An Open Approach to Vulnerability Management

Red Hat's Methodology

Version 1.5

March 28, 2024

Red Hat

Contents

Introduction

Red Hat has published many articles, blogs, and other resources describing different facets of handling security vulnerabilities in our products. This document builds on those efforts, bringing it all together and helping our customers and communities better understand how Red Hat categorizes, addresses, and responds to security vulnerabilities.

Guided by the ethos of open source, this document is a transparent snapshot of our vulnerability management process provided for informational purposes only. It is a living and constantly evolving example of our risk based approach to security.

Red Hat welcomes feedback and comments from its customers, partners, and open source communities. As custodians of this document, please direct any feedback or comments to Red Hat Product Security.

Defining a vulnerability

Software flaws can happen regardless of the development model, be it open source or otherwise, and despite meticulous reviews. A defect or bug with security implications is referred to as a vulnerability. A vulnerability in software is a weakness or absence of a safeguard resulting in an unplanned adverse outcome.

Identifying and analyzing these vulnerabilities is vital to protecting the Red Hat portfolio used by our customers. Red Hat Product Security and its Product Security Incident Response Team (PSIRT) have been serving Red Hat, our subscribers, communities, and partners since September 2001. Red Hat Product Security oversees over 400,000 components and versions that are included within currently supported products and cloud services. Detailed information about coverage and support for products within the Red Hat portfolio can be found on the Product Life Cycles page.

References

- The Source of Vulnerabilities, How Red Hat finds out about vulnerabilities

How Red Hat reports and evaluates vulnerabilities

Vulnerabilities[1] are identified using an industry standard called the Common Vulnerabilities and Exposures (CVE). Every security defect impacting a component within the Red Hat portfolio has an assigned CVE identifier. Red Hat is a CVE Numbering Authority (CNA) for all Red Hat-branded software and also supplies CNA services for many open source projects. As of September 2022, Red Hat has also become a Root CNA for open source projects.

Red Hat Product Security is a member of the Forum of Incident Response and Security Teams (FIRST) and participates in the FIRST CVSS SIG. Red Hat uses the Common Vulnerability Scoring System (CVSS) industry standard as an additional measurement of each vulnerability we address. All CVEs impacting Red Hat products are issued a CVSS base score.

[1] In this document, we use *vulnerabilities* and *CVEs* interchangeably. We report on all security issues that are applicable to Red Hat software.

Every fixed and unfixed CVE affecting Red Hat's software portfolio has a public entry in the Red Hat CVE database on the Red Hat Customer Portal, which includes a severity classification, score, description, available mitigations, and an explanation of our score. We identify and classify the type of vulnerability using an industry standard called the Common Weakness Enumeration (CWE). Red Hat Product Security also collects and analyzes more detailed technical information on publicly accessible bugs in Bugzilla or Jira.

Every vulnerability reported to Red Hat Product Security is reviewed and analyzed by our team of open source software security specialists, with input and consultation from product engineering teams. These engineers understand how our products are composed, curated, hardened, packaged, delivered, and used by our customers. Their breadth of knowledge and experience in security-focused supply chain practices provide critical insights into the potential impacts of these vulnerabilities on our products and services. Additionally, Red Hat associates may find and report vulnerabilities in open source software to Red Hat Product Security, who then coordinate with other vendors, as appropriate.

Common Vulnerabilities and Exposures (CVE)

The goal of the CVE program is to establish a common vulnerability identification for all hardware and software vendors affected by a given vulnerability. It is common to see one CVE impact multiple vendors since they potentially source their components from the same upstream supplier. A CVE has the following format:

CVE-XXXX-YYYY

XXXX is the year the CVE was issued, and the *YYYY* is the unique number issued by the relevant CVE Numbering Authority (CNA). Software suppliers reference the CVE identifier in a security advisory or bulletin that notifies end-consumers of the vulnerability existing within a particular product or service requiring end-consumer action. Vulnerability aggregators and third-party security scanners leverage these CVE IDs as part of their processes.

References

- Red Hat CVE Database
- New and Improved CVE Pages
- MITRE's CVE site
- Red Hat extends CVE program expertise as newly-minted Root organization

Common Weakness Enumeration (CWE)

Common Weakness Enumeration (CWE) is a community-developed list of common software and hardware weakness types that have security ramifications. Weaknesses are flaws, faults, bugs, or other errors in software or hardware implementation, code, design, or architecture that if left unaddressed could negatively impact the information security triad: confidentiality, integrity, or availability (CIA) of an impacted component. Every vulnerability that impacts components provided by Red Hat will be evaluated using CWE for subsequent root cause analysis. CWE is described in the following format:

CWE-ZZZ

ZZZ is the unique identifier for the type of weakness as described in the CWE List.

