isolated apps are not handled correctly. This could lead to local escalation of privilege with no additional execution privileges needed. User interaction is not needed for exploitation. Product: Android Versions: Android-8.0, Android-8.1, Android-9, and Android-10 Android ID: A-140055304	https://source.android.com/security/bulletin/2020-01-01	CNA:Android Date: 2019-10-17 State: PUBLISHED

What does a CVE Record Contain? (1 of 2)



The ID by itself does not contain much information. It may have a hint about the year the CVE was assigned.

What does a CVE Record Contain? (2 of 2)



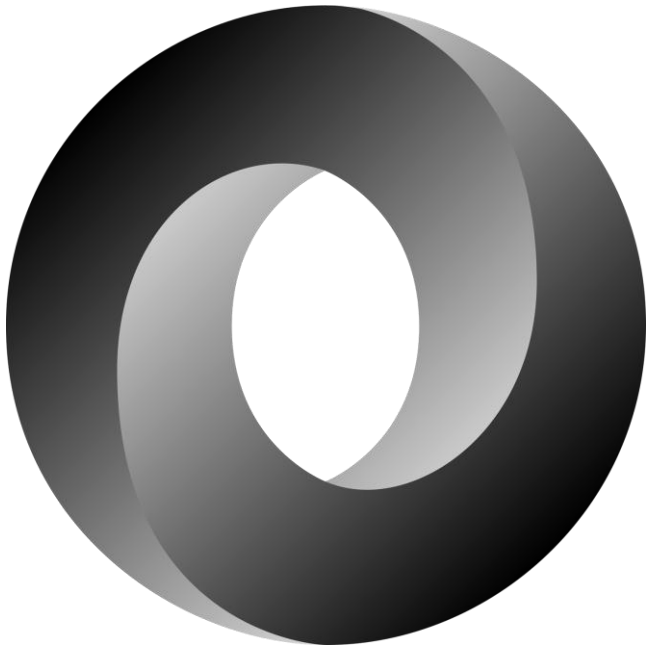
Minimum Information in a Record

- ID
- CNA
- affected things
- public reference

Additionally useful identifying information

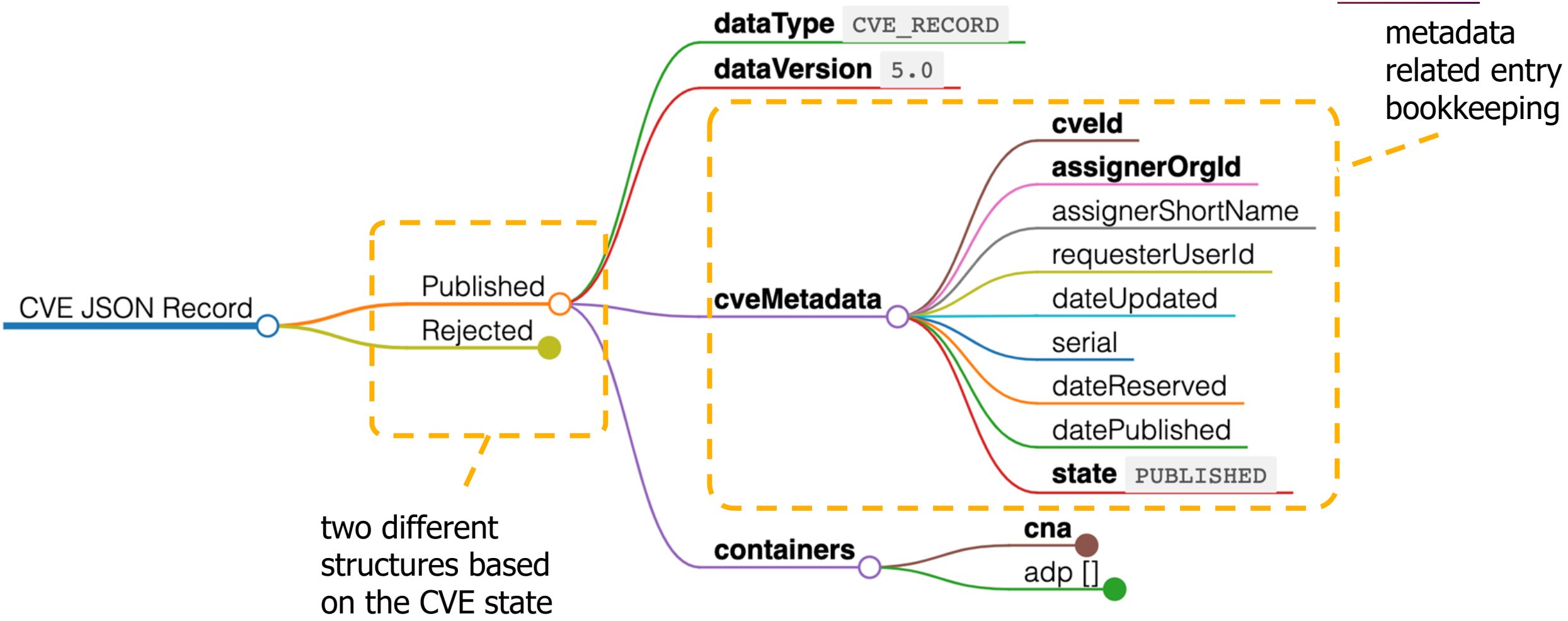
- dates
- credits
- scores
- translations
- type of vulnerability
- solutions
- impact
- exploits

Why JSON?

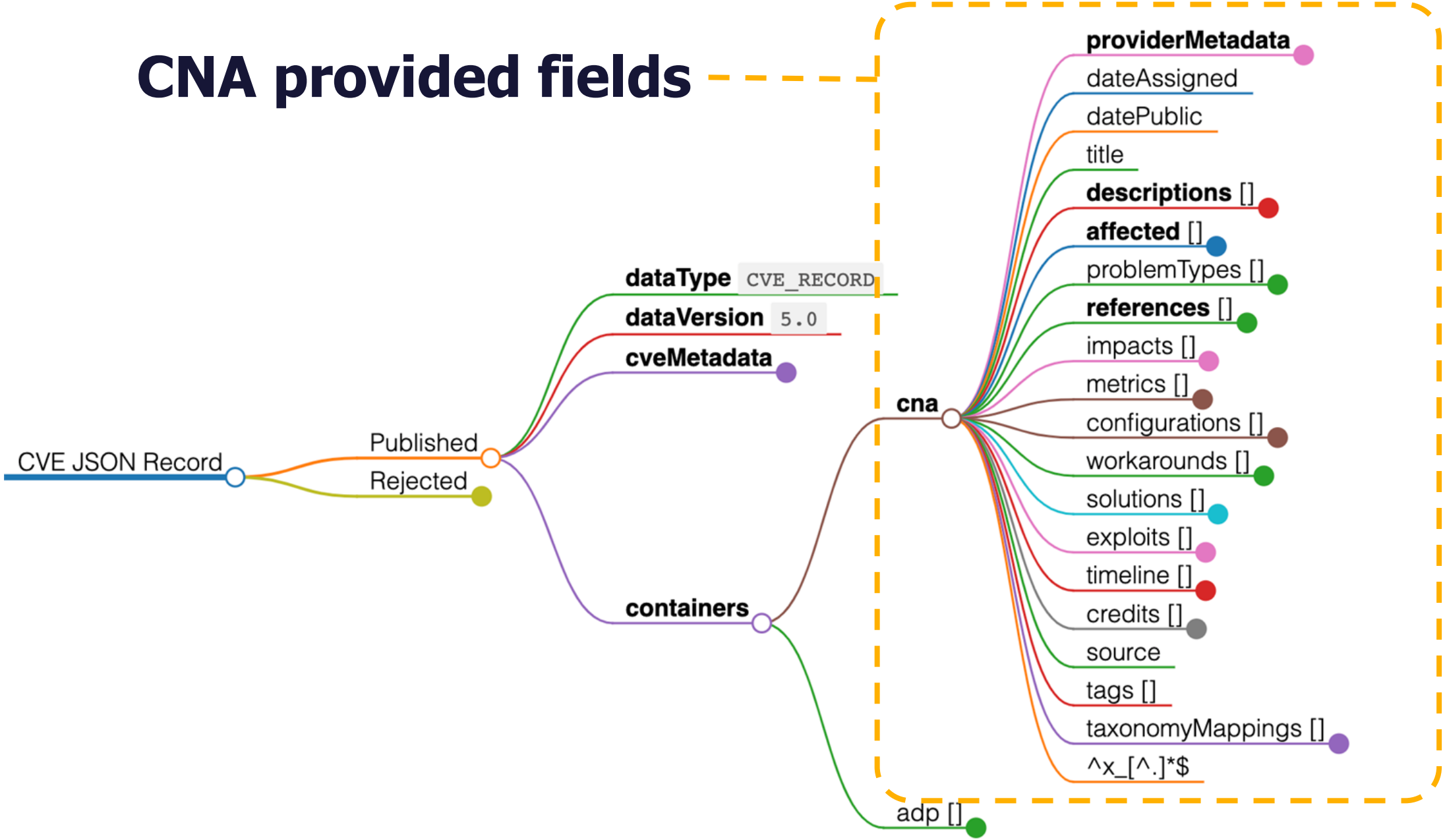


- **JavaScript Object Notation (JSON) is a lightweight data-interchange format**
- **Easy for humans to understand, organize data**
- **It is easy for machines to encode, parse, validate, and use**
- **A JSON Schema to codify rules:**
  - data types (strings, numbers, dates, arrays)
  - required fields
  - allowed values and patterns (minimum, maximum, regex)

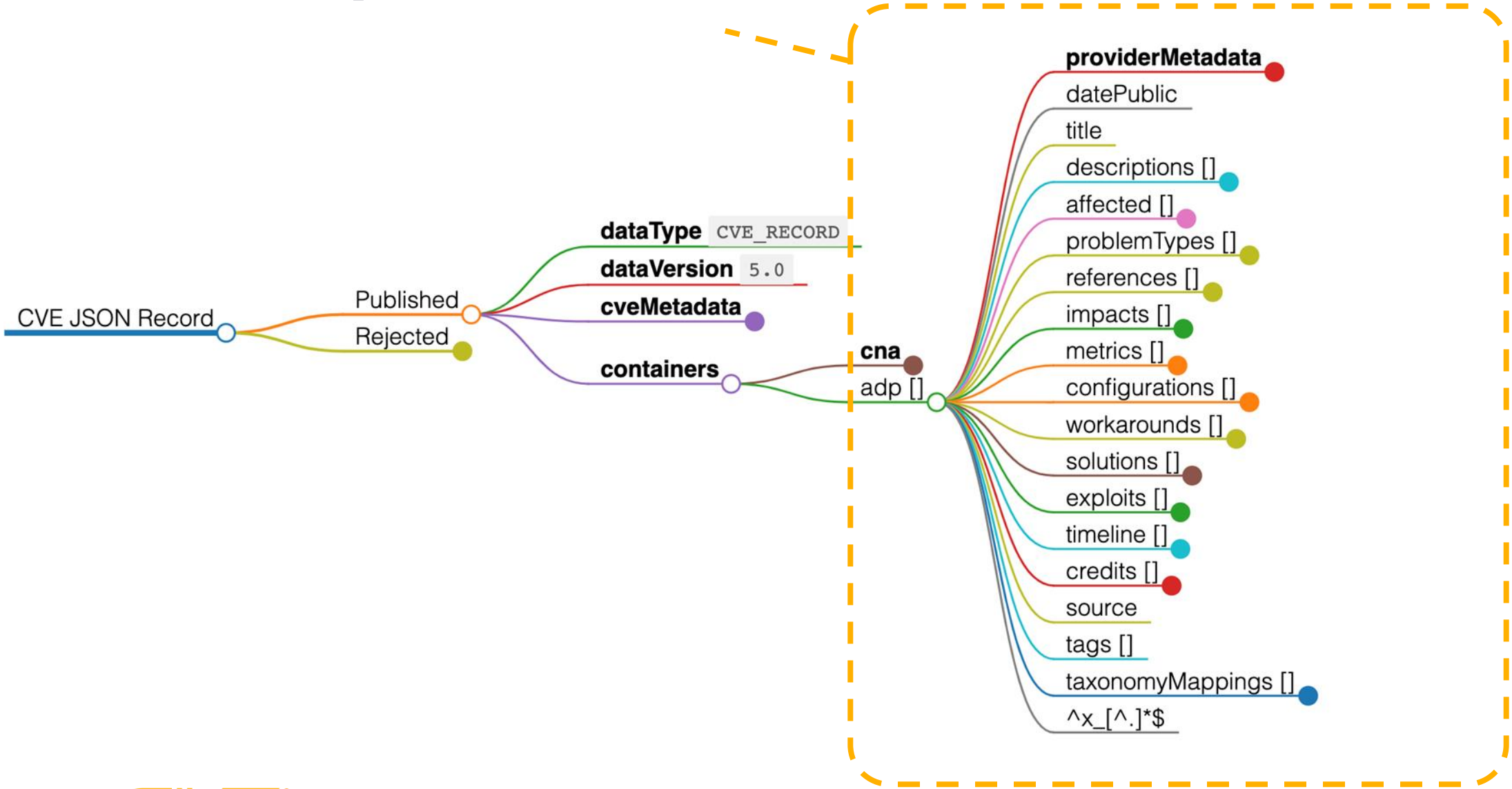
JSON Structure: Overview

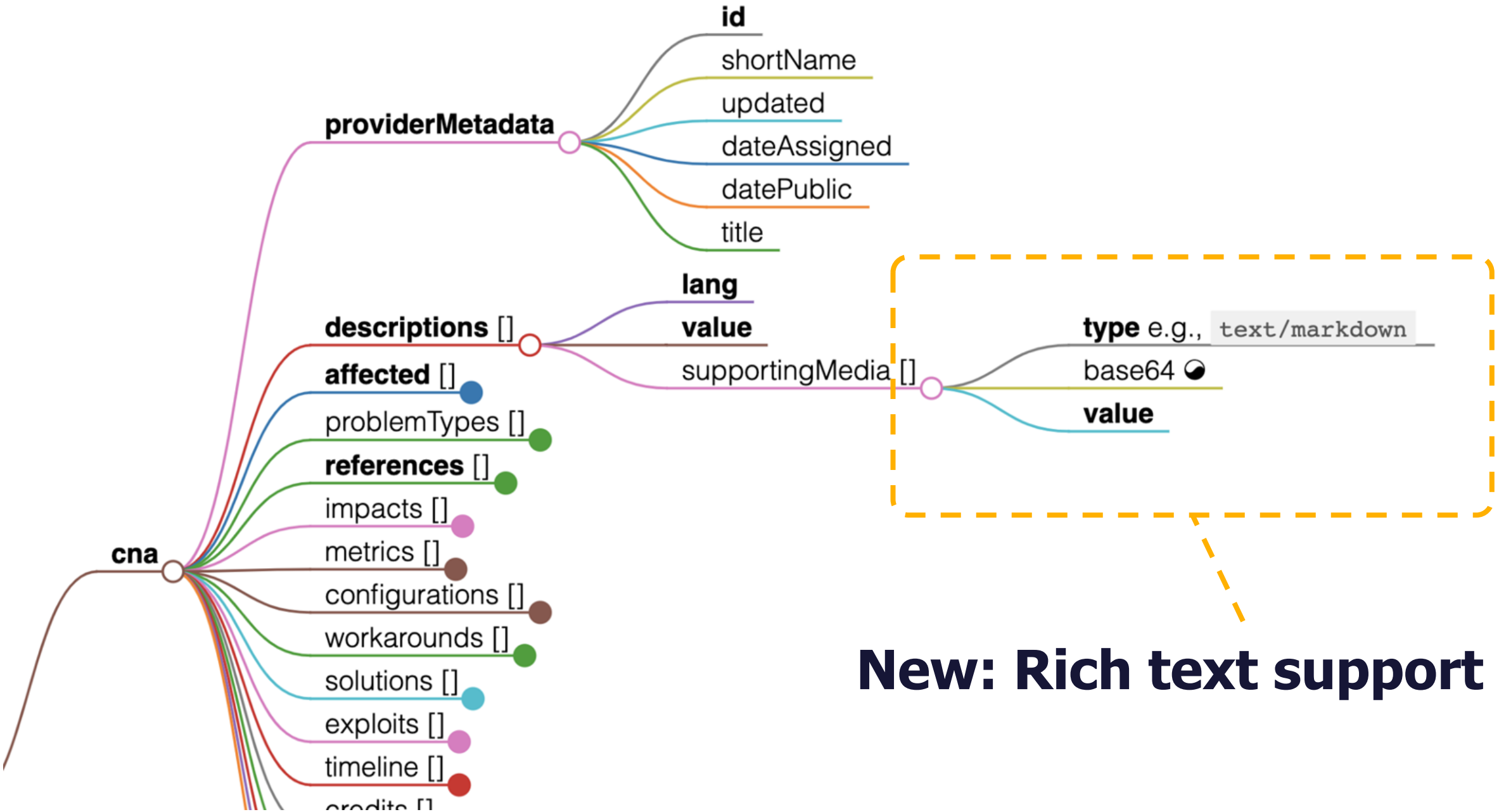


CNA provided fields



New: ADP provided fields

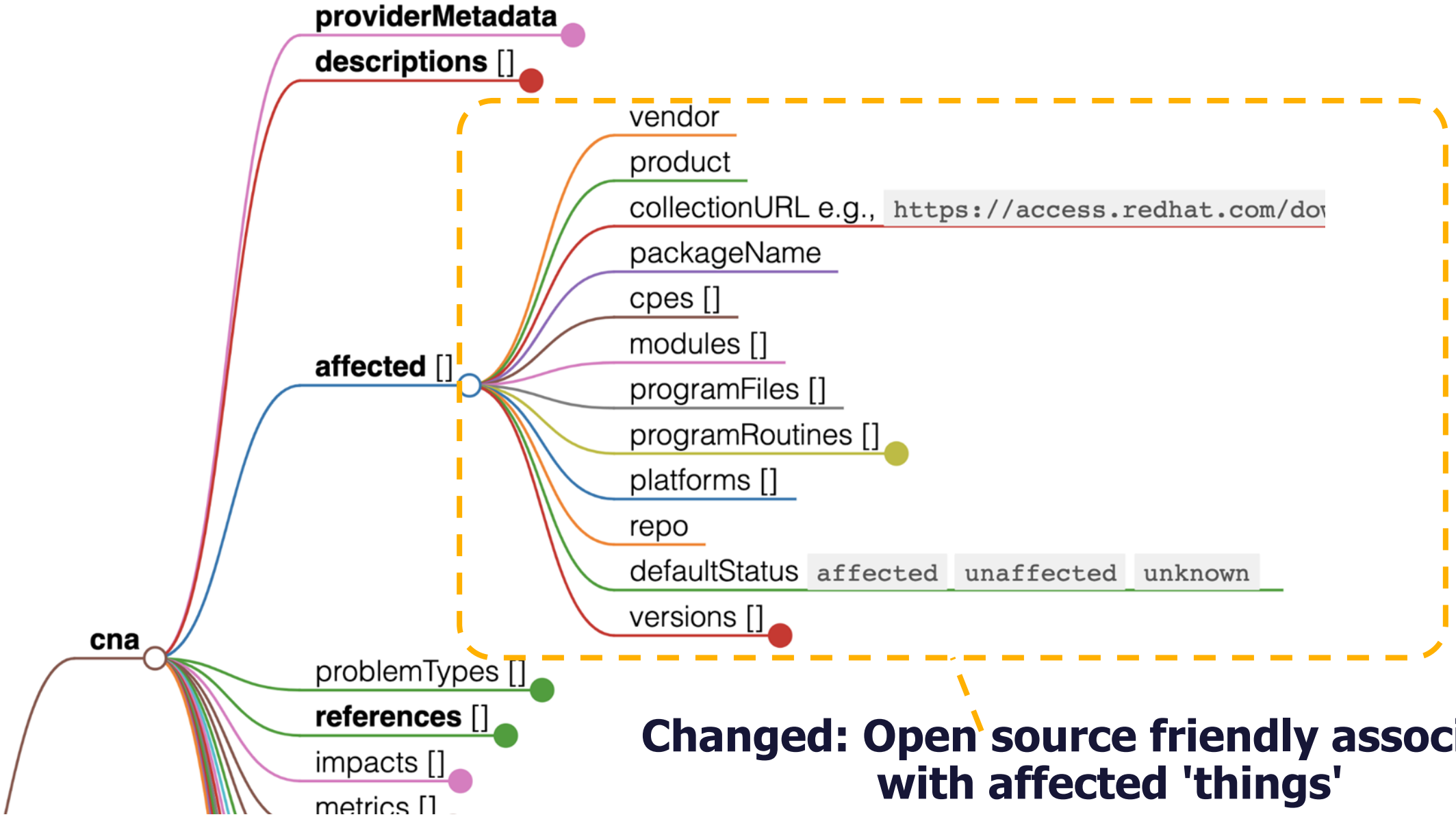




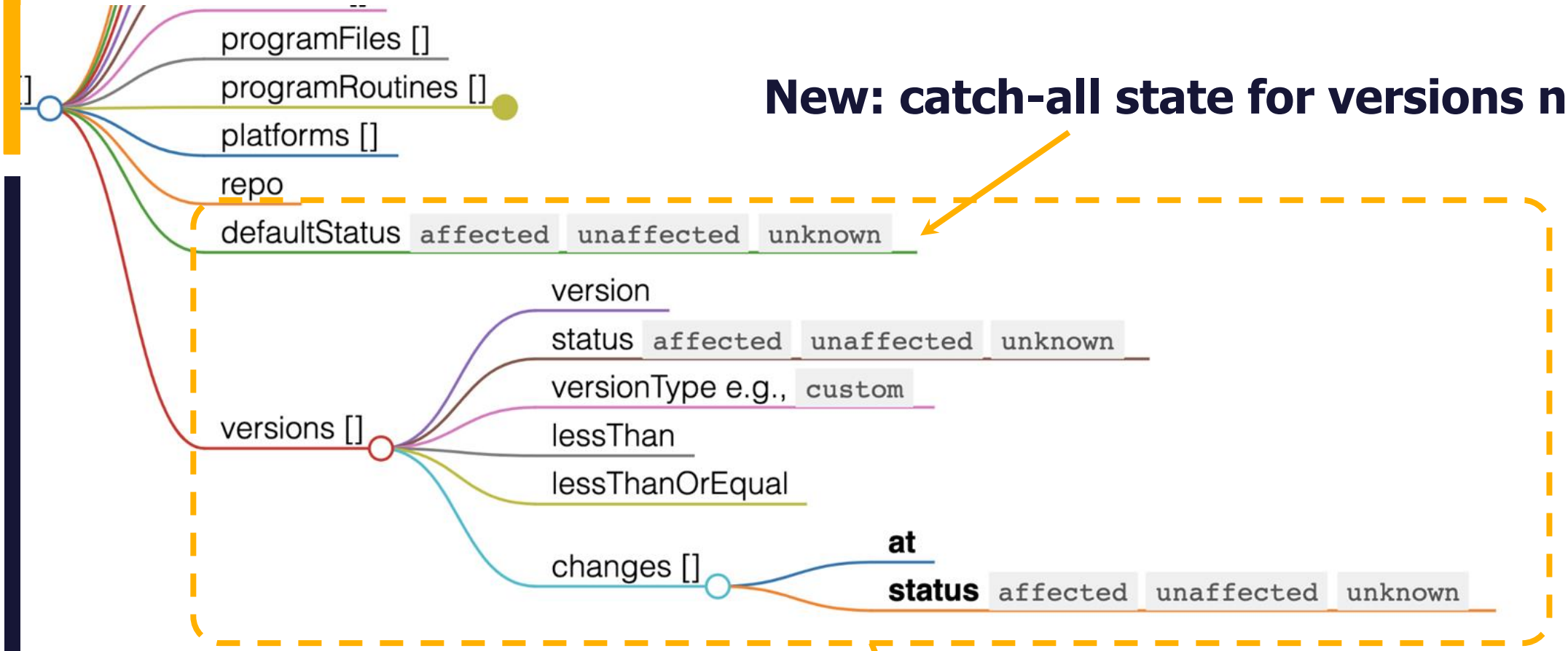
New: Rich text support

```
      "descriptions": [
        {
          "lang": "en",
          "value": "OS Command Injection vulnerability parseFilename function of example.php in the Web Management Interface of Example.org Example Enterprise on Windows, macOS, and XT-4500 allows remote unauthenticated attackers to escalate privileges. This issue affects: 1.0 versions before 1.0.6, 2.1 versions from 2.16 until 2.1.9.",
        },
        {
          "lang": "eo",
          "value": "OS-komand-injekta vundebleco parseFilename funkcio de example.php en la Web Administrado-Interfaco de Example.org Example Enterprise ĉe Windows, macOS kaj XT-4500 permesas al malproksimaj neaŭtentikigitaj atakantoj eskaladi privilegiojn. Ĉi tiu afero efikas: 1.0-versioj antaŭ 1.0.6, 2.1-versioj de 2.16 ĝis 2.1.9.",
        }
      ],
```

```
    "descriptions": [
      {
        "lang": "en",
        "value": "In getProcessRecordLocked of ActivityManagerService.java isolated apps are not handled correctly. This could lead to local escalation of privilege with no additional execution privileges needed. User interaction is not needed for exploitation.\n\n Product: Android\nVersions: Android-8.0, Android-8.1, Android-9, and Android-10\nAndroid ID: A-140055304",
        "supportingMedia": [
          {
            "type": "text/html",
            "base64": false,
            "value": "In <tt>getProcessRecordLocked</tt> of <tt>ActivityManagerService.java</tt> isolated apps are not handled correctly. This could lead to local escalation of privilege with no additional execution privileges needed. User interaction is not needed for exploitation.<br><br><b>Product</b>: Android<br><b>Versions</b>: Android-8.0, Android-8.1, Android-9, and Android-10<br><b>Android ID</b>: A-140055304"
          }
        ]
      }
    ],
```



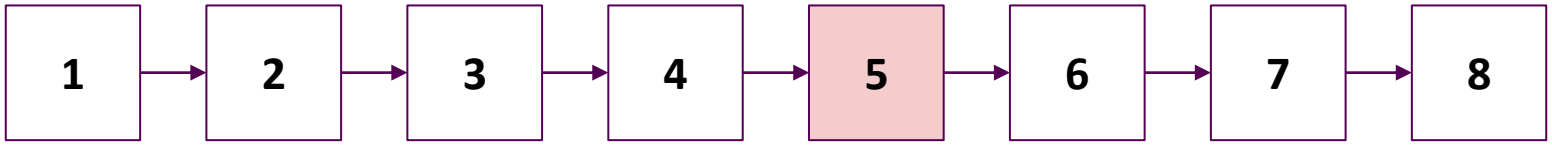
Changed: Open source friendly association with affected 'things'



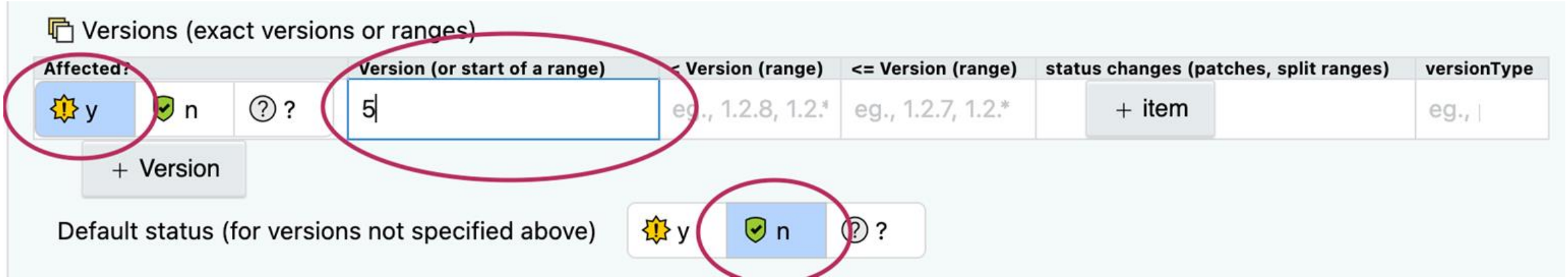
New: catch-all state for versions not listed

Changed: automation friendly listing of version or range of versions

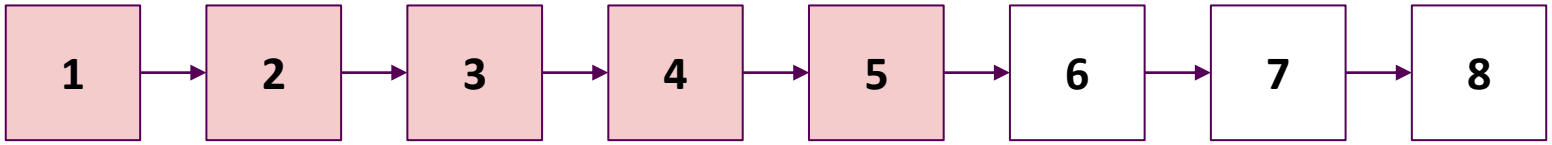
Software releases:



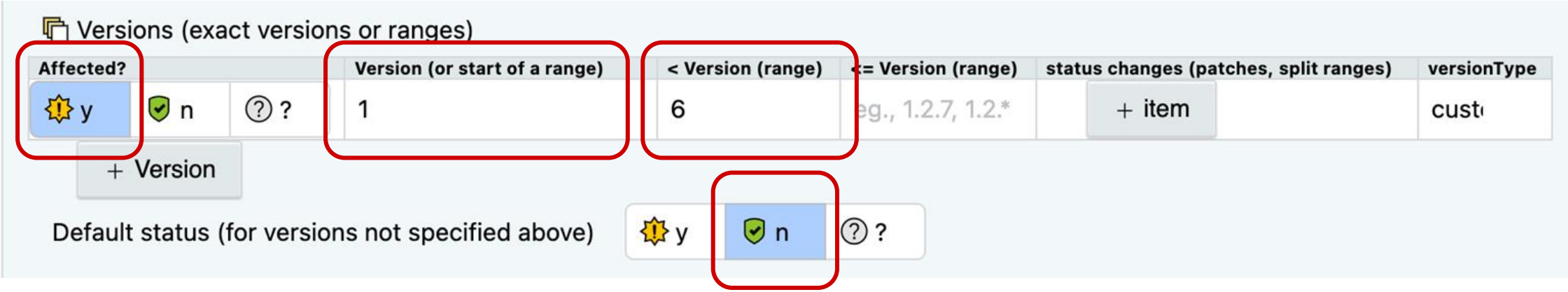

```
affected:
      versions:
            version: 5, status:
affected
      defaultStatus: unaffected
```



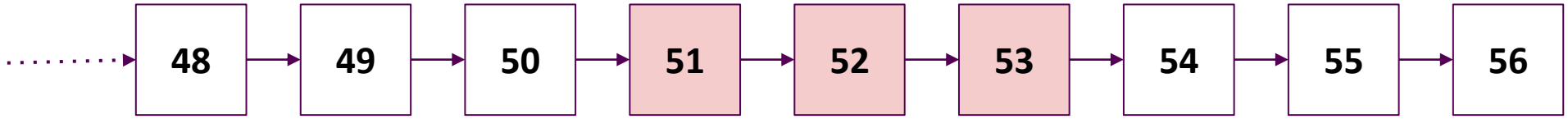
Software releases:




```
affected:
      versions:
            version: 1, lessThan: 6,
      status: affected, versionType: custom
      defaultStatus: unaffected
```

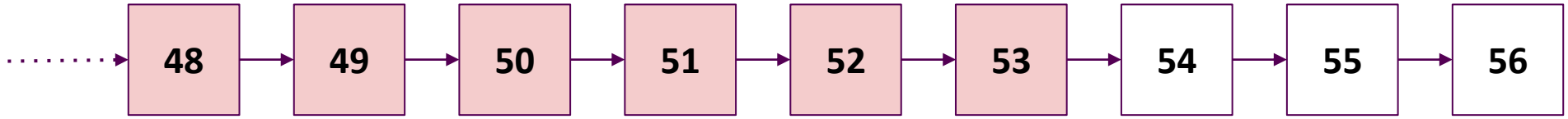


Software releases:

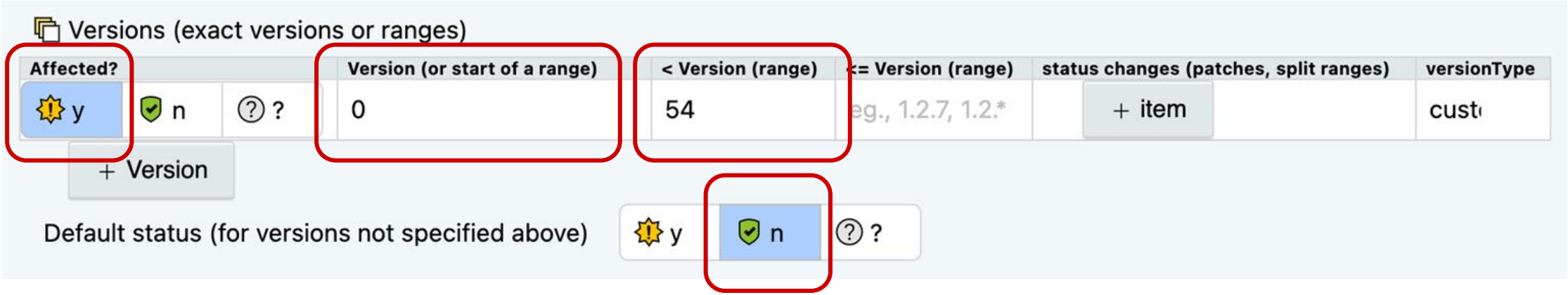



```
affected:
      versions:
            version: 51, lessThan: 53,
      status: affected, versionType: custom
      defaultStatus: unaffected
```

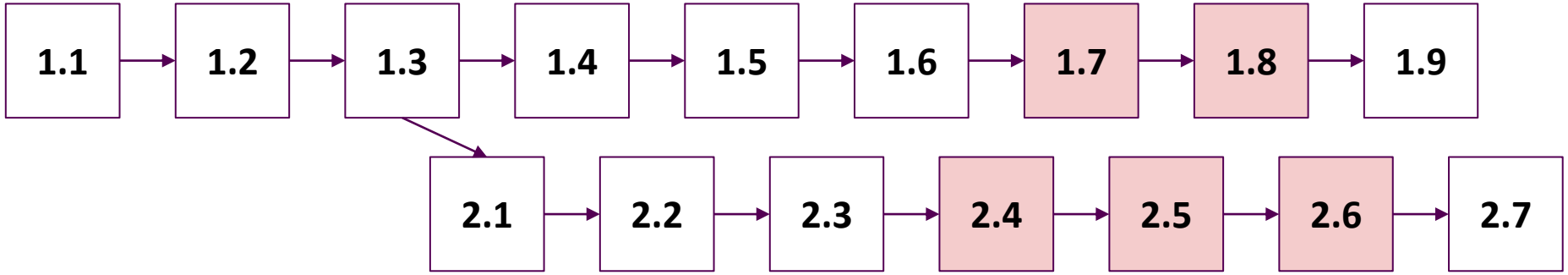
Software releases:




```
affected:
      versions:
            version: 0, lessThan: 53,
      status: affected, versionType: custom
      defaultStatus: unaffected
```



Software branches:



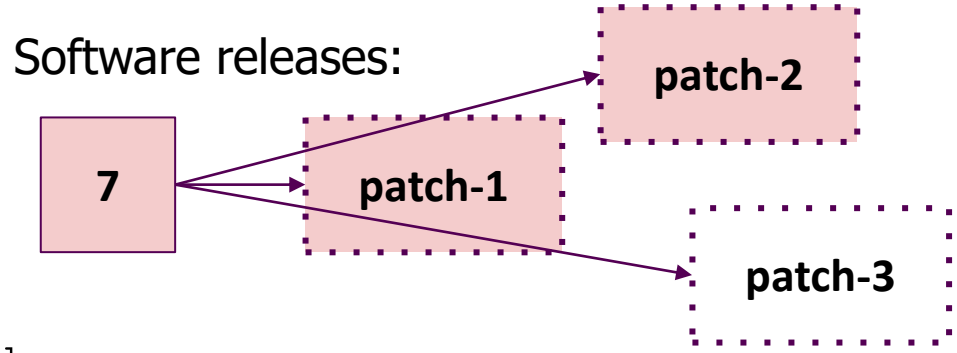

```
affected:
      versions:
            version: 1.7, lessThan: 1.9, status: affected
            version: 2.4, lessThan: 2.7, status: affected
      defaultStatus: unaffected
```

Versions (exact versions or ranges)

Affected?	Version (or start of a range)	< Version (range)	<= Version (range)	status changes (patches, split ranges)	versionType	
y / n / ?	1.7	1.8	eg., 1.2.7, 1.2.*	+ item	cust	
y / n / ?	2.4	2.7	eg., 1.2.7, 1.2.*	+ item	cust	

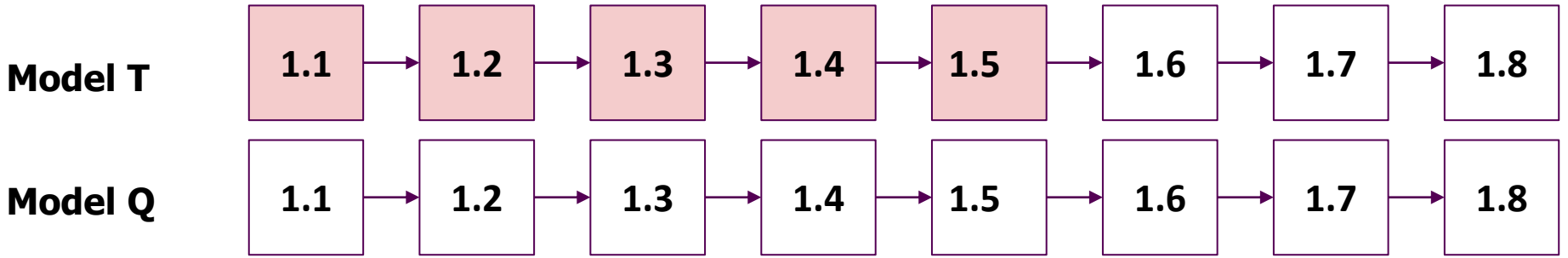
+ Version

Default status (for versions not specified above) y / n / ?



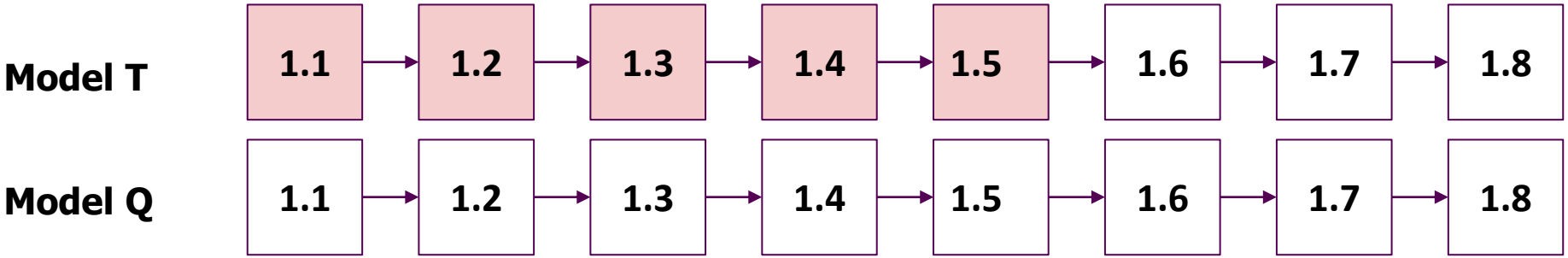

```
affected:
      versions:
            version: 7, status: affected
            changes:
          at: patch-3, status: unaffected
      defaultStatus: unaffected
```

Software releases:



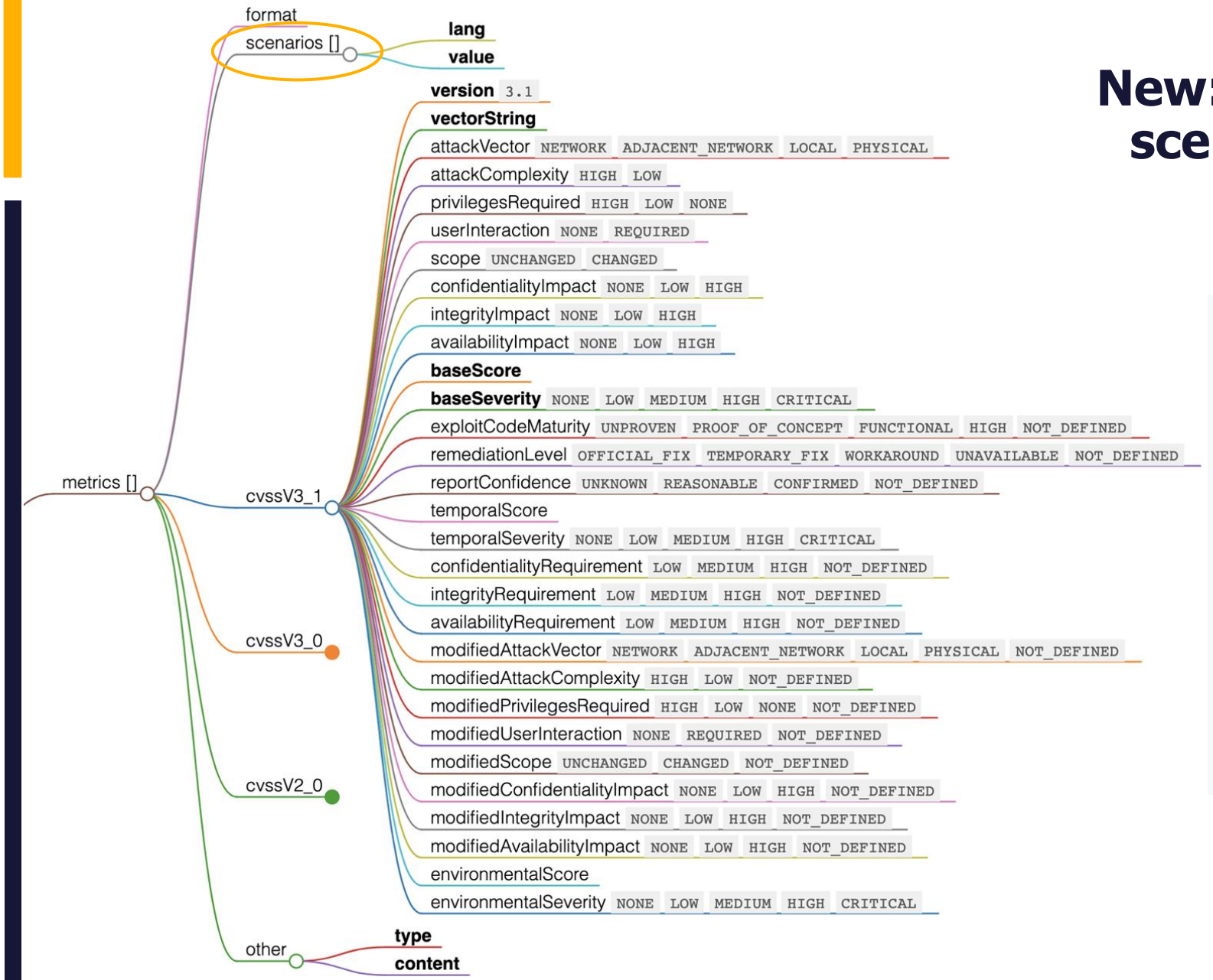

```
affected:
      platforms: [ Model T ]
      versions:
            version: 1.1, lessThan: 1.6,
      status: affected, versionType: custom
      defaultStatus: unaffected
```

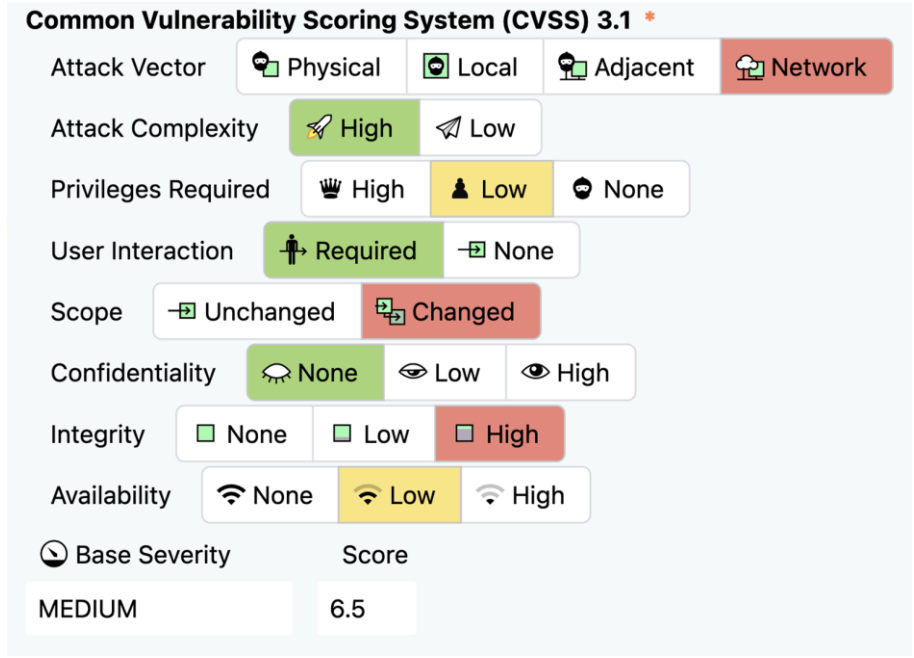
Software releases:



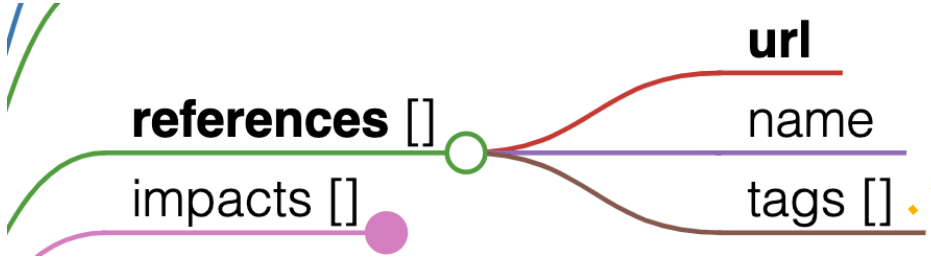

```
affected:
• platforms: [ Model T ]
     versions:
          version: 1.1, lessThan: 1.6,
     status: affected, versionType: custom
     defaultStatus: unaffected
• platforms: [ Model Q ]
     versions:
          version: 1.1, lessThan: *,
     status: unaffected, versionType: custom
     defaultStatus: unaffected
```

New: support for multiple scenario-based scoring





New: Reference tags



`broken-link` `customer-entitlement` `exploit` `government-resource` `issue-tracking` `mailing-list` `mitigation` `not-applicable` `patch` `permissions-required` `media-coverage` `product` `related` `release-notes` `signature` `technical-description` `third-party-advisory` `vendor-advisory` `vdb-entry`

New: CVE tags

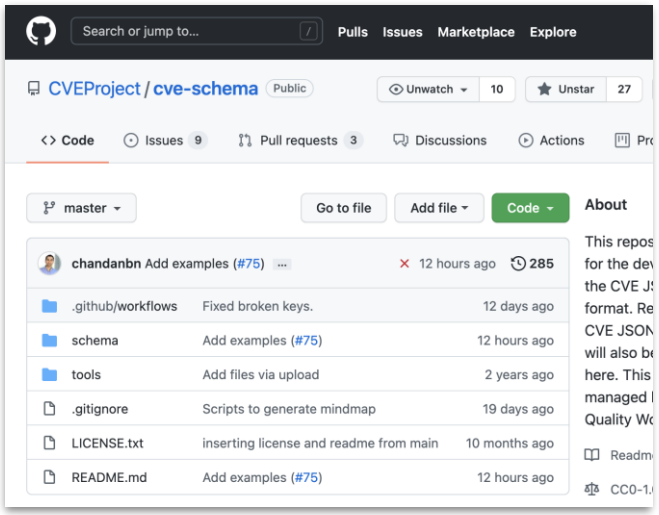
`unsupported-when-assigned` (CNA only)
`exclusively-hosted-service` (CNA only)
`disputed` (CNA or ADP)

New: Credit types

`finder` `reporter` `analyst` `coordinator` `remediation developer` `remediation reviewer` `remediation verifier` `tool` `sponsor` `other`

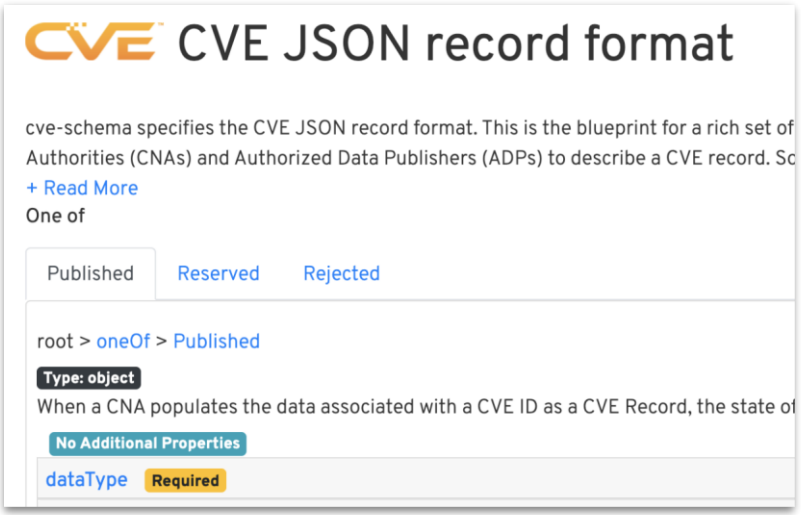
Resources

Git repo, issue tracking



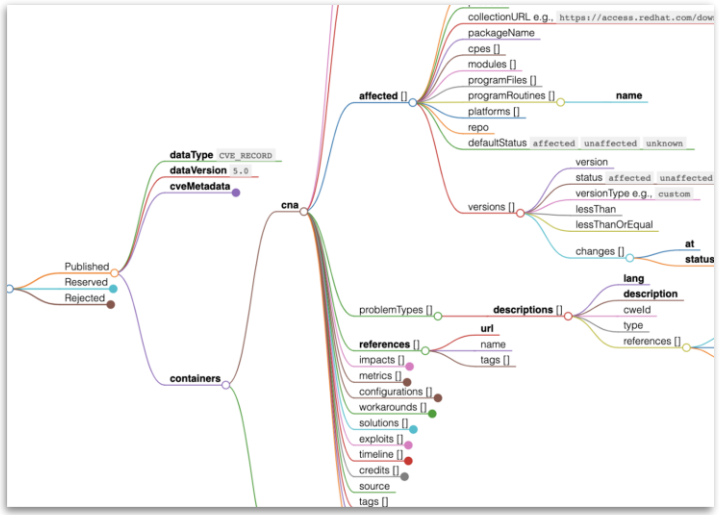
github.com/CVEProject/cve-schema

Schema documentation



cveproject.github.io/cve-schema/schema/v5.0/docs/

Mindmap



cveproject.github.io/cve-schema/schema/v5.0/docs/mindmap

Break until 1:00 p.m. EDT

How to Get a CVE Services Account (Dave Morse)

What is a CVE Services Account?

- **CVE Services access requires authentication**
  - Except for some limited API use for general public data retrieval
- **Account Types**
  - Organizational Admin (OA)
    - Able to use the CVE Services API to create/update CVEs
    - Creates and Manages users within an organization (CNA)
  - User
    - Able to use the CVE Services API to create/update CVEs

CVE Services Account Requirements

- **Requirements**
  - Must be an authorized user/administrator for an active CNA
  - Must agree to the terms (link)
- ***Note: "group" accounts - special, limited conditions**
  - Requires additional agreements between CNA and CVE Program
  - Requires additional tracking of user activity by CNA

Where CVE Services Accounts are Used

Reserve CVE IDs	**IDR**
Submit CVE Records	**RSUS** (*with JSON 5.0 only*)
User Registry (CNA manages its users)	**IDR**

NOTE: Record Submission via the GitHub CVEList Pilot or via CVE Request Web form are *JSON 4.0 only*.
NOTE: CVE Services has testing and production instances. Separate credentials are required for each.

Requesting a CVE Services Account

- **Requests must go through a CNA's Root**
  - Roots confirm CNA's active status
  - Roots provide CNAs with guidance for next steps
- **Once a Root confirms a CNA for CVE Services accounts**
  - CNA completes application for an OA account
  - Secretariat reviews application
  - Secretariat generates credentials for CNA OA

Managing CVE Services Accounts

- **Secretariat has overall management responsibility**
  - Generates OA accounts
  - Resets OA credentials
  - Responds to misuse or other issues
- **CNA OAs manage their organization's user accounts**
  - API endpoints for user creation, update
  - Each of the three clients is both an IDR client and an RSUS client
  - Two of the supported clients are GUI for entering vulnerability details

CVE Record Workflow Tutorial
(Art Manion)

CVE Workflow

- **Implicit Actions**
  - Be a CNA
  - Have a CVE-worthy vulnerability
  - Get an organization administrator account
- **Explicit Review**
  - Manage users
  - Reserve CVE ID
  - Create CVE Record content in JSON 5.0
  - Publish
  - Update
  - Reject

CVE Clients
(Art Manion)

CVE Clients

- **Vulnogram**
  - Live: https://vulnogram.github.io/, use the CVE 5.0 (beta) tab, uses 0.1.0-dev branch
  - Source: https://github.com/Vulnogram/Vulnogram
- **cveClient**
  - Live: https://certcc.github.io/cveClient/
  - Source: https://github.com/CERTCC/cveClient
- **cvelib**
  - Live: https://github.com/RedHatProductSecurity/cvelib
- **CVE.js**
  - Source: https://github.com/xdrr/cve.js
  - Not reviewed

CVE Client Comparison

Project	Language	License	User management	Reservation	Editing	Submission
Vulnogram	JavaScript, Node.js	MIT	Yes	Yes	Yes	Yes
cveClient	JavaScript	MIT+CMU	Yes	Yes	Yes	Yes
cvelib	Python 3	MIT	Yes	Yes	No	Yes
CVE.js	JavaScript (ECMAScript 6?)	MIT	Yes	Yes	No	Yes

CVE Clients: Additional Information

- **Security concerns about browser-based JavaScript web applications**
  - Where are my API keys? https://github.com/CERTCC/cveClient/blob/main/RISKS.md
  - Dependencies: https://github.com/CERTCC/cveClient/blob/main/README.md
- **Does the client support the full API?**
  - Use Services 2.1 API specification
  - prod: https://cveawg.mitre.org/api-docs/
  - test: https://cveawg-test.mitre.org/api-docs/
- **Demonstration scripts for cveclient**
  - https://github.com/zmanion/CVE

Closing Remarks
(Chris Levendis)

The mission of the CVE Program is to identify, define, and catalog publicly disclosed cybersecurity vulnerabilities.

There is one CVE Record for each vulnerability in the catalog. The vulnerabilities are discovered then assigned and published by organizations from around the world that have partnered with the CVE Program. Partners publish CVE Records to communicate consistent descriptions of vulnerabilities. Information technology and cybersecurity professionals use CVE Records to ensure they are discussing the same issue, and to coordinate their efforts to prioritize and address the vulnerabilities.

Learn more www.cve.org