Red Hat Product Security Risk Report: 2019

Introduction

The 2019 edition of the Red Hat® Product Security Risk Report is an overview of security vulnerabilities that impacted Red Hat products for the 2019 calendar year. We review large and small security vulnerabilities that were publicly announced throughout the year, as well as the data and metrics that were produced for these vulnerabilities across all of our solutions. We also review several high-impact, high-profile security vulnerabilities.

By "product," we mean a Red Hat offering listed at https://access.redhat.com/products. We reviewed vulnerabilities and the severity rating assigned to them by Red Hat, and then looked at which issues were of meaningful risk and which issues were exploited. This report is based on publicly available data and was prepared by a group of security specialists within Red Hat referred to as Red Hat Product Security. Red Hat Product Security assigns a Common Vulnerabilities and Exposures (CVE) name to every security issue we fix. If we fix a general bug that later turns out to have had a security implication, we go back and assign a CVE name to that issue. Every CVE fixed has an entry in our public database in the Red Hat Customer Portal, as well as a public bug report with more technical detail. In this report, we use "vulnerabilities" and "CVEs" interchangeably.

Every vulnerability reported to Red Hat Product Security is reviewed and analyzed by our team of open source security specialists. These specialists are experienced Red Hat engineers who understand how our offerings are composed, curated, hardened, packaged, delivered, and used by our customers. This breadth of experience and insight into Red Hat Product Engineering supply chain practices helps provide critical insights into the potential impact of these vulnerabilities on our products and services.

Red Hat Product Security has more than 18 years of focused security experience and expertise. We have worked through the conversion of physical servers to virtual, virtual to cloud, legacy applications being decomposed into containers, and beyond. Along that journey, we have forged deep bonds with the open source community, which has earned us wisdom and insights into the challenges open source faces when it comes to security.

How Red Hat works with vulnerability reports

Red Hat Product Security is a member of the Forum of Incident Response and Security Teams (FIRST) Common Vulnerability Scoring System (CVSS) Special Interest Group (SIG) and uses the industry standard CVSS as an additional measurement of each vulnerability we address. All CVEs impacting Red Hat products are issued a CVSSv3.x score. CVSS is useful to describe how an attack works. However, there are limitations in what it can describe, so Red Hat does not use CVSS to prioritize vulnerabilities.

Instead, Red Hat Product Security uses a four-point scale to describe the severity of a particular bug based on rigorous analysis of the flaw. This scale was designed to align closely with similar scales used throughout the industry by other vendors and upstream open source communities. The severity levels are intended to help users determine the degree of risk issues could pose to them. Ideally, this prioritized risk assessment helps customers understand how they are exposed and allows them to

Across all Red Hat products, and for all issue severities, we fixed 1,313 vulnerabilities by releasing more than 968 security advisories in 2019.

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 company to company.

Our four-point scale rates vulnerabilities as Low, Moderate, Important, or Critical. Critical vulnerabilities pose the most severe risk to an organization. As described in our rating methodology, a Critical vulnerability could be exploited remotely over a network or the internet and could be automated (by a worm, for example). We expand this definition, as do many of our peers, to include flaws that affect web browsers or browser plug-ins that users might be susceptible to if they visited malicious or compromised websites.

We have provided in-depth explanations about risk management over the last year to help explain why end users need to conduct their own assessments based on their own risk appetites, priorities, and compliance demands. Managing risk is highly variable from environment to environment, and while tools like CVSS are one element in that risk analysis, the score alone does not accurately depict risk. When Red Hat Product Security reviews a flaw, we do so in the context of how that software is sourced, built, packaged, and deployed. As we issue a true CVSS score for Red Hat software, we assume our products are used as designed, with security-focused defaults and settings in place. If changes are made to system settings or security controls outside of the baseline recommendations, system administrators should take that into account as they evaluate the risk a vulnerability might pose inside their unique environments.

2019 also brought a lot of procedural and technical changes for Red Hat Product Security that were driven by feedback from our customers and partners. Over the years, we have been involved in countless customer and partner conversations about vulnerabilities, support life cycles, backporting, and many other open source security-related topics.