References

- Red Hat is now CWE Compatible
- MITRE's CWE site
- Red Hat's CWE journey

Common Vulnerability Scoring System (CVSS)

CVSS conveys how a particular vulnerability works and what aspects of CIA are impacted by the flaw. CVSS is not a measurement of risk. Red Hat Product Security uses CVSS as part of our holistic approach to vulnerability assessment and its impact on our software portfolio. We also conduct other assessments, such as developing reproducers, analyzing the impact on layered products, and determining if the exploitability of the flaw is mitigated or reduced due to security features or our build and compiling practices.

The CVSS standard allows an individual to analyze a security defect through the use of a series of metrics that help describe certain aspects of the flaw. It is divided into Base Metrics, and two optional analyses: Temporal and Environmental.

CVSS:Q/AV:vv/AC:vx/PR:xx/UI:xy/S:yy/C:z/I:z/A:z

The CVSS v3 Base Metric group covers the following constant aspects of a vulnerability:

- Attack Vector (AV). This aspect expresses the proximity to the vulnerable system required for an attack and how the vulnerability is exploited.
- Attack Complexity (AC). This aspect speaks to the difficulty of executing an attack and what factors are needed for it to be successful.
- User Interaction (UI). This aspect determines whether the attack requires an active human to participate or if the attack can be automated.
- Privileges Required (PR). This aspect documents the level of user authentication required for the attack to be successful.
- Scope (S). This aspect determines whether an attacker can impact a component beyond its security scope or authority.
- Confidentiality (C). This aspect determines whether unauthorized parties can access information resources and to what extent.
- Integrity (I). This aspect measures the impact to the trustworthiness and veracity of data.
- Availability (A). This aspect measures the impact on authorized user access to data or services.

A formula translates these measurements into a single, numerical base score, ranging from 0.0 (no impact) to 10.0 (highest base impact). Refer to the Common Vulnerability Scoring System v3.1: User Guide for detailed descriptions of the Base Metrics. It is important to note that the CVSS Base Metrics were designed to be used with the other CVSS metric groups, notably the Temporal and Environmental Metrics, to provide an accurate representation of risk in customer environments. Alone, the Base Metrics offer a shallow view of the vulnerability itself without accounting for deployment or the environment.

As described in the Common Vulnerability Scoring System v3.1: Specification Document (emphasis added for clarity):

> *"The Common Vulnerability Scoring System (CVSS) captures the **principal technical characteristics** of software, hardware and firmware vulnerabilities. Its outputs include*

> *numerical scores indicating the* ***severity of a vulnerability relative to other vulnerabilities****."*
>
> *"Base Scores are usually* ***produced by the organization maintaining the vulnerable product****, or a third party scoring on their behalf. It is typical for only the Base Metrics to be published as these do not change over time and are common to all environments.* ***Consumers of CVSS should supplement the Base Score with Temporal and Environmental Scores specific to their use of the vulnerable product to produce a severity more accurate for their organizational environment.*** *Consumers may use CVSS information as input to an organizational vulnerability management process* ***that also considers factors that are not part of CVSS*** *in order to* ***rank the threats*** *to their technology infrastructure and make informed remediation decisions. Such factors may include: number of customers on a product line, monetary losses due to a breach, life or property threatened, or public sentiment on highly publicized vulnerabilities. These are outside the scope of CVSS."*

The use of CVSS, particularly the CVSS Base Score alone, in risk assessments has been misunderstood and misused in the industry for a long time. The software industry has adopted the poor practice of directly mapping a severity rating to CVSS and then prioritizing risk based solely on that rating, ignoring the Temporal and Environmental scores. The CVSS v3.1 specification was updated to address this problem. As described in the CVSS User Guide for changes in version 3.1 (emphasis added for clarity):

> *"The CVSS Specification Document has been updated to emphasize and clarify the fact that* ***CVSS is designed to measure the severity of a vulnerability and should not be used alone to assess risk****.*
>
> *Concerns have been raised that the CVSS Base Score is being used in situations where a comprehensive assessment of risk is more appropriate. The CVSS v3.1 Specification Document now clearly states that the CVSS Base Score represents only the intrinsic characteristics of a vulnerability which are constant over time and across user environments. The CVSS Base Score should be supplemented with a contextual analysis of the environment, and with attributes that may change over time by leveraging CVSS Temporal and Environmental Metrics. More appropriately,* ***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."*

Find further information on the Understanding Red Hat security ratings page.

