



CVE management on Red Hat Advanced Cluster Security



Table of contents

Executive summary	3
Red Hat’s approach to vulnerability management	3
Red Hat vulnerability assessment process	3
Enhanced security strategy for enterprise workloads	4
Contextual guidance with CVSS and EPSS	5
CVE detection and scanning in Red Hat Advanced Cluster Security	5
Red Hat security data	6
Red Hat Advanced Cluster Security vulnerability scanners	6
StackRox Scanner	6
Scanner V4	6
Security data sources feeding Red Hat Advanced Cluster Security	7
Streamlining CVE management in Red Hat Advanced Cluster Security	7
Isolating platform CVEs for actionable focus	7
Understanding and utilizing Red Hat VEX files	9
Example workflow for CVE triage using VEX	13
Classification and operational response to platform CVEs	13
Remediable risks	14
a. Fix available	14
b. Workaround provided	14
c. Defer CVEs with justification via Ansible Playbooks	14
False positives	14
a. OpenShift not impacted due to usage/configuration	15
b. Inherited package status: OpenShift depends on Red Hat Enterprise Linux	15
c. Fix exists in RHSA, but CVE still flagged	15
d. Mark as false positive via Ansible Playbooks	16
Residual risks	16
a. CVEs without fix or workaround	16
Conclusion	16
Appendices	16
a. Cleaning vulnerability report using Excel pivot table	18
b. Clean vulnerability report using the sample VBA macros	19
c. Examples of reading VEX files	24

CVE management on Red Hat Advanced Cluster Security

Executive summary

As Kubernetes adoption accelerates across industries, enterprises running Red Hat® OpenShift® increasingly face the challenge of managing a growing volume of platform common vulnerability exposures (CVEs).

Unlike traditional virtual machine (VM) environments, Red Hat OpenShift's architecture combines immutable operating systems (OSes), declarative workloads, and tightly integrated platform components—all of which complicate vulnerability management. Without the right contextual tools, security teams risk becoming overwhelmed by CVEs that may not be applicable, actionable, or truly exploitable in their environment.

Red Hat Advanced Cluster Security for Kubernetes offers Kubernetes-native security tools by providing not just vulnerability scanning, but also policy-powered enforcement, Kubernetes context awareness, and platform integration. However, even with Red Hat Advanced Cluster Security, effective CVE triage remains complex due to differences in CVE data sources, upstream vulnerability scoring, platform dependencies, and inherited Red Hat Enterprise Linux® components.

This detail introduces a structured approach to streamlining platform CVE management on Red Hat OpenShift clusters using Red Hat Advanced Cluster Security, built on 4 pillars:

1. Red Hat's approach to vulnerability assessment.
2. Vulnerability scanning and data sources in Red Hat Advanced Cluster Security.
3. Streamlining CVE triage with Vulnerability Exploitability eXchange (VEX) context.
4. Actionable classification and operational response.

Red Hat's approach to vulnerability management

Red Hat vulnerability assessment process

Red Hat employs a comprehensive vulnerability assessment process anchored by its [severity rating system](#), which classifies vulnerabilities into 4 distinct categories:

- ▶ **Critical.** Flaws that could allow remote code execution or significant impact with no user interaction.
- ▶ **Important.** Issues that may lead to system compromise, but typically require some level of user involvement.
- ▶ **Moderate.** Flaws that pose limited risk and often require specific conditions to be exploitable.
- ▶ **Low.** Minor vulnerabilities with minimal impact or low likelihood of exploitation.

Red Hat takes a proactive stance when it comes to critical and important CVEs. Regardless of a product's lifecycle phase—whether it is in full support, maintenance, or extended life—these high-severity issues are always considered in scope and are addressed accordingly. Fixes for such vulnerabilities are often delivered outside scheduled release cycles, especially when urgency or risk levels demand it.

In 2023, the average time to fix critical CVEs was 9 days, which improved to 7 days in 2024—a 22% faster response time. For important CVEs, response times improved from 48 days to 31 days, a 35% increase in speed. Notably, over half of all critical CVEs were patched within a week, with some fixed on the same day they became public.¹

For moderate CVEs, Red Hat takes a case-by-case approach. These vulnerabilities are typically addressed in scheduled major or minor releases, but Red Hat will asynchronously fix CVEs which are known exploits.

For low CVEs, Red Hat remediates them with the same urgency as general bug fixes rather than security escalations. They are often not fixed unless they are bundled with higher-severity updates or if exploitation is reported.

Importantly, if Red Hat receives evidence of active exploitation of a vulnerability—regardless of its initial severity—it will prioritize remediation across its product portfolio.

For further insight into Red Hat's transparent approach and response timelines, refer to the [An Open Approach to Vulnerability Management: Red Hat's Methodology](#) and the [Product Security Risk Report](#).

Enhanced security strategy for enterprise workloads

To meet the evolving security needs of enterprise environments, Red Hat has introduced a 2-part security-enhancement strategy:

- 1.** For Red Hat Enterprise Linux, Red Hat is expanding Extended Update Support (EUS) and Extended Lifecycle Support (ELS) to include critical, important, and moderate CVEs with a Common Vulnerability Scoring System (CVSS) score of 7.0 or higher. This gives organizations broader coverage for high-impact vulnerabilities in regulated and security-sensitive industries.
- 2.** Red Hat has introduced the Red Hat Enterprise Linux Security Select Add-On, providing customers with greater flexibility to meet the dynamic security requirements of today's operational landscape. With this add-on:
 - a.** Red Hat commits to a 90-day turnaround for fixes and patches.
 - b.** Customers receive an initial 10-pack of nonstandard patches, with the option to purchase additional fixes as needed.

The Security Select Add-On is available to premium Red Hat Enterprise Linux subscribers who also have ELS, EUS, or Enhanced Extended Update Support (EEUS).

Additionally, purchasing the Security Select Add-On requires customers to have a dedicated security Technical Account Manager (TAM). If a security TAM is not already in place, they must be acquired to purchase the Security Select Add-On.

More information is available in the [Red Hat Enterprise Linux Security Select Add-On](#) overview.

¹ Red Hat detail. "[Red Hat Product Security Risk Report 2024](#)," Red Hat, April 11 Apr. 2025.

Contextual guidance with CVSS and Exploit Prediction Scoring System (EPSS)

Red Hat supports vulnerability management by combining impact-based and likelihood-based scoring systems to guide customer prioritization:

- ▶ **CVSS.** Red Hat applies its own CVSS base scores based on the [CVSS v3 standard](#), tailored to reflect product-specific mitigations, configurations, and hardening (e.g., SELinux). These scores and severity ratings may differ from those published in the National Vulnerability Database (NVD) and are recommended for more accurate risk assessments of Red Hat products.
- ▶ **EPSS.** EPSS estimates the likelihood (between 0% and 100%) of a vulnerability being exploited in the wild within the next 30 days. The score is produced by the [EPSS framework](#). EPSS is especially useful for prioritizing remediation of moderate and low severity CVEs. Red Hat Advanced Cluster Security 4.7 and later surfaces EPSS scores natively within the CVE view to support exploitability-based triage.

This dual approach helps organizations make risk-informed decisions, balancing technical severity with real-world exploitability.

CVE detection and scanning in Red Hat Advanced Cluster Security

CVE triaging challenges in Kubernetes environments

In Kubernetes environments, security teams often face overwhelming volumes of CVEs detected across diverse layers: user workloads, platform components, and underlying infrastructure. Most tools lack the built-in context-awareness needed to prioritize these vulnerabilities based on real business risk, technical relevance, and actual exploitability. This creates significant manual effort in triaging CVEs effectively.

Red Hat security data

Red Hat offers several machine-readable data formats to support vulnerability management, compliance automation, and supply chain transparency. **VEX documents** follow the [Common Security Advisory Framework \(CSAF\)](#) standard, allowing for standardized sharing of vulnerability information. Red Hat publishes both product-level security advisories and per-CVE VEX documents. Published in JavaScript Object Notation (JSON) format, VEX specifies affected or unaffected product states, including whether a fix is available or planned.