Evolving insights and responses

A common theme we have seen in 2019 is customers grappling with ever-mounting compliance obligations and their use of third-party scanners to help them manage vulnerabilities across their enterprises. While scanning tools can provide a useful "single pane of glass" view of vulnerabilities across an enterprise-wide environment, they generally do a poor job of articulating risks specific to a technology or implementation. Typically, these scanners rely upon tools like the National Vulnerability Database (NVD) managed and maintained by NIST (the U.S. National Institute of Standards and Technology). NVD is an aggregation point, collecting data from numerous sources. Historically, NVD displayed only one score per CVE, so if multiple vendors used and deployed the same package, only one score was displayed. Generally speaking, this is fine – the vulnerability itself is the same irrespective of vendor, hence why the same CVE identifier is used regardless of vendor specifics. This is because the CVE is tied to the defective code, not the outcome of the code that may change depending on vendor-supplied configurations, compilation options, unique deployments, or use.

However, scanning tools cannot replace the support provided by Red Hat Engineering and Red Hat Product Security. Red Hat Engineering spends a great deal of time and effort making upstream packages enterprise-ready by regression testing, hardening, and tweaking the package to meet our customers' unique business demands and our release standards. Red Hat Product Security consults on package selection and curation as packages are selected for use in our products and services, and helps to understand what vulnerabilities or security challenges might exist prior to fully releasing a package as part of a Red Hat-branded offering.

To help highlight the value of a Red Hat subscription and to assist in illustrating the differences between Red Hat-branded offerings and other, less applicable perspectives, Red Hat Product Security redesigned our CVE pages in 2019 after listening to customer feedback. Our website now features a wealth of data, including a comparison of the Red Hat Product Security scoring and severity of a CVE alongside the NIST NVD score (when available). We have documented these changes in our new and improved CVE pages blog entry.

The feedback we received from our customers and partners did not stop at a desire for clearer, more detail-filled vulnerability pages. Understanding risk is more complicated than ever. Migrating from legacy in-house deployments to hybrid cloud installations opens new attack surfaces and additional complexities in deployment. When Red Hat Enterprise Linux® 8 was released in May 2019 at Red Hat Summit, the Engineering team also released a new tool called the Red Hat Universal Base Image (UBI). It is an immutable, authoritative Red Hat Enterprise Linux-based container designed specifically for use in cloud deployments across whatever cloud you may be running in. This new delivery method, combined with evolving scanning tools and increased customer compliance obligations, required Red Hat to evolve.

New technologies can lead to new complications. Many of our aforementioned scanning partners are still grappling with how precisely to review and risk-score containers. The Red Hat Universal Base Image, widely used outside of the customers' datacenters, exposed some process gaps (detailed below). This evolution of usage and location facilitated changes to older models of thinking. Red Hat has a policy of publicly sharing security data about our products, and we were an early adopter of the OVAL data sharing standard. We continue to actively participate as a board member in that industry working group. The "classic" use of OVAL data was to report the issues that were fixed via traceable security advisories (other avenues spoke to potential risks, such as our Security Vulnerability Data API). Typically, this allowed a customer to answer the question: "Which available vulnerability patches are applicable to a given Red Hat product?"

We have witnessed an increasing number of requests to "apply all known patches," deferring to simplicity over precision. To help our customers and partners better understand the full scope of vulnerabilities that may exist and to give them data that is critical to enable them to conduct assessments of their own risks within their business environments, we evolved our OVAL data feeds. Throughout the latter half of the year, Red Hat Product Security worked on retooling the data to provide a full feed of known issues, fixed or otherwise, in Red Hat Enterprise Linux. We plan to continue to expand this feed to include RPM-based layered products next, and eventually to account for all products in the future.

2019 also was the year of a fairly sizable change to the support life cycle of our flagship product, Red Hat Enterprise Linux. Because Red Hat Enterprise Linux is the foundation of all of our products and services, we felt it was important to expand the scope of what we supported. Based on previous trends (explored in more detail below), we have recognized that more vulnerabilities were discovered this year than any previous year. Our customers have ever-increasing compliance demands, so our Red Hat Enterprise Linux product team decided to expand what fixes are included as part of our Extended Update Support life cycle to include Important-rated issues (which typically cover the largest share of issues within the scope of regulatory frameworks like PCI-DSS and others). Previously, Red Hat was more selective about which Important-rated issues were addressed in the Extended Update Support streams of Red Hat Enterprise Linux.

If you look solely at Linux vendors, the same CVE can have different effects, depending on how the product is compiled or deployed.

Vulnerabilities

Across the Red Hat product portfolio, and for all issue severities, we fixed more than 1,300 vulnerabilities by releasing more than 960 security advisories in 2019. While more than the number of CVEs we addressed in 2018 (1,272), it is not a significant increase, especially compared to the number of CVEs in 2015 (1,336) and 2016 (1,342). Addressing CVEs provides a steady stream of work that needs to be addressed by the open source community and Red Hat, as well as customers. Ensuring that systems are up to date with the latest fixes is critical to the foundation of organizational security.