It is important to note that Red Hat's assessments are limited to Base Metrics, which reflect the base characteristics of a vulnerability. This information becomes a starting point for organizations to determine the real impact of a particular vulnerability to them, taking into account their own implementation, application deployment, and risk environment.

Each organization is unique, with its own requirements and challenges, and all risks are not created equal, nor are they the same across companies. Consequently, over time, knowledge of the characteristics of a vulnerability may change, for example, an automatable exploit may be available six months after disclosure rather than the day of disclosure. Thus, it is important for organizations to augment the Base scoring by using the Environmental and Temporal Metrics to better reflect the characteristics and risks a vulnerability may represent.

While commonly considered optional, the Environmental and Temporal Metrics are critical to using CVSS as any kind of risk-related score. The Base Score alone must only be used to prioritize which vulnerabilities to focus on mitigating, when mitigations are available. It must not be used as a measurement of risk without using these additional metrics that require user input as to the environment and point-in-time potential exploit availability.

Along with a Red Hat CVSS score, Red Hat provides a Red Hat severity rating (discussed below) for all vulnerabilities impacting our products. This information is our primary guidance for our customers.

References:

- Understanding Red Hat security ratings
- How Red Hat uses CVSSv3 to Assist in Rating Flaws

Temporal and Environmental analysis

The Temporal and Environmental reviews are important yet frequently overlooked areas of the CVSS analysis. These are methods that end-users should be aware of and use in their own Risk and Vulnerability Management programs. Temporal review allows for the Base Metric score provided by a vendor, such as Red Hat, to be modified based on details around current exploitation techniques, the existence of attacks leveraging the vulnerability, or the availability of patches or workarounds for the defect.

The other critical measurement is the Environmental Metrics. This is where the practitioner can add organizational-specific details about mission-critical data, systems, or controls that might exist in the end-consumer's environment that could alter the impact or probability of an attack being successfully executed.

CVSS Base Metric scores are generic and based on default or most common configurations. They are *not* tailored to any one organization's configuration, sensitive data or systems, controls, regulatory or legal obligations, nor risk appetites. Consumers are always advised to conduct their own assessment of CVEs using all available data to inform their risk calculus.

CVSS scoring differences

Red Hat Product Security is the authoritative source for vulnerability data and scoring information for Red Hat products, services, and their components. A qualified engineer with direct technological experience reviews each CVE. Red Hat's scores are based on how our software is selected, compiled, built, and configured *at delivery*. The scoring reflects actual data and testing wherever possible. Each product component may be impacted differently by a specific vulnerability, so it is possible to see varying CVSS scores between different offerings and even between differing versions.

The Red Hat portfolio is largely based on upstream open source software. As part of our productization processes, changes are made to make that code easier to digest for enterprises. Therefore, *the software Red Hat provides is not necessarily identical to what could be obtained or used from upstream.* Vulnerability analysis on those upstream projects and components is often not always directly applicable to a Red Hat offering.[2] As a result, the CVSS scores of an upstream package often differ from those for a Red Hat product.

While popular, third-party vulnerability aggregators, such as the National Vulnerability Database (NVD), are *not authoritative* regarding how a given issue can impact a component or product. Red Hat is a member of the CVE Board and CVSS Working Groups. We make every effort to work with our industry partners, peers, and entities like NVD. Still, *a user should double-check any conclusion found in the NVD or by any other aggregator with the CVSS score and metrics determined by Red Hat* as shown in our CVE database for any particular issue.

[2] Conversely, vulnerability analysis on Red Hat products is not always applicable to upstream projects.

Depending on how issues are discovered or reported, sometimes NVD and other aggregators may report different scoring information. Red Hat Product Security takes this seriously and actively works with organizations like MITRE (which maintains NVD) to provide the appropriate technical details. We actively collaborate with them when there are unique differences with a Red Hat implementation of a package, library, or component.

References

- Security flaws and CVSS rescore process with NVD
- Security flaws mitigated by compiler optimizations

Red Hat Severity Ratings

Red Hat Product Security uses a four-point scale to describe a particular bug's severity based on rigorous analysis of the flaw. We designed this scale to align closely with similar scales used throughout the industry by other vendors and upstream open source communities. Our intent for the Red Hat Severity Rating is to help users determine which issues could pose more risk.

Ideally, this prioritized risk assessment helps customers understand how they may be exposed and enables them to better schedule updates to the systems they manage. We recognize that each business is unique, with its own requirements and challenges, and that all risks are not created equal, nor are they the same across companies.

The four-point scale rates vulnerabilities as Low, Moderate, Important, or Critical. Critical vulnerabilities pose the most severe threat to an organization. As described in our rating methodology, a Critical vulnerability could be exploited remotely over a network, the internet, or automated in an attack, such as by a worm. Like many of our peers, we expand this definition to include flaws that affect web browsers or browser plug-ins that users might be susceptible to from malicious or compromised websites.

CRITICAL	IMPORTANT	MODERATE	LOW
A remote unauthenticated user can execute arbitrary code Does not require user	Allows local users to gain privileges Unauthenticated remote users can view resources	Vulnerabilities are more difficult to exploit Are exploitable via an unlikely configuration	Unlikely circumstances required to exploit Impact is of minimal consequence

interaction i.e. Worms	Authenticated remote users can execute arbitrary code		

When Red Hat Product Security reviews a flaw, we look at how the software is sourced, built, packaged, and deployed. A CVSS Base score for Red Hat software assumes our products are used as designed, with security-focused defaults and settings in place. If subsequent changes are made to system settings or security controls, system administrators must account for that as they evaluate the risk a vulnerability might pose inside their unique environments. There are also scenarios where a Red Hat risk rating will differ from product to product for the same CVE. This demonstrates the flexibility in using our ratings to measure risk based on the product and not the base characteristics of the vulnerability.

It is important to remember that no vendor can tell a business what is important to them nor dictate actions to take to protect their sensitive data. CVSS and the Red Hat Severity Rating are baselines of our software, a starting point for consumers to begin their own risk assessment. It is also important to note that CVSS Base scores do not directly map to the Red Hat Severity Rating; CVSS is used as a guide to assist with understanding a vulnerability. The Red Hat Severity Rating is Red Hat's standardized rating that speaks to the risk posed by a vulnerability.

References

- Red Hat Severity Ratings
- What does the severity rating in the security advisory mean?

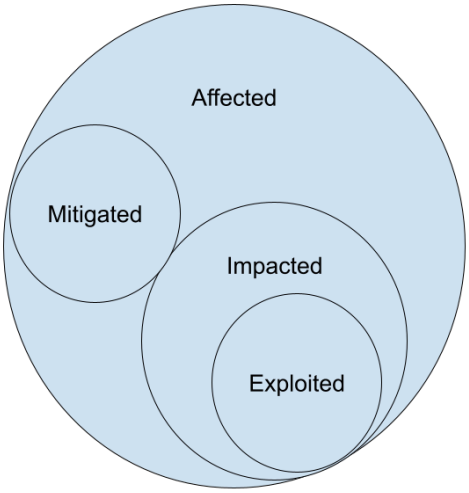
Security key terminology

Understanding the key terminology used to explain the product security statuses, risks, and impacts caused by the vulnerabilities is important. A good understanding of every term below avoids misunderstandings. Additionally, it allows one to better understand the main purpose of the vulnerability management process including the risk assessment step.

Affected. This term has broad scope and means any software containing vulnerable code shipped to the end user.
Impacted or Vulnerable. These terms mean any software that contains the vulnerable code and a real impact to the CIA.

Exploited or Exploitable. The vulnerability can be exploited.
Mitigated. The code is affected, but a mitigation exists, such as limited functionality or security controls.

The following diagram illustrates the relationship between flaw states:



Mitigating vulnerabilities

Knowing that there is a vulnerability is just the beginning. It is essential to have a software lifecycle management process in place and stay alert to key documentation, fixes, patches, mitigations, and workarounds. Red Hat helps customers stay up to date about advisories that have been issued, providing several simple paths to access signed and authorized updates.

Additionally, Red Hat's Customer Experience and Engagement (CEE) organization is available to help customers with questions about patched and unpatched vulnerabilities. If customers ever have questions about exposure, unfixed vulnerabilities, or other security-related questions, they are encouraged to reach out to their support representative.

References

- Red Hat CVE Database
- Red Hat Security Advisory Database

- Red Hat Product Security Center
- Red Hat Security Errata RSS
- OVAL Data feed
- CSAF Data feed
- Red Hat Security Vulnerability Data API

Life cycle considerations

Red Hat provides publicly available life cycle pages that describe how we support our software products during their lifetimes. There are three primary phases common to most offerings: Full Support, Maintenance Support, and the Extended Life phase. The level of support subscribers can expect varies depending on which phase of the life cycle a product is in. However, there are a set of common expectations. Note that vulnerabilities are prioritized and fixed based on Red Hat severity ratings.

During all phases of a product's support life cycle, Red Hat-rated Critical and Important Severity issues will be addressed, typically outside of a scheduled major or minor release. This may vary depending on the particular affected software and upon the complexity of the patch itself, taking into account whether the update is a backport or rebase, which is described in more detail in the following section. Refer to the Red Hat Risk Report for an understanding of how quickly vulnerabilities have been addressed.

Critical and Important fixes are applicable to all phases of a life cycle, be it Full Support, Maintenance Support, or the Extended Life phase. Low and Moderate fixes can become out of scope.

Vulnerabilities rated Moderate are treated on a case-by-case basis. Some Moderate vulnerabilities may be cause for concern and proactively addressed by Red Hat. Other Moderate vulnerabilities may be assessed as less concerning and not be fixed.[3]

For example, a vulnerability in Red Hat Enterprise Linux (RHEL) that has little bearing on its own but can be used in such a way that it has a greater impact on Red Hat OpenShift. In these instances, a proactive approach will likely be taken to mitigate the vulnerability on Red Hat OpenShift through the patching on RHEL. Typically, Moderate issues are corrected in an

[3] Low and Moderate fixes are not guaranteed to be fixed in any phase of a product's life cycle.

upcoming major or minor release and not asynchronously, as with Critical and Important fixes, at the discretion of Product Engineering teams.

Red Hat-rated Low severity issues have a minimal impact if successfully exploited, and the odds of exploitation are deemed small. Low severity issues may be addressed with other fixes for more severe issues at the next major or minor release, or at Product Engineering's discretion. Typically, Low severity security fixes are treated with the same urgency as bug fixes.

For example, a Low Severity flaw is only exposed to existing users with administrative privileges, meaning that *without* administrative privileges already in place, the flaw is not exploitable. This flaw can only be exploited through a compromise of an actual administrative user account by an attacker, or if the user *were* the attacker, commonly referred to as "insider threat". In a situation like this, the potential risk posed due to the presence of the flaw is itself insignificant. The potential exposure of an already-compromised administrative account is more significant in this case.

When vulnerabilities of this kind are reported, they are assigned CVE names for completeness and tracking. Because they are unlikely to have a significant impact or pose any real risk, fixes are not often produced. Implementing such a fix may actually *introduce* more risk, including incompatibilities with other software or functionality, instability of the running software, additional dependencies or libraries, and the possible introduction of bugs or new vulnerabilities.

References

- Red Hat Product Life Cycles page
- Red Hat Risk Report (2020)
- Red Hat Risk Report (2022)

Known exploited vulnerabilities

In addition to proactively addressing security vulnerabilities across the product life cycle, Red Hat Product Security also tracks reports of vulnerabilities known to be actively exploited. Data shows the majority of exploited vulnerabilities rated Critical and Important are typically addressed prior to being actively exploited. If a report is received that a vulnerability is actively being exploited and has not previously been fixed across the Red Hat portfolio, remediation is prioritized.

Vulnerabilities with a Critical or Important severity are the most cost effective targets for attackers, which is why addressing these vulnerabilities are prioritized. However, when exploitation of a flaw is observed in the wild, a fix will be prioritized regardless of severity rating.

In 2021, we began tracking exploit information, starting with the Cybersecurity and Infrastructure Security Agency (CISA) known exploited vulnerabilities catalog. Based on that list, only 26 out of the 1596 vulnerabilities reported were actively exploited (1.6%). Out of the 10 Critical vulnerabilities, only 1 was listed as actively exploited (10%), and of the 283 Important vulnerabilities in the same time period, only 7 were listed as actively exploited (2.5%)[4].

References

- Red Hat Product Security Risk Report (2021)
- CISA known exploited vulnerabilities catalog

Product composition

Backporting and rebasing

As a commercial open source software provider, Red Hat derives most of its software from upstream open source repositories and communities. To provide enterprise-ready stability, we use two techniques to provide software updates: backporting upstream code and updating to newer upstream versions, known as rebasing.

Backporting is performed when a particular software feature, enhancement, or fix is taken from a newer software version and applied to an older version of the same software. This is done to minimize additional code changes found in the newer upstream version. A common example is that Red Hat will backport features and enhancements from newer upstream kernel code into our stable Enterprise kernel.

There are two primary reasons for opting to backport distinct changes to current versions rather than use the new upstream version: the first is to ensure Application Programming Interface (API) and Application Binary Interface (ABI) compatibility with other software that depends on the component being updated. The second is to reduce the risk of introducing new potential bugs, compatibility issues, and new vulnerabilities that may be present in other features of

[4] Refer to the 2021 Red Hat Product Security Risk Report's "Known exploits of vulnerabilities in 2021" section for further details.

newer upstream versions that could adversely affect currently supported and stabilized versions. Backporting fixes have been proven to reduce the risk of new, unknown vulnerabilities.

As vulnerabilities are discovered in components that Red Hat provides and it is determined that the shipped software is not affected, that information is provided through our CVE pages and associated security metadata.

Every updated package increments the *release* number, which is a mechanism to determine whether or not a package is newer or older than a vulnerable release. This is often referred to as the N-V-R (Name-Version-Release) of a package.[5] Often in major releases, when significant new features are included or when backporting is not practical, Red Hat will "rebase" a package to a new upstream version. Rebasing is when the existing package is replaced with a newer version from upstream, thus, future releases and backports are "based" on this new version. When a package is rebased, the full version number, rather than just the Red Hat release number, is updated to reflect the new upstream version.

Containers

Containers are not shipped in the N-V-R format, they are identified by hashes. However, its internal content is versioned.

For each container that we ship, it has a minimal base content and libraries containing the surrogate that is required to provide functionality to the services. This base infrastructure is rebuilt with the latest and tested packages provided by Red Hat Enterprise Linux that contain all available CVE fixes to avoid the risks of a direct upstream rebase. The overall security container health is graded on the Container Health Index, explained in a latter section.

However, some container scanners don't know how to properly interpret the Red Hat security data feed, running checks in the scanner based on Name-Version, without taking into account the Red Hat release ID which documents the fixed CVEs and generates false positive alerts to the customers. This can be a major source of frustration for everyone involved.

References

- What is Red Hat's security patch and backport practice?

[5] For example, openssl-1.1.1d-4el8 would indicate the "openssl" package name, a "1.1.1d" version, and a "4el8" release, indicating the fourth release of this version for RHEL 8. A subsequent update to the package that does not change the version of the software would have a "5el8" release.

- Security Backporting Practice
- What is backporting and how does it affect Red Hat Enterprise Linux (RHEL)?
- Is your software fixed?
- Security flaws on unsupported products or products with limited support

Content signing

It is critical for consumers to be able to verify that the software they are using is authentic and untampered with. All RPM-based[6] and container image content is signed using the authorized Red Hat signing server. End-users should always check to verify the software they have downloaded and are about to install is genuine and authentic, only using known trusted sources for any software installed within their environment.

References

- Product Signing Keys
- How to sign rpms with GPG
- Securing RPM signing keys
- Verifying image signing for Red Hat Container Registry
- How to test verifying image signatures?
- OpenShift Container Platform v4.14: Container image signatures

Red Hat Security Advisories (RHSA)

Red Hat publishes several forms of advisories. Red Hat Security Advisories (RHSA) are published whenever an update to a product contains a security fix. Any RHSA can include fixes for multiple CVEs, and as such, always inherits the highest Red Hat Severity Rating of the CVEs being corrected.

Red Hat publishes advisories over numerous channels directly to subscribers and the larger open source community.

References

- Red Hat Security Advisories database

[6] Content delivered using the RPM Package Manager (RPM) format, such as for Red Hat Enterprise Linux and other products.

- Explaining Red Hat Errata (RHSA, RHBA, and RHEA)
- The RHSA notifications you want, right in your Inbox
- Anatomy of a Red Hat Security Advisory
- Red Hat Product Errata Advisory Checker (Customer Portal login required)

Red Hat management tools

Red Hat's portfolio includes several products and services that help simplify the management of your Red Hat assets. These tools are tightly integrated with the data provided by Red Hat Product Security and offer different paths to understanding where vulnerabilities might lie in a customer's portfolio and how it can address them.

References

- Red Hat Satellite
- Red Hat Ansible
- Red Hat Insights
- What is Clair?
- Red Hat closes acquisition of StackRox
- Creating a central patch management with ansible
- Managing the security of your Red Hat Enterprise Linux environment with Red Hat Insights
- Insights Security Hardening Rules

RPM package tools: yum/dnf

Many of Red Hat's offerings leverage RPM packages. This format and the associated utilities offer some unique capabilities regarding vulnerabilities. For example, customers using RHEL may choose to only install security updates and ignore any non-security updates that may be available.

References

- Is it possible to limit yum so that it lists or installs only security updates?
- How do I check if a specific kernel is vulnerable to a specific CVE?
- How do I check the changes of a proposed package update?
- Can I install/run packages from different versions of RHEL?
- How to use yum to download a package without installing it

Learning about vulnerabilities

Red Hat security data

All materials related to vulnerabilities impacting the Red Hat portfolio are publicly available after the vulnerability has been publicly disclosed. Our data is published on our award-winning Customer Portal, and in several industry-standard human- and machine-readable formats. Red Hat uses two well recognized methods to provide this data: OVAL and CSAF VEX.

References

- Security Data
- Understanding Red Hat products' vulnerabilities

Red Hat Security Data API

Red Hat has published information about vulnerabilities affecting our product portfolio since 1999. Since that time, delivery and formats have changed based on technological standards and consumer-demand. In 2016, we launched the Red Hat Security Data API that enables customers to interact with our security data through a modern API.

References

- Vulnerability API Blog
- Vulnerability API documentation

Old security data formats: Open Vulnerability and Assessment Language (OVAL) and Common Vulnerability Reporting Format (CVRF)

Over the years, Red Hat published most vulnerability data using the OVAL and CVRF data formats to provide security information about Red Hat offerings. However, the security data landscape is constantly changing and making adjustments and improvements to meet new industry standards and customer requirements is necessary.

Red Hat has been heavily involved in providing our customers and open source communities with access to security data since 2002, when we became a founding board member of Open Vulnerability and Assessment Language (OVAL). Red Hat announced OVAL compatibility in 2006. The OVAL data format is no longer sufficient to support all of the current requirements

for security scanning in containerized products with non-RPM content, or the representation of products and components version ranges.

Red Hat provided two feeds for consumers to obtain information in the OVAL format: v1 and v2. Since 1 April 2023, new content has not been published in the OVAL and DS v1 format. On 1 July 2023, all OVAL v1 and DS v1 data are compressed and moved to the following archive locations:
https://access.redhat.com/security/data/archive/oval_v1_20230706.tar.gz
https://access.redhat.com/security/data/archive/ds_v1_20230706.tar.gz

Full support of OVAL v2 content will continue until the end of 2024, after which customers and partners will be encouraged to use our CSAF and SBOM data files for their security data needs.

References

- What is OVAL and how can I use it to learn about security issues?
- Evolving OVAL
- Red Hat Security Advisories in CVRF

New security data format: Common Security Advisory Framework (CSAF VEX)

In June 2022, Red Hat started publishing security advisories in a new CSAF format as a beta version. In February 2023, we officially announced that the CSAF format is the official replacement to the old CVRF format for Red Hat security advisories. All released advisories are publicly available under the https://access.redhat.com/security/data/csaf/v2/advisories/ path.

CSAF files include the VEX profile to express which components of a specific product release have been patched to fix a particular CVE (fixed status) and which components are not affected by that CVE (known-not-affected status).

In October 2023, we published VEX files for every CVE that exists in the Red Hat CVE database to cover all security statuses defined in the CSAF VEX profile:

- **Fixed**: Information that the specific CVE is fixed in a particular product and components with a link to the released CSAF advisory
- **Known Affected**: Confirmation that the specific component and product is affected by a particular CVE and no fix is available

- **Known Not Affected**: Confirmation that the specific component and product are not affected by a particular CVE
- **Under Investigation**: Information that the Red Hat Product Security team is verifying the applicability and impact of a specific CVE to a particular product and component

By publishing data in the CSAF VEX format, Red Hat can provide transparent information about the applicability of a particular public CVE to all related products and their components in a machine-readable format.

Red Hat VEX files are currently considered as beta versions and are publicly available under the https://access.redhat.com/security/data/csaf/beta/vex/ path.

References

- The future of Red Hat security data
- CSAF VEX blog
- Vulnerability Exploitability eXchange (VEX) blog

Red Hat Container Health Index (CHI)

Containers are an architectural format that enable cloud computing and agile software development. Red Hat OpenShift is a Kubernetes-based application platform that employs container technologies, supported by the solid foundation of Red Hat Enterprise Linux and SELinux.

Containers are a different deployment and delivery architecture than traditional Linux-based platforms and require different methods and tools to monitor, assess, and address security vulnerabilities with the ability to manage updates to that infrastructure.

The Red Hat Container Catalog is a platform that provides container images. The Container Health Index (CHI) is a means to provide security information to consumers to understand how up-to-date the containers they desire to use are. Containers are also given a score based on the age and the criticality of the oldest flaw that is applicable to the container image. It is important to note that the CHI is based on unapplied, yet available, security fixes to the underlying products and does not account for vulnerabilities present for which there is no fix.

References

- Container Health Index grades as used inside the Red Hat Container Catalog

- Security Scoring and Grading for Container Images
- Resources on Red Hat Container Security
- Scanning pods for vulnerabilities - OpenShift Container Platform 4.5 Security
- The OpenShift Security Guide Book Download

Red Hat Incident Response Plan

Every year, thousands of potential threats need to be sorted through, prioritized, and corrected. Red Hat created the Incident Response Plan (IRP), containing the coordination process to help provide standardized, concise, clear advice, detailing the "what, where, who, why, and how" of Red Hat's response, irrespective of the severity of the security vulnerability. The IRP proactively prepares Red Hat via Product Security to effectively handle security incidents related to products and services produced by Red Hat.

References

- Understanding Red Hat's Product Security Incident Response Plan
- Security Bulletins

Red Hat Product Security Risk Report

Red Hat publishes an annual Product Security Risk Report detailing the threats, vulnerabilities, and fixes that impacted the Red Hat portfolio during the calendar year. These reports explore risk response statistics and insights from our work beyond vulnerability response, contextualizing them within the scope of events that occurred throughout the year. The information we provide in these reports is part of our continued dedication to transparency in our work, aiming to serve our customers and the community with valuable insight on the security issues we all face.

References

- 2022 Red Hat Product Security Risk Report
- The history of open source risk reporting
- Red Hat Risk Report: A tour of 2020's branded security flaws

Third-party security scanners

Understanding what vulnerabilities exist in any organization's environment is a critical yet daunting task. Companies rarely run one technology alone throughout their enterprise, so they

often rely on third-party security scanners that detect flaws across multiple technologies. If these tools use the appropriate vulnerability data, they can give a more accurate picture of actions that need to be taken by patch management staff.

But when they do not, the scanning tools can create quagmires of false positives and inaccuracies that distract administrative staff and slow down addressing meaningful issues.

Red Hat has a new, free certification program for third-party security scanners. Using these certified tools helps our customers get the most precise and accurate data possible from their vulnerability scanner. Red Hat welcomes additional participants in this certification program. The Red Hat Vulnerability Scanner Certification program was initiated in February 2021 to address inconsistencies in third-party scanners scanning Red Hat products.

Many scanning tool products in the market need more reliable sources of data. By contrast, certified scanners use authoritative data produced by Red Hat Product Security, which enables them to show users the Severity Ratings and CVSS scores produced by Red Hat as experts on Red Hat software who understand backported software correctly. Thus, a user of a certified scanner that properly employs this data can see relevant information in the right way; Showing few, if any, false positives and more importantly, reduced false negatives. By using Red Hat's recommended security data, the certified scanner can know what software has fixes available and the vulnerabilities for which no patch remediation currently exists.

References

- Red Hat Vulnerability Scanner Certification
- Introducing Red Hat Vulnerability Scanner Certification
- Third-party Security Ratings and Backporting
- Determining your risk
- "To be, or not to be," vulnerable... How customers and partners can understand and track Red Hat security vulnerabilities
- Tutorial on how to process vulnerability scans

Conclusion

This paper has described the frameworks, standards, techniques, and tools Red Hat uses around managing vulnerabilities discovered within components of our software portfolio. For further information, please visit the Red Hat Product Security Center.

If you have a question about a specific security vulnerability or believe that you may have knowledge of such a vulnerability, please contact us at one of the methods listed on our Security Contacts and Procedures page.

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 integrate new and existing IT applications, develop cloud-native applications, standardize on our industry-leading operating system, and automate, secure, and manage complex environments. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500. As a strategic partner to cloud providers, system integrators, application vendors, customers, and open source communities, Red Hat can help organizations prepare for the digital future.