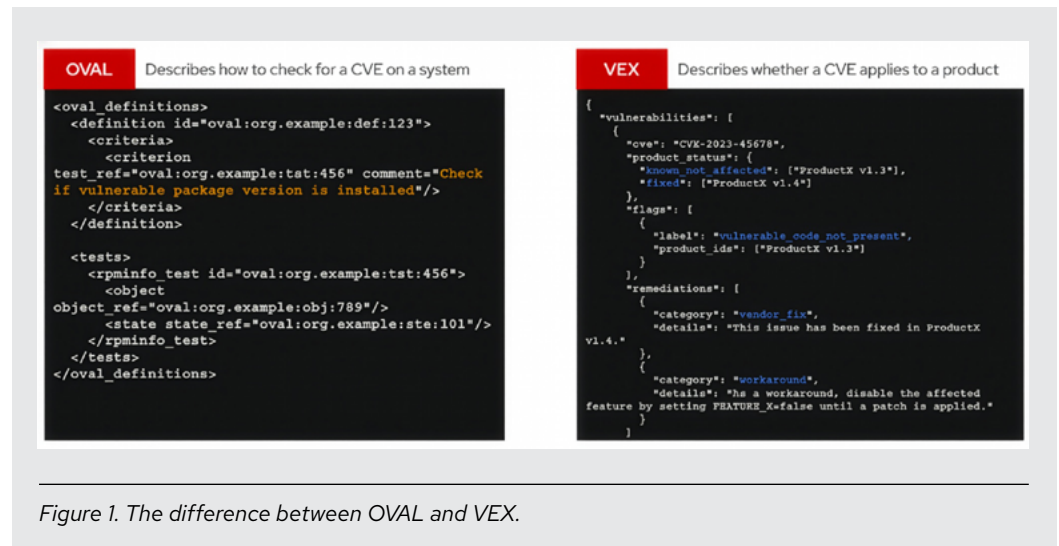
Open source vulnerability (OSV) records provide vulnerability metadata for Red Hat-maintained RPM packages and are derived from CSAF advisories. These records are made publicly available via OSV.dev and align with the OpenSSF's broader efforts to enhance software supply chain transparency. Customers are encouraged to consume these records through OSV.dev to integrate with software composition analysis (SCA) tools and software bill of materials (SBOM) validation workflows.

SBOM documents list all software components, versions, licenses, and provenance details associated with a Red Hat product or image. Provided in SPDX format, these SBOMs support secure software supply chain initiatives by improving visibility into component origins and dependencies.

Open Vulnerability and Assessment Language (OVAL) definitions, while now deprecated in favor of CSAF and VEX, are still available for legacy use cases. These XML-based definitions support compliance and vulnerability checks—particularly in tools such as Open Source Security Compliance Solution (OpenSCAP)—by mapping vulnerabilities to affected packages in Red Hat Enterprise Linux and other supported products. For complete system evaluation, users must assess all OVAL streams relevant to their installed products.

OVAL and VEX serve distinct purposes in vulnerability management. OVAL is a low-level, system-specific format that describes how to detect the presence of a vulnerability by checking if a known vulnerable package version is installed on a system. It is used in compliance tools such as OpenSCAP to scan systems for unpatched software.

In contrast, VEX operates at a higher level, indicating whether a product is affected by a CVE. It provides product-centric context such as known unaffected versions, fixed versions, and mitigation information like vendor fixes or workarounds. VEX also supports flags such as `vulnerable_code_not_present` to suppress false positives. This shift to VEX enables more accurate, scalable, and automation-friendly vulnerability reporting aligned with modern supply chain security practices.



Red Hat Advanced Cluster Security vulnerability scanners

StackRox Scanner

The StackRox Scanner is the default vulnerability scanner for Red Hat Advanced Cluster Security 4.7 and earlier versions. It is based on a fork of Clair v2 and serves as the primary tool for identifying vulnerabilities within container images.

However, as technology evolved, the StackRox Scanner has been deprecated and is now being phased out in favor of a more modern and accurate solution.

Scanner V4

Scanner V4, introduced in Red Hat Advanced Cluster Security 4.4, is built on Claircore and represents the next generation of image vulnerability scanning within Red Hat Advanced Cluster Security. It is the default scanner for new installations of Red Hat Advanced Cluster Security 4.8 and later versions.

This scanner offers improved accuracy for detecting vulnerabilities across container images and is capable of analyzing both OS packages and language-specific components. Scanner V4 was developed to bring consistency to vulnerability scanning across Red Hat products, including Red Hat Advanced Cluster Security and Red Hat Quay.

It also broadens support across both programming languages (such as Golang) and OSes (including Oracle Linux, SUSE Linux Enterprise, and Photon OS). This expanded coverage helps organizations identify risks more comprehensively and reliably.

Security data sources feeding Red Hat Advanced Cluster Security

Scanner V4 and StackRox Scanner consume a variety of upstream data feeds to identify and prioritize CVEs in container images.

StackRox Scanner uses structured data sources such as Red Hat OVAL to map CVEs to Red Hat packages, alongside vulnerability trackers from Alpine, Debian, Ubuntu, and Amazon Linux. To enrich vulnerability records, it references the NVD for supplemental metadata such as CVSS scores and descriptions. To address gaps caused by format inconsistencies or missing upstream data, StackRox maintains a manual entry database for Linux and NVD edge cases.

In contrast, Scanner V4 modernizes vulnerability ingestion by adopting more scalable and curated feeds. It sources vulnerability data from Red Hat VEX documents, which provide official Red Hat stances on CVEs and include references to [Red Hat Security Advisories \(RHSAs\)](#). For application-layer vulnerabilities in ecosystems such as Python, Java, JavaScript, Ruby, or Go, OSV.dev is used, often providing GitHub Security Advisory (GHSA) IDs. The NVD is still used to fill missing metadata when necessary.

For non-Red Hat base images, Scanner V4 pulls from distro-specific OVAL feeds (e.g., Ubuntu, SUSE, Oracle, Photon) and security trackers from Amazon Linux, Debian, and Alpine. A manual database in the upstream StackRox project remains available to handle exceptions and upstream inconsistencies.

This multisource architecture lets Red Hat Advanced Cluster Security balance coverage, accuracy, and suppression of false positives—empowering security teams with actionable and contextual vulnerability insights.

Streamlining CVE management in Red Hat Advanced Cluster Security

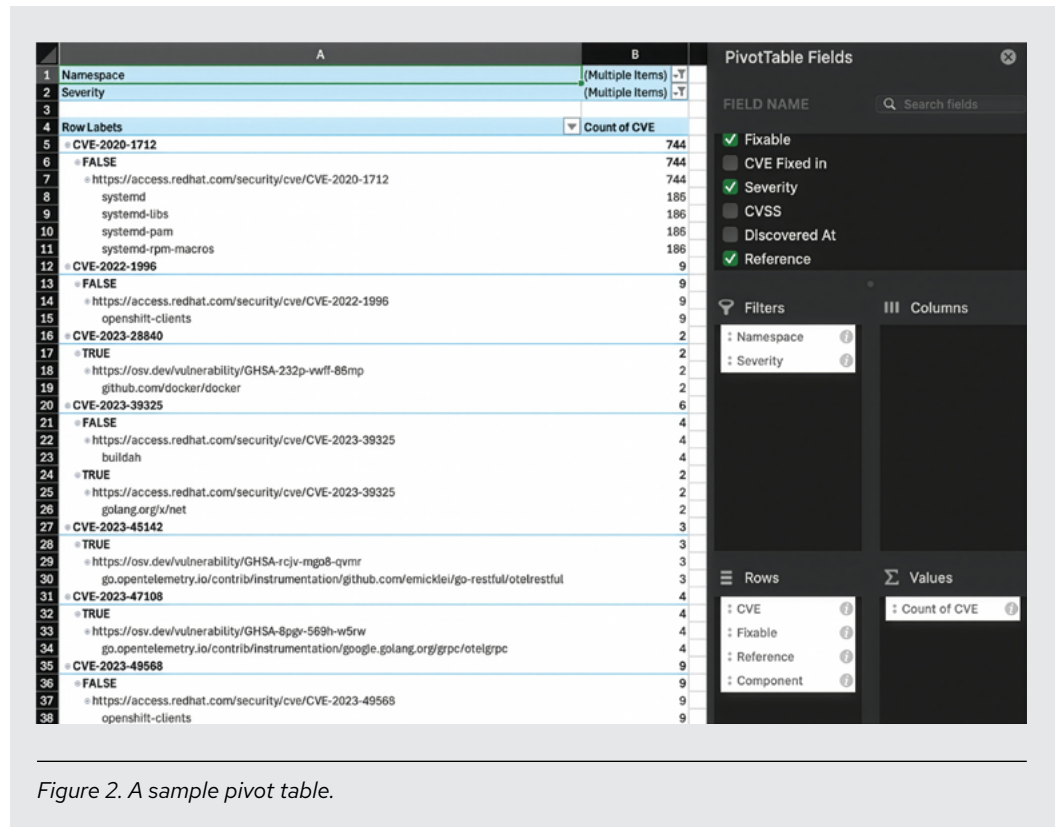
Isolating platform CVEs for actionable focus

Managing the flood of CVEs detected by Red Hat Advanced Cluster Security can be overwhelming, especially when trying to prioritize what truly matters across platform components. The **1st step** in streamlining this process is to download the CSV vulnerability report directly from the Red Hat Advanced Cluster Security console or from the scheduled email reports for your target cluster.

Once you have the report, the **2nd step** is to clean and filter the data by CVE ID. This can be done using Excel pivot tables or custom VBA macros to narrow down the list to only critical and important severity platform-related CVEs:

Option 1. Pivot table (refer to [appendix A](#) for detailed instructions)

- ▶ Filter by namespace and severity.
- ▶ Display rows for CVE, fixable status, reference link, and affected component.
- ▶ Use counts to break down the number of CVEs by category.



Option 2. VBA macros (refer to [appendix B](#) for detailed instructions)

- ▶ Automate pivot table creation using Excel macros.
- ▶ [Sample](#) script is available for reference.

For the **3rd step**, enrich the data using the Red Hat CVE Database and VEX files to determine whether a fix exists, if a workaround is available, or if it is a false positive.

And as a **final step**, categorize each CVE as remediable, false positive, or residual, and take action using Ansible Playbooks—such as deferring CVEs or marking false positives—to improve dashboard accuracy and reduce security team workload.

Understanding and utilizing Red Hat VEX files

Red Hat VEX files provide essential context for more effective CVE triage by helping security teams determine the actual relevance and impact of vulnerabilities on Red Hat products.

Metadata. VEX files contain detailed metadata for each vulnerability, including the CVE ID, CWE classification, and discovery date:

```
{
  "cve": "CVE-2023-49568",
  "cwe": {
    "id": "CWE-400",
    "name": "Uncontrolled Resource Consumption"
  },
  "discovery_date": "2024-01-12T00:00:00+00:00"
}
```

Additional descriptive information is provided through the **notes** object, summarizing the vulnerability, its impact, and Red Hat's position and evaluation of the vulnerability:

```
{
  "notes": [
    {
      "category": "description",
      "text": "A denial of service (DoS) vulnerability was found in the go library go-git. This issue may allow an attacker to perform denial of service attacks by providing specially crafted responses from a Git server, which can trigger resource exhaustion in go-git clients.",
      "title": "Vulnerability description"
    },
    {
      "category": "summary",
      "text": "go-git: Maliciously crafted Git server replies can cause DoS on go-git clients",
      "title": "Vulnerability summary"
    }
  ],
}
```

```
{
  "category": "other",
  "text": "This problem only affects the go implementation
and not the original git cli code. Applications using only
in-memory filesystems are not affected by this issue. Clients
should be limited to connect to only trusted git servers to
reduce the risk of compromise.",
  "title": "Statement"
},
{
  "category": "general",
  "text": "The CVSS score(s) listed for this vulnerability do
not reflect the associated product's status, and are included
for informational purposes to better understand the severity
of this vulnerability.",
  "title": "CVSS score applicability"
}
]
```

Product tree. The product tree maps the affected component to specific Red Hat products and their corresponding product IDs:

```
{
  "category": "default_component_of",
  "full_product_name": {
    "name": "kernel as a component of Red Hat Enterprise Linux 6",
    "product_id": "red_hat_enterprise_linux_6:kernel"
  },
  "product_reference": "kernel",
  "relates_to_product_reference": "red_hat_enterprise_linux_6"
}
```

Product status. This section indicates the vulnerability status across different Red Hat products—whether it is fixed, affected, not affected, or under investigation:

```
{
  "product_status": {
    "fixed": [],
    "known_affected": [],
    "known_not_affected": [],
    "under_investigation": []
  }
}
```

Justification flags. Justification flags explain why a product is unaffected by a CVE, such as `vulnerable_code_not_present`:

```
"flags": [
  {
    "label": "vulnerable_code_not_present",
    "product_ids": [
      "7Client-7.9.Z:kernel-headers-0:3.10.0-1160.99.1.el7.ppc64",
      "7Client-7.9.Z:kernel-headers-0:3.10.0-1160.99.1.el7.ppc64le",
      "..."
    ]
  }
]
```

Remediations section. This section provides actionable guidance including vendor fixes, workarounds, and no planned fix.

Vendor fix example:

```
{
  "category": "vendor_fix",
  "details": "For details on how to apply this update, which includes the changes described in this advisory, refer to:\n\nhttps://access.redhat.com/articles/11258\n\nThe system must be rebooted for this update to take effect.",
  "product_ids": [
    "7Server-7.4.AUS:kernel-0:3.10.0-693.112.1.el7.src",
    "7Server-7.4.AUS:kernel-0:3.10.0-693.112.1.el7.x86_64",
    "...",
  ],
  "url": "https://access.redhat.com/errata/RHSA-2023:4819"
}
```

Workaround example:

```
{
  "category": "workaround",
  "details": "Users can restrict the usage of gitRepo volumes in their cluster using policies such as `ValidatingAdmissionPolicy`. \n\nThe following CEL expression can be used as part of the policy to restrict the use of gitRepo volumes:\n\n~~~\nhas(object.spec.volumes)|| !object.spec.volumes.exists(v, has(v.gitRepo))\n~~~",
  "product_ids": [
    "...",
    "red_hat_ansible_automation_platform_2:automation-controller",
    "red_hat_openshift_container_platform_4:openshift4/cnf-tests-rhel8",
    "...",
  ]
}
```

No planned fixed example:

```
{
  "category": "no_fix_planned",
  "details": "Will not fix",
  "product_ids": [
    "red_hat_openshift_container_platform_4:systemd"
  ]
}
```

More details on VEX structure can be found in the [Red Hat Security Data Guidelines](#).

Example workflow for CVE triage using VEX

To triage a vulnerability using Red Hat's VEX data, start by identifying the affected package or component from the cleaned vulnerability report. For instance, you may observe a flagged component such as `systemd`, `openshift-clients`, or `libnetwork` in the report.

Next, open the relevant VEX JSON file and search for the flagged component. Locate the associated Red Hat product in the `product_tree` section, such as `red_hat_enterprise_linux_9` or `red_hat_openshift_container_platform_4`. For example, when searching for `systemd` in the context of CVE-2020-1712, you may find different product statuses depending on the platform.

Review the `product_status` object to determine Red Hat's official stance on the CVE. Key status values include `known_not_affected`, `known_affected`, or `fixed`. Pay attention to justification flags in the flags section such as `vulnerable_code_not_present`, which provide deeper insight into why a product may not be impacted. For CVE-2020-1712, Red Hat Enterprise Linux 9 is marked as `known_not_affected`, while Red Hat OpenShift is marked `known_affected`.

Further context is often provided in the `statement` and `remediations` sections. These sections may link to relevant RHSAs, offer workaround instructions, or clarify product behavior. For example, CVE-2023-49568 includes a note that the vulnerable Git functionality exists only in the Go-based implementation, not the native Git CLI.

With this information, classify the CVE based on these data points as a remediable risk, false positive, or residual risk. This will be covered in the next section.

Finally, take the appropriate action within Red Hat Advanced Cluster Security. False positives can be suppressed. Remediable risks should be deferred with justifications and workaround guidance. Residual risks should be monitored, with compensating controls applied or escalated via Red Hat Support if necessary.

Refer to examples in [appendix C](#).

Classification and operational response to platform CVEs

Effective classification and remediation of CVEs within Red Hat platforms, especially Red Hat OpenShift, requires structured interpretation of VEX data and operational discipline. CVEs can be categorized into actionable groups depending on the availability of fixes or mitigations.

Remediable risks

A CVE is considered a remediable risk when a fix is available or when a valid workaround is provided in the absence of a patch. This classification lets teams take proactive remediation steps or apply compensating controls while maintaining platform integrity.

a. Fix available

When a CVE is listed in a RHSA or marked as **fixed** in the VEX file, it indicates that Red Hat has released a patch for the issue. These CVEs should be prioritized for resolution. The recommended action is to upgrade to the latest recommended Red Hat OpenShift version that incorporates the fix.

If an immediate upgrade is not feasible due to operational constraints, organizations should apply compensating controls. These may include limiting external exposure, disabling the affected components if not in use, and enforcing stricter ingress/egress network policies. In Red Hat Advanced Cluster Security, security teams can defer the CVE with a clear justification that appropriate compensating controls are already in place.

b. Workaround provided

Some CVEs may be marked as **known_affected** or **no_fix_planned** in the VEX document but include a documented workaround. In such cases, the risk is still considered remediable as long as the workaround is feasible and effectively mitigates the threat. Security teams should apply the provided workaround promptly and defer the CVE in Red Hat Advanced Cluster Security, including a justification referencing the workaround.

Examples:

- ▶ **CVE-2023-49568.** Affects openshift-clients in Red Hat Enterprise Linux 9. No fix provided, but exposure can be limited to trusted Git servers.
- ▶ **CVE-2024-10220.** Mitigated by applying a ValidatingAdmissionPolicy to prevent gitRepo volume usage.

c. Defer CVEs with justification via Red Hat Ansible® Automation Platform playbooks

Organizations may need to temporarily defer certain CVEs, especially in tightly controlled environments or when applying a fix immediately would disrupt critical operations. Organizations can use Ansible Automation Platform playbooks to facilitate this process, allowing teams to send authenticated application programming interface (API) requests to Red Hat Advanced Cluster Security Central to defer selected CVEs. These deferrals can be configured with an expiry date and tied to future remediation milestones (for example, after an upgrade). The playbooks support token-based authentication and can be parameterized with a security focus using Ansible vault, protecting credential protection during automation runs. Sample playbooks can be accessed in this [GitHub Repository](#).

d. Recommended Red Hat OpenShift version strategy

Maintaining a stable and security-focused Red Hat OpenShift environment also requires careful version management. Red Hat recommends running either the current (N) or 1 minor release behind (N-1) version of Red Hat OpenShift to remain within the supported lifecycle and access critical patches. Within your selected minor version, it is essential to stay on the latest available z-stream release. For long-term stability and extended support options, favor even-numbered EUS versions such as Red Hat OpenShift 4.16 or 4.18. Plan your upgrade cycle to move between EUS versions, making sure there is minimal operational disruption while staying ahead of emerging security threats.

False positives

A CVE is classified as a false positive when it appears in vulnerability reports, but does not pose a real or exploitable risk within the Red Hat OpenShift environment. This may be due to nonapplicable usage, sandboxed execution, disabled features, or the component not being shipped at all. Red Hat VEX files help validate these cases through product status fields and justification flags, allowing teams to confidently suppress such findings and reduce alert fatigue without compromising security posture.

a. Red Hat OpenShift not impacted due to usage/configuration

False positives occur when a CVE appears in Red Hat Advanced Cluster Security reports, but does not represent an actual exploitable risk within the Red Hat OpenShift environment. In some cases, the affected component may be present, but is either unused, sandboxed, or configured in a way that prevents exploitation. In other cases, the component might not even be shipped by Red Hat, and the CVE appears purely due to upstream visibility. VEX data helps clarify these scenarios through product status indicators such as `known_not_affected` or remediations status indicators such as `no_fix_planned`.

Examples include:

- ▶ **CVE-2022-1996.** The `go-restful` package is a transitive dependency but not used in Red Hat OpenShift runtime context. VEX marks Red Hat OpenShift as `no_fix_planned`.
- ▶ **CVE-2023-28840.** Code only present in Fedora; not shipped in Red Hat products. False positive due to non-applicability.
- ▶ **CVE-2024-1485.** Affected code is shipped but unused in Red Hat OpenShift; exploitability is low, and mitigation is possible by blocking devfile parsing from untrusted sources.
- ▶ **CVE-2024-41110.** Regression affects only non-Red Hat environments. VEX confirms Red Hat products do not ship the affected component.
- ▶ **CVE-2024-45337.** Multiple Red Hat OpenShift components marked differently in VEX, but the affected code is not used, rendering it non-exploitable.

b. Inherited package status: Red Hat OpenShift depends on Red Hat Enterprise Linux

In some cases, a vulnerability exists in a shared package inherited from Red Hat Enterprise Linux, but if Red Hat Enterprise Linux itself is unaffected, Red Hat OpenShift is equally not exposed to the risk.

For example:

CVE-2020-1712. A `systemd` vulnerability where Red Hat Enterprise Linux 9 is marked `known_not_affected` in VEX. Since Red Hat OpenShift inherits the same base, it is also not at risk despite Red Hat Advanced Cluster Security potentially flagging it as `known_affected` with a `no_fix_planned` status.

c. Fix exists in RHSA, but CVE still flagged

There are instances where a CVE has already been patched through an RHSA, yet the vulnerability continues to appear in Red Hat Advanced Cluster Security dashboards. This typically occurs because VEX data is not retrospectively updated after initial publication, which can cause Red Hat OpenShift components to remain flagged even after remediation.

d. Mark as False Positive via Ansible Automation Platform Playbooks

To manage these false positives effectively, organizations can use Ansible Automation Platform Playbooks to mark CVEs as false positives in Red Hat Advanced Cluster Security. These playbooks interact with the Red Hat Advanced Cluster Security Central API to suppress irrelevant CVEs, helping teams maintain focus on genuine risks. The playbooks use Ansible Vault to protect credentials and rely on token-based authentication for secure execution. Sample playbooks can be accessed [here](#).

Residual risks

a. CVEs without fix or workaround

Residual risks refer to vulnerabilities where Red Hat acknowledges the issue but has not yet provided a fix or documented workaround. These vulnerabilities typically appear in VEX files with a status of `known_affected` and are often still under investigation or represent issues that are too complex to address in the short term. In such cases, no patch version is available, and no remediation guidance has been published.

Organizations encountering residual risks should raise a support case with Red Hat for further guidance and clarification. In parallel, it is recommended to apply compensating controls to minimize potential exposure. This can include restricting access to the impacted components, implementing strict admission controls, and enhancing runtime monitoring to detect any signs of exploitation or abnormal behavior.

These proactive measures help reduce the likelihood of successful exploitation while awaiting formal fixes or mitigation guidance from Red Hat.

Conclusion

Managing platform CVEs in Kubernetes environments such as Red Hat OpenShift requires more than traditional vulnerability management approaches. The volume of CVEs, combined with the complexity of containerized infrastructure, makes manual triage inefficient and often ineffective.

This detail introduced a structured approach to CVE management using Red Hat Advanced Cluster Security, including the use of Scanner V4, Red Hat VEX files, and actionable classification into remediable risks, false positives, and residual risks. By applying this framework, organizations can shift from reactive patching to proactive, risk-based decision-making.

Operational tools such as Ansible Automation Platform playbooks, pivot tables, and VEX analysis help streamline CVE triage, reduce noise, and allow teams to focus on vulnerabilities that truly matter. Embedding these practices not only strengthens security posture but also supports compliance and operational efficiency.

With the right processes and tools, organizations can cut through CVE overload, prioritize effectively, and enhance the resilience of their Red Hat OpenShift environments.

Appendices

A. Clean vulnerability reports using Excel pivot table

To streamline CVE analysis from Red Hat Advanced Cluster Security reports, you can use Excel pivot tables for better filtering and visualization.

Start by selecting the columns of your vulnerability report data, then create a pivot table.

Apply filters by “Namespace” and “Severity” to focus only on CVEs categorized as critical and important. For namespaces, filter to include only relevant platform namespaces.

- ▶ Red Hat OpenShift:
 - ▶ Namespaces starting with `openshift-` or `kube-`.
- ▶ Layered products:
 - ▶ Namespaces starting with `rhacs-operator`.
 - ▶ Namespaces starting with `open-cluster-management`.
 - ▶ Namespaces named `stackrox`, `multicluster-engine`, `aap`, or `hive`.

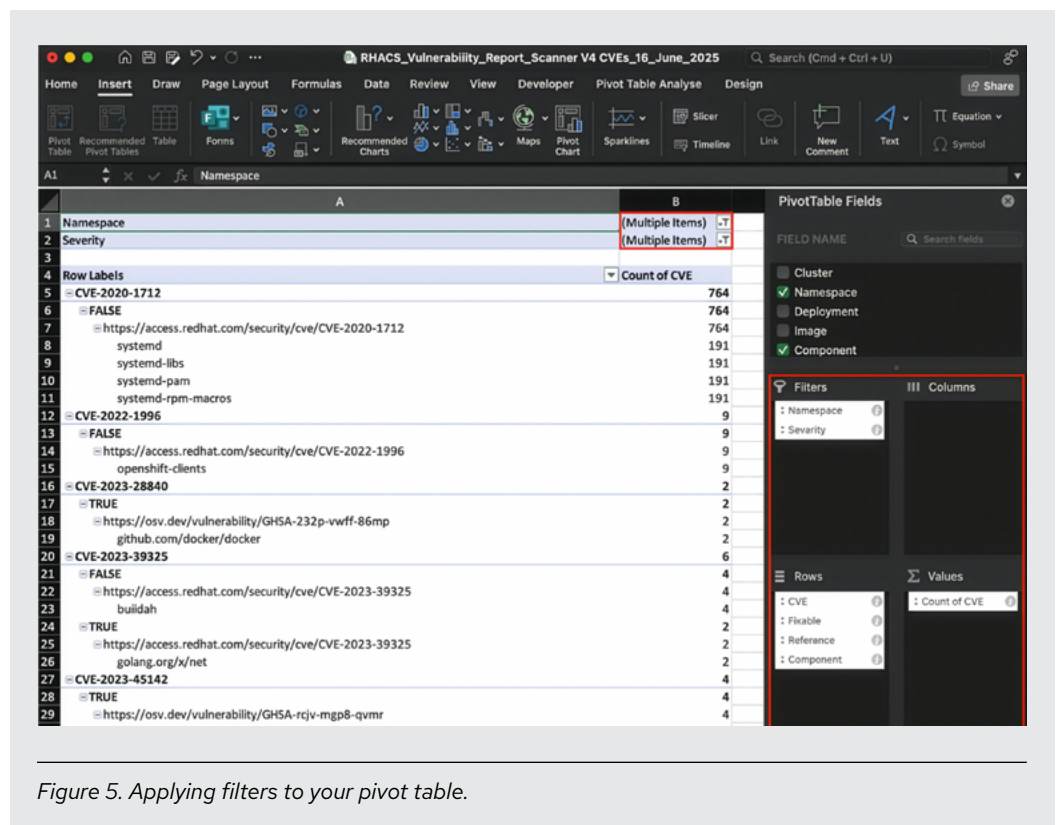


Figure 5. Applying filters to your pivot table.

For the pivot table rows, display fields such as CVE, fixable status, reference link, and affected component. Use value counts to break down the number of CVEs by each category. This method allows teams to quickly identify and focus on high-priority vulnerabilities affecting core platform components.

B. Clean vulnerability reports using the sample VBA macros

To automate the creation of pivot tables for CVE analysis, you can use the sample VBA macros with the following steps:

First, download the vulnerability report for the target cluster from Red Hat Advanced Cluster Security or from the scheduled email reports. Convert the vulnerability report from CSV format to XLSM format to enable macros. Copy the VBA macro code from the provided [link](#).

Second, open Excel and navigate to the developer tab. If the developer tab is not visible, right-click on the ribbon, select “customize the ribbon,” and tick the checkbox for “developer.”

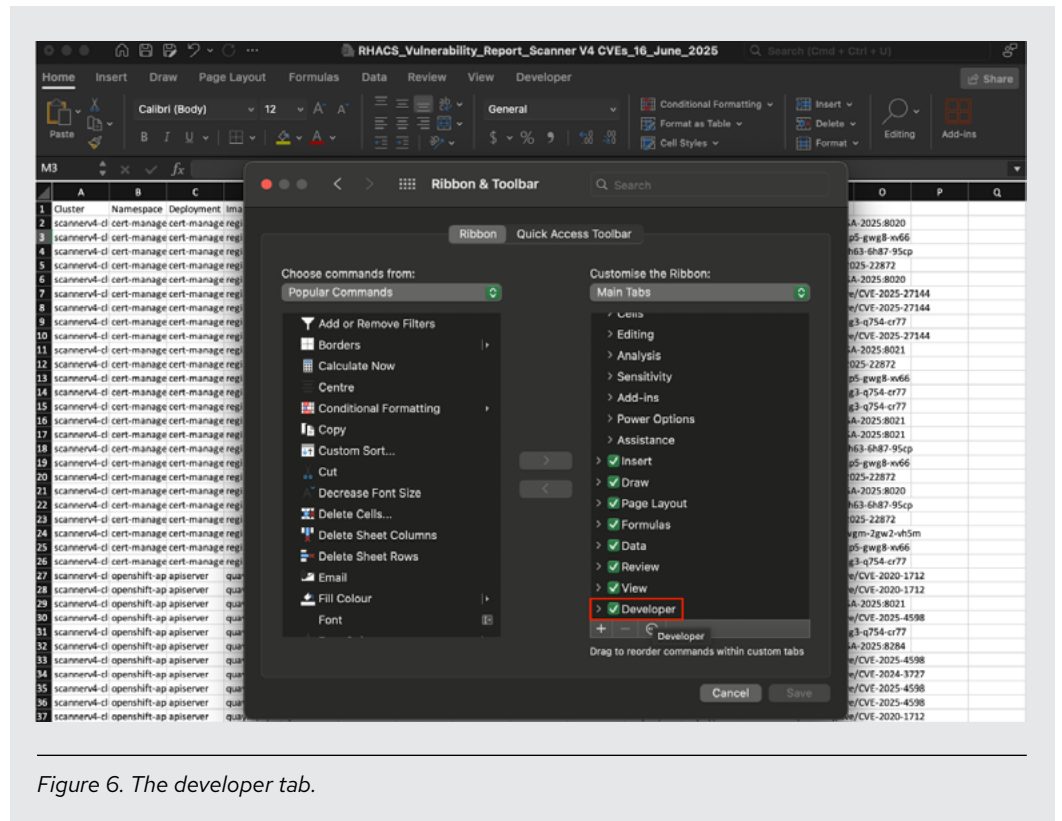
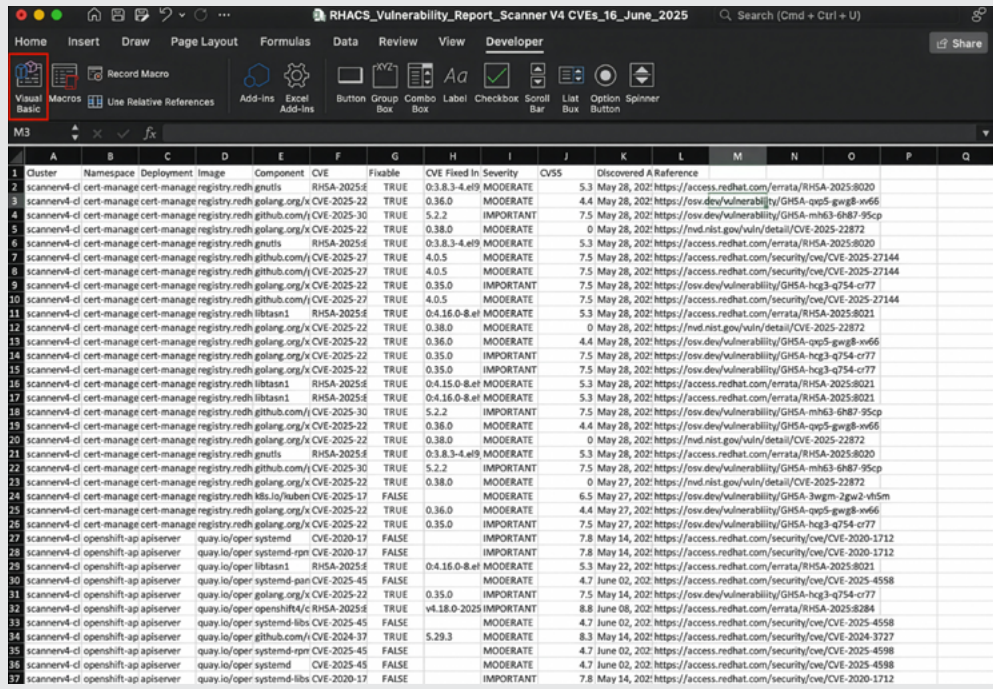


Figure 6. The developer tab.

Once the developer tab is visible, click on “Visual Basic” to open the VBA editor. Create a new module within your workbook and paste the copied VBA code.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Cluster	Namespace	Deployment	Image	Component	CVE	Fixable	CVE Fixed In	Severity	CVSS	Discovered At	Reference					
2	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	gnutls	RHSA-2025-4	TRUE	0.3.8.3-4.e19	MODERATE	5.3	May 28, 2022	https://access.redhat.com/errata/RHSA-2025-8020					
3	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	golang.org/x	CVE-2025-22	TRUE	0.36.0	MODERATE	4.4	May 28, 2022	https://osv.dev/vulnerability/GHSA-qw68-vw66					
4	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	github.com	CVE-2025-30	TRUE	5.2.2	IMPORTANT	7.5	May 28, 2022	https://osv.dev/vulnerability/GHSA-mh63-6h87-95cp					
5	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	golang.org/x	CVE-2025-22	TRUE	0.38.0	MODERATE	0	May 28, 2022	https://nvd.nist.gov/vuln/detail/CVE-2025-22872					
6	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	gnutls	RHSA-2025-4	TRUE	0.3.8.3-4.e19	MODERATE	5.3	May 28, 2022	https://access.redhat.com/errata/RHSA-2025-8020					
7	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	github.com	CVE-2025-27	TRUE	4.0.5	MODERATE	7.5	May 28, 2022	https://access.redhat.com/security/cve/CVE-2025-27144					
8	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	libtasn1	RHSA-2025-4	TRUE	0.4.15.0-8.e1	MODERATE	5.3	May 28, 2022	https://access.redhat.com/errata/RHSA-2025-8021					
9	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	golang.org/x	CVE-2025-22	TRUE	0.35.0	IMPORTANT	7.5	May 28, 2022	https://osv.dev/vulnerability/GHSA-hq3-q754-cr77					
10	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	github.com	CVE-2025-27	TRUE	4.0.5	MODERATE	7.5	May 28, 2022	https://access.redhat.com/security/cve/CVE-2025-27144					
11	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	libtasn1	RHSA-2025-4	TRUE	0.4.16.0-8.e1	MODERATE	5.3	May 28, 2022	https://access.redhat.com/errata/RHSA-2025-8021					
12	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	golang.org/x	CVE-2025-22	TRUE	0.38.0	MODERATE	0	May 28, 2022	https://nvd.nist.gov/vuln/detail/CVE-2025-22872					
13	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	golang.org/x	CVE-2025-22	TRUE	0.36.0	MODERATE	4.4	May 28, 2022	https://osv.dev/vulnerability/GHSA-qw68-vw66					
14	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	golang.org/x	CVE-2025-22	TRUE	0.35.0	IMPORTANT	7.5	May 28, 2022	https://osv.dev/vulnerability/GHSA-hq3-q754-cr77					
15	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	github.com	CVE-2025-27	TRUE	0.35.0	IMPORTANT	7.5	May 28, 2022	https://osv.dev/vulnerability/GHSA-hq3-q754-cr77					
16	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	libtasn1	RHSA-2025-4	TRUE	0.4.15.0-8.e1	MODERATE	5.3	May 28, 2022	https://access.redhat.com/errata/RHSA-2025-8021					
17	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	libtasn1	RHSA-2025-4	TRUE	0.4.16.0-8.e1	MODERATE	5.3	May 28, 2022	https://access.redhat.com/errata/RHSA-2025-8021					
18	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	github.com	CVE-2025-30	TRUE	5.2.2	IMPORTANT	7.5	May 28, 2022	https://osv.dev/vulnerability/GHSA-mh63-6h87-95cp					
19	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	golang.org/x	CVE-2025-22	TRUE	0.36.0	MODERATE	4.4	May 28, 2022	https://osv.dev/vulnerability/GHSA-qw68-vw66					
20	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	golang.org/x	CVE-2025-22	TRUE	0.38.0	MODERATE	0	May 28, 2022	https://nvd.nist.gov/vuln/detail/CVE-2025-22872					
21	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	gnutls	RHSA-2025-4	TRUE	0.3.8.3-4.e19	MODERATE	5.3	May 28, 2022	https://access.redhat.com/errata/RHSA-2025-8020					
22	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	github.com	CVE-2025-30	TRUE	5.2.2	IMPORTANT	7.5	May 28, 2022	https://osv.dev/vulnerability/GHSA-mh63-6h87-95cp					
23	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	golang.org/x	CVE-2025-22	TRUE	0.38.0	MODERATE	0	May 27, 2022	https://nvd.nist.gov/vuln/detail/CVE-2025-22872					
24	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	libtasn1	RHSA-2025-4	TRUE	0.4.15.0-8.e1	MODERATE	6.5	May 27, 2022	https://osv.dev/vulnerability/GHSA-3wqg-7q2w-vh5m					
25	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	golang.org/x	CVE-2025-22	TRUE	0.36.0	MODERATE	4.4	May 27, 2022	https://osv.dev/vulnerability/GHSA-qw68-vw66					
26	scannerv4-cl	cert-mgmt	cert-mgmt	registry.redh	golang.org/x	CVE-2025-22	TRUE	0.35.0	IMPORTANT	7.5	May 27, 2022	https://osv.dev/vulnerability/GHSA-hq3-q754-cr77					
27	scannerv4-cl	openshift-ap	apiserver	quay.io/oper	systemd	CVE-2020-17	FALSE		IMPORTANT	7.8	May 14, 2022	https://access.redhat.com/security/cve/CVE-2020-1712					
28	scannerv4-cl	openshift-ap	apiserver	quay.io/oper	systemd-rpm	CVE-2020-17	FALSE		IMPORTANT	7.8	May 14, 2022	https://access.redhat.com/security/cve/CVE-2020-1712					
29	scannerv4-cl	openshift-ap	apiserver	quay.io/oper	libtasn1	RHSA-2025-4	TRUE	0.4.16.0-8.e1	MODERATE	5.3	May 22, 2022	https://access.redhat.com/errata/RHSA-2025-8021					
30	scannerv4-cl	openshift-ap	apiserver	quay.io/oper	systemd-pan	CVE-2025-45	FALSE		MODERATE	4.7	June 02, 2022	https://access.redhat.com/security/cve/CVE-2025-4598					
31	scannerv4-cl	openshift-ap	apiserver	quay.io/oper	golang.org/x	CVE-2025-22	TRUE	0.35.0	IMPORTANT	7.5	May 14, 2022	https://osv.dev/vulnerability/GHSA-hq3-q754-cr77					
32	scannerv4-cl	openshift-ap	apiserver	quay.io/oper	openhm/rh	RHSA-2025-4	TRUE	v4.18.0-2025	IMPORTANT	8.8	June 08, 2022	https://access.redhat.com/errata/RHSA-2025-8284					
33	scannerv4-cl	openshift-ap	apiserver	quay.io/oper	systemd-lib	CVE-2025-45	FALSE		MODERATE	4.7	June 02, 2022	https://access.redhat.com/security/cve/CVE-2025-4598					
34	scannerv4-cl	openshift-ap	apiserver	quay.io/oper	github.com	CVE-2024-37	TRUE	5.29.3	MODERATE	8.3	May 14, 2022	https://access.redhat.com/security/cve/CVE-2024-3727					
35	scannerv4-cl	openshift-ap	apiserver	quay.io/oper	systemd-rpm	CVE-2025-45	FALSE		MODERATE	4.7	June 02, 2022	https://access.redhat.com/security/cve/CVE-2025-4598					
36	scannerv4-cl	openshift-ap	apiserver	quay.io/oper	systemd	CVE-2025-45	FALSE		MODERATE	4.7	June 02, 2022	https://access.redhat.com/security/cve/CVE-2025-4598					
37	scannerv4-cl	openshift-ap	apiserver	quay.io/oper	systemd-lib	CVE-2020-17	FALSE		IMPORTANT	7.8	May 14, 2022	https://access.redhat.com/security/cve/CVE-2020-1712					

Figure 7. Opening the VBA editor.

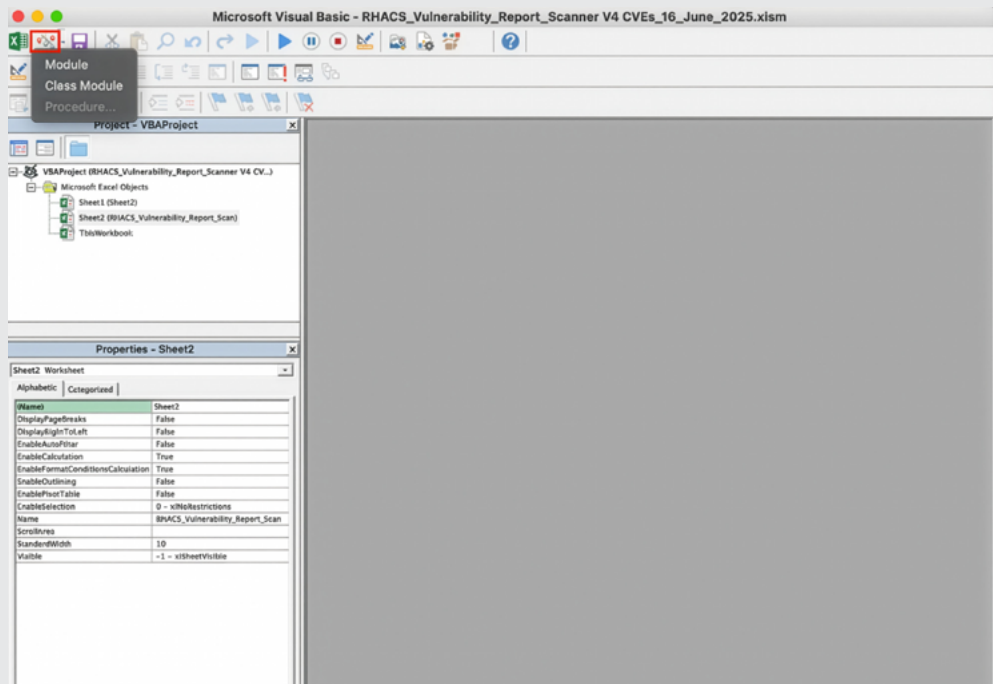


Figure 8. The view in the VBA editor.

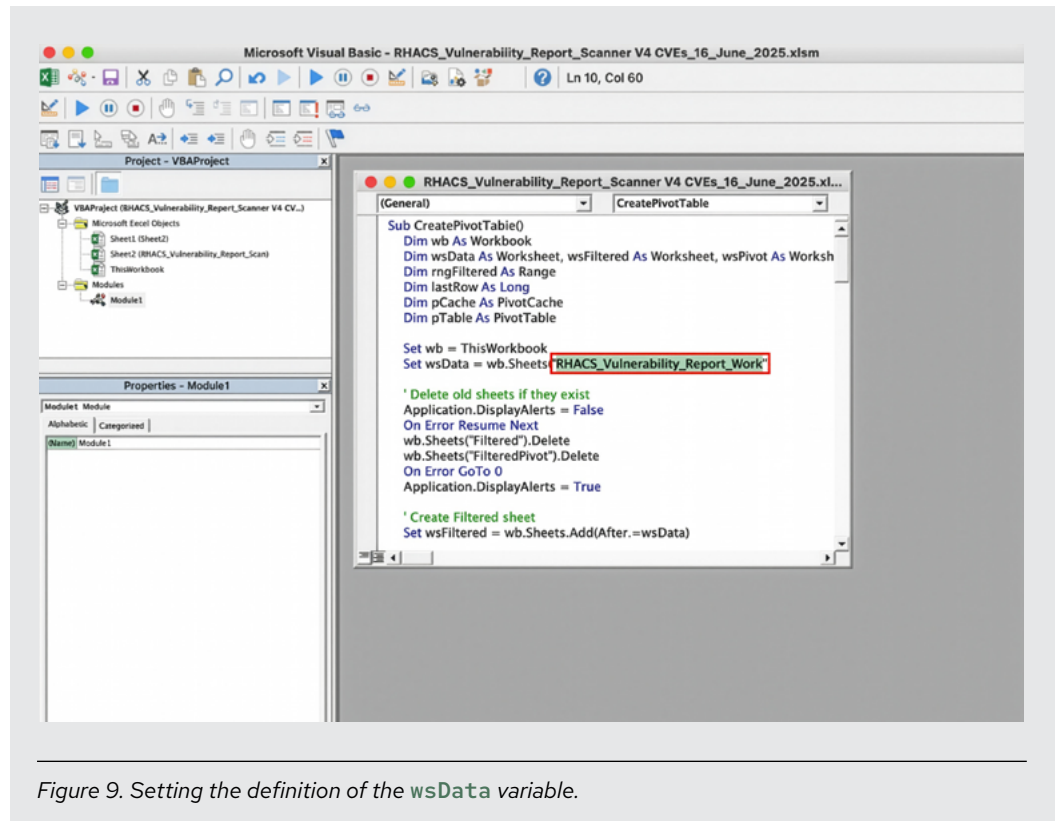


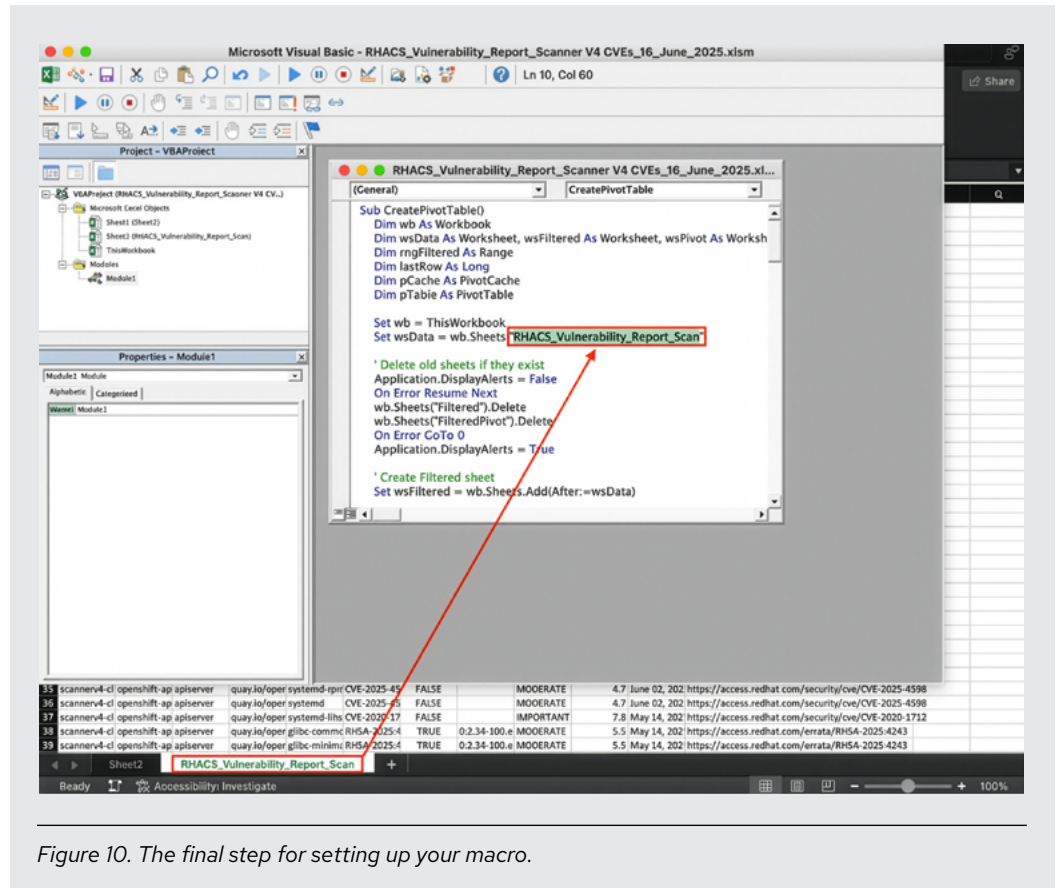
Figure 9. Setting the definition of the `wsData` variable.

Locate the section of the code that defines the `wsData` variable and update it to match the name of your worksheet. For example:

```
wsData = wb.Sheets("RHACS_Vulnerability_Report_Work")
```

The default name for your Excel sheet in the vulnerability report is usually something like `RHACS_Vulnerability_Report_<first 4 letters of the report name>`. Make sure this matches the actual name in your Excel file.

Save the file.



Once the macro is set up, return to your Excel sheet, click on "macros," select the macro you just created, and run it. The macro will generate the pivot table automatically, allowing you to filter and visualize the CVE data efficiently.

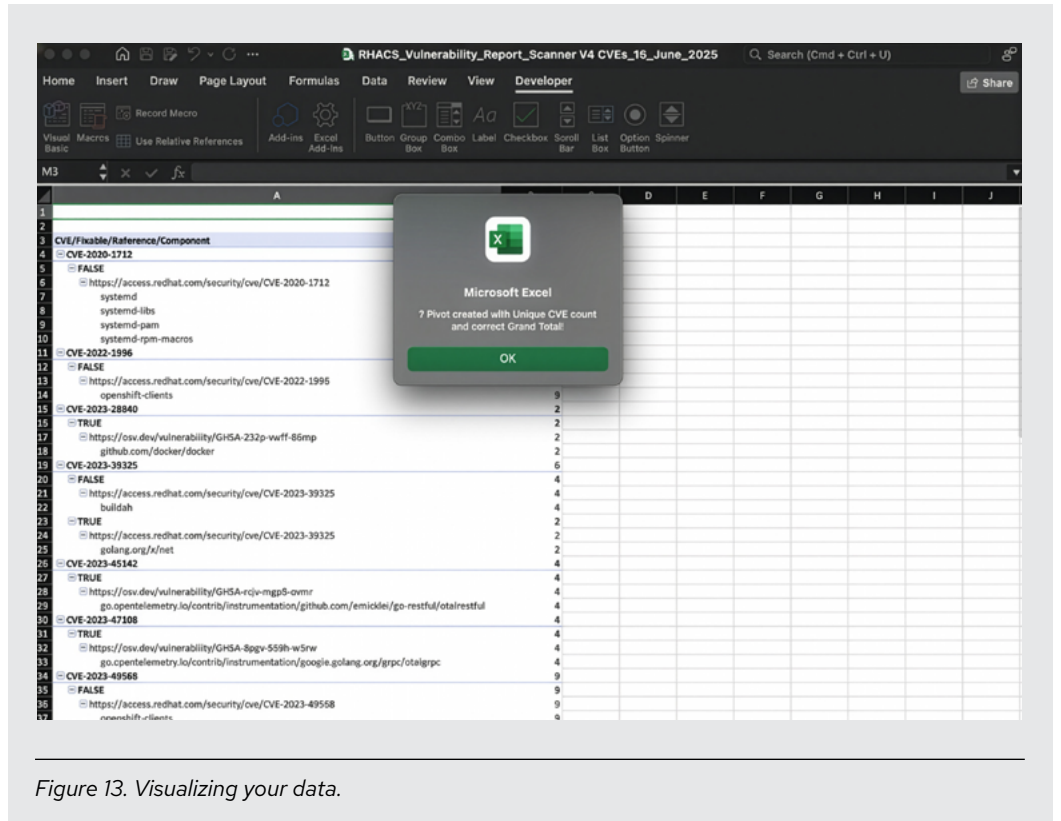


Figure 13. Visualizing your data.

C. Examples of reading VEX files

To read and interpret Red Hat VEX files for CVE triage, begin by logging into the [Red Hat CVE database](#). Search for the specific CVE you wish to review and download the corresponding VEX file in JSON format.

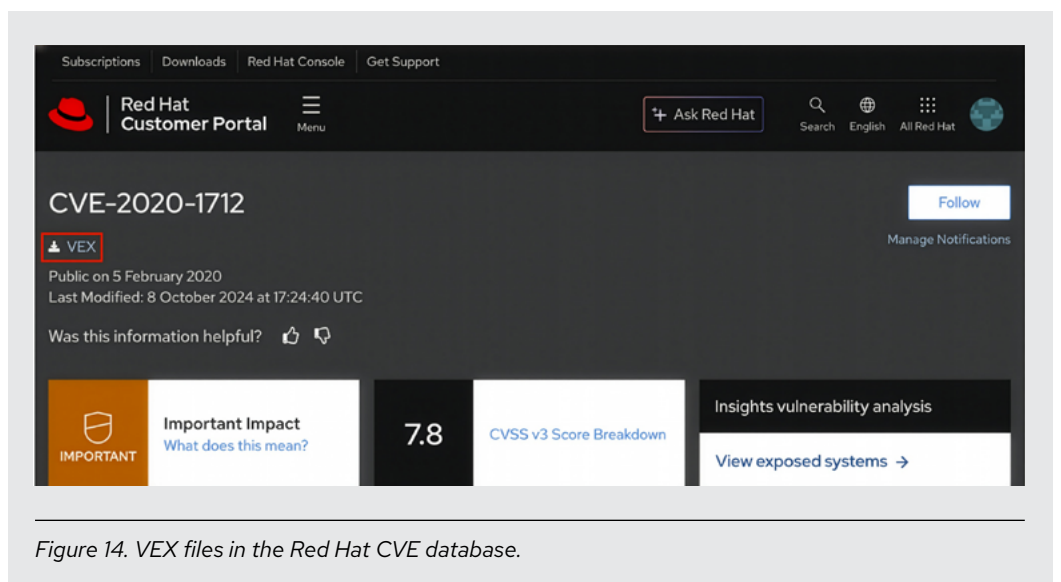


Figure 14. VEX files in the Red Hat CVE database.

Red Hat VEX files can also be found in a [public repository](#).

The following step-by-step example illustrates how to use VEX files for assessing different CVEs:

- 1. Identify the affected component.** Review the cleaned vulnerability report to identify the flagged package or component. For instance:
 - a. CVE-2020-1712.** Detected on the systemd component in Red Hat OpenShift Container Platform.
 - b. CVE-2023-49568.** Detected on openshift-clients in Red Hat OpenShift Container Platform.
 - c. CVE-2023-28840.** Detected on libnetwork/overlay driver.
- 2. Search in VEX.** Open the downloaded VEX JSON file and search for the component name. Review the associated Red Hat products in the `product_tree` section:
 - a. CVE-2020-1712.** systemd as a component of Red Hat Enterprise Linux 9 and Red Hat OpenShift Container Platform 4
 - b. CVE-2023-49568.** openshift-clients as a component of Red Hat OpenShift Container Platform 4.
- 3. Check product status.** Look at their statuses in the `product_status` object and check for any justification in the `flags` section:
 - a. CVE-2020-1712.**
 - i. `red_hat_enterprise_linux_9:systemd`
 1. `product_status > known_not_affected`
 2. `flags > vulnerable_code_not_present`
 - ii. `red_hat_openshift_container_platform_4:systemd`
 1. `product_status > known_affected`
 - b. CVE-2023-49568.**
 - i. `red_hat_openshift_container_platform_4:openshift-clients`
 1. `product_status > known_affected`
- 4. Review Red Hat's position and remediation.** Review the `statement` section for Red Hat's position and the `remediations` section for advisory references for additional context:
 - a. CVE-2020-1712.**
 - i. Statement. Red Hat confirms systemd in Red Hat Enterprise Linux 9 has no vulnerable code.
 - ii. Remediation. There is no fix planned for systemd in Red Hat OpenShift. As Red Hat Enterprise Linux 9 is not affected, Red Hat OpenShift inherits this safe state.
 - b. CVE-2023-49568.**
 - i. Statement. The issue affects only the Go Git implementation, not the native Git CLI.
 - ii. Remediation. No fixes exist but a workaround is provided, which is to restrict to using only trusted Git sources.

c. CVE-2023-28840.

- i. Statement. The affected component is confirmed as not shipped by Red Hat and only impacts Fedora community products.

5. Apply classification. Based on the above data points, classify the CVE in Red Hat Advanced Cluster Security using 1 of the following categories:

- a. False positive.** The CVE does not impact the Red Hat OpenShift environment based on Red Hat's assessment, such as in the case of CVE-2020-1712 and CVE-2023-28840.
- b. Remediable risk.** The CVE has a documented workaround or mitigation, though no immediate fix is available, such as CVE-2023-49568.
- c. Residual risk.** The CVE remains without a fix, workaround, or clear mitigation guidance and may require compensating controls while awaiting further updates from Red Hat.

6. Customer action. Take appropriate action within Red Hat Advanced Cluster Security:

- a.** For false positives such as CVE-2020-1712, CVE-2023-28840, and similar cases, mark the CVE as a false positive. No further action required.
- b.** For remediable risks such as CVE-2023-49568, apply the recommended workaround and defer the CVE with justification.
- c.** For residual risks, monitor the situation, apply compensating controls as needed, and raise a Red Hat support case if necessary.

Following this structured approach helps teams enforce consistent, defensible CVE classification and response based on authoritative Red Hat guidance.



About Red Hat

Red Hat is the world's leading provider of enterprise open source software solutions, using a community-powered approach to deliver reliable and high-performing Linux, hybrid cloud, container, and Kubernetes technologies. Red Hat helps customers develop cloud-native applications, integrate existing and new IT applications, and automate and manage complex environments. [A trusted adviser to the Fortune 500](#), Red Hat provides [award-winning](#) support, training, and consulting services that bring the benefits of open innovation to any industry. Red Hat is a connective hub in a global network of enterprises, partners, and communities, helping organizations grow, transform, and prepare for the digital future.

North America

1 888 REDHAT1
www.redhat.com

**Europe, Middle East,
and Africa**

00800 7334 2835
europe@redhat.com

Asia Pacific

+65 6490 4200
apac@redhat.com

Latin America

+54 11 4329 7300
info-latam@redhat.com

f facebook.com/redhat
X x.com/RedHat
in linkedin.com/company/red-hat

redhat.com