All software programs are different and contain different bugs, so using a vulnerability count is not a fair measure of comparing one product to another to deem one "more secure" than another. Assigning and reporting practices vary greatly throughout the industry (as we have cited previously) and from product to product. If you look solely at Linux vendors, the same CVE can have different effects, depending on how the product is compiled or deployed.

Even within a single product like Red Hat Enterprise Linux, there can be high variability. Red Hat Product Engineering crafts a default deployment configuration that system administrators have the flexibility to alter – enabling or disabling features, but "out of the box," the product has a suggested set of configurations. Additionally, not every package is installed, nor are some even likely to be installed, in an enterprise installation.

As software matures, traditional practices evolve or are abandoned, and features are added or removed. Thus, a comparison of vulnerabilities between versions of the Red Hat products yields interesting, but not drastically useful comparisons beyond trends or reflective efforts put into building those offerings. The offerings are a snapshot in time of development practices used within the open source community, and will evolve and advance over time. It is typical to see high volumes of CVEs and bug fixes addressed as new major versions of software are released.

Table 1 compares the vulnerability counts of a subset of our Red Hat Enterprise Linux product family and other products within our portfolio. A single Red Hat Security Advisory (RHSA) will often fix multiple vulnerabilities across multiple versions of a product. We view the vulnerability count as a general indication of the amount of effort a customer will spend to both understand and then patch or mitigate the issue within their environment.

Table 1. RHSA Comparison Chart - 2019

Product	Critical	Important	Moderate	Low
All	40 v	566 ^	303 ^	59 ^
Red Hat Enterprise Linux 6, 7, 8	39 v	411 ^	151 ^	45 ^
Red Hat Enterprise Linux 7 - default install	9	1108	72	28
Red Hat Enterprise Linux 8 - default install	8	84	42	14
Red Hat JBoss® Enterprise Application Server - all supported versions	0	13	15	2

Product	Critical	Important	Moderate	Low
Red Hat OpenShift® Container Platform - all supported versions	0 v	41 ^	48 ^	6 ^
Red Hat OpenStack® Platform - all supported versions	0 --	27 ^	25 ^	5 ^

Legend

v = trend down ^ = trend up -- = no trend change

To review these numbers closer, we see that we issued 40 Critical security advisories that addressed 27 Critical CVEs. It is interesting to note that 41% of these Critical security advisories were issued within 1 day of the issue becoming public. The average delivery time for Critical advisories was within 7 days of the issue becoming public, with a median of 3 days. A record-breaking 340 Important CVEs were addressed through 566 RHSAs. For these vulnerabilities, 18% had initial patches available within 1 business day, with the average delivery time of 69 days and the median 23 days.

The year at a glance:

- 2,714 security issues were reported to Red Hat Product Security (slightly down from 2018)
- 1,313 CVEs were addressed throughout 2019, a 3.2% increase from 2018
- 968 Red Hat Security Advisories were issued, a record increase over previous years
- 40 Critical advisories addressed 27 Critical vulnerabilities
- 41% of Critical issues were addressed within 1 business day
- 85% of Critical issues were addressed within 1 week

The underlying Red Hat Enterprise Linux infrastructure helps ensure that the layered products running on top of it have a stable foundation. The layered offerings, like Red Hat OpenShift Container Platform, Red Hat OpenStack Platform, and Red Hat single sign-on, have their own unique flaws that need to be tracked and addressed as well, but the magnitude and volume of changes in these solutions is far smaller than the base operating system (which provides 1,348 packages for the default Red Hat Enterprise Linux 8.1 – with GUI – installation).

This highlights Red Hat's unique position up and down the stack. Red Hat understands the nuances of systems installed using our portfolio of products and can provide advice and insight on what the risks are in that combined solution. Otherwise, system administrators and DevOps practitioners would not only need to think through the deployment of 95 updates for their OpenShift Container Platform infrastructure, but also the 646 updates released for Red Hat Enterprise Linux in 2019. With practices honed over 25 years of creating and supporting Red Hat Enterprise Linux, our newer offerings inherit all of these proven practices and the knowledge that helps Red Hat react quickly as flaws are discovered and reported.

Table 1 reflects numbers related to the default installations of those products. Red Hat products are delivered in a generally secured state with reasonable, security-focused defaults (which are intended to cover the maximum number of reasonable use cases) and services enabled. Customers wishing to

reduce their threat footprint should consider additional hardening beyond the defaults as detailed in documents like the Red Hat Enterprise Linux 8 security hardening guide. The steps and techniques there help further protect systems. Along with that guidance, customers can install or remove packages and processes they do not need to reduce the number of potential threats they might be exposed to throughout the normal course of operations. When default security features are disabled (like turning off Security-Enhanced Linux, for example) the risk profile of that system is drastically altered, increasing the potential for additional security risks and impacts.

Table 2. Product package comparison

Product	Number of packages
Red Hat Enterprise Linux 8.1 - default w/GUI	1,348 RPMs
Red Hat Enterprise Linux 8.1 - minimal	405 RPMs
Red Hat Enterprise Linux 8.1 - full	2,321 RPMs [525 (Base OS) + 1796 (AppStream)]
Red Hat Enterprise Linux 7.7 - default	343 RPMs
Red Hat Enterprise Linux 7.7 - full	2,319 RPMs
Red Hat Enterprise Linux 6.10 - default	672 RPMs
Red Hat Enterprise Linux 6.10 - full	1,925 RPMs
Red Hat OpenStack Platform 15	736 RPMs + underlying OS
Red Hat OpenShift Container Platform 4.2	200 components + underlying OS
Red Hat JBoss Enterprise Application Platform 7.2.4	530 jars + underlying OS
Red Hat Satellite 6.6	355 packages + 357 from RHEL+ underlying OS

Table 2 shows how many packages are installed for each product based off of some of the choices the administrator makes as the system is being deployed. It is always advisable to install as few packages and components as possible to establish a smaller attack surface. Fewer packages and components also means less maintenance as updates and security advisories are released throughout that system's life cycle.

Vulnerability trends

Typically when looking at a specific product, we find fewer vulnerabilities over time as issues get addressed through our security backporting practices (where fewer code changes, which may introduce new issues, are included). We use the term "backporting" to describe the action of taking a fix for a security flaw out of the most recent version of an upstream software package and applying that fix to an older version of the package we distribute. Our backporting efforts enable us to deploy automated updates to customers with less risk.

Manage your risks. Don't let your risks manage you.

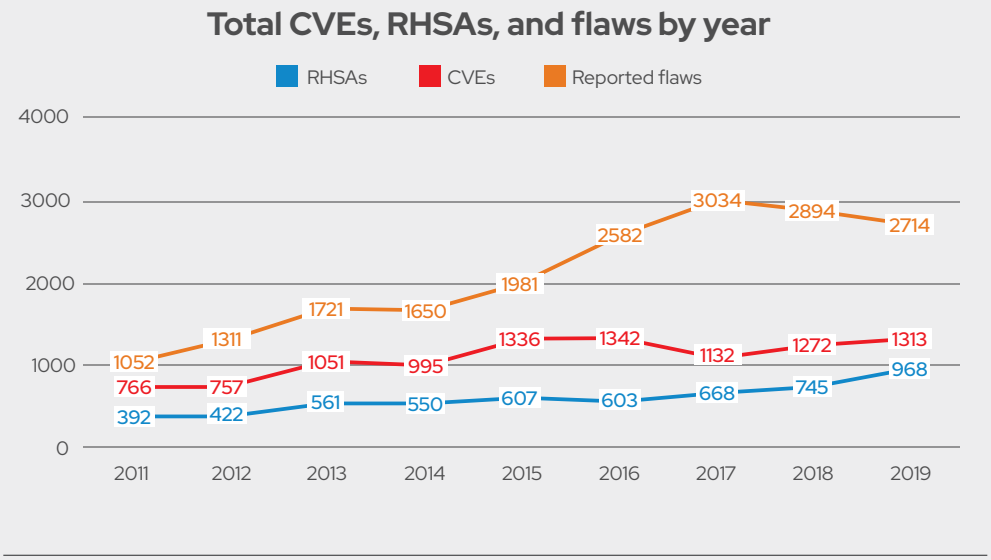

Figure 1. Total CVEs, RHSAs, and flaws

In 2019, there was a slight decline in the number of reported issues we reviewed, but there was an increase in the number of CVEs that were addressed and the number of security advisories that were published to address them. We also saw the largest number of Red Hat Security Advisories published to date.

Figure 1 describes all of the known and reported issues across our product portfolio and accounts for the CVEs that impacted our solutions. Particularly keen-eyed readers may recall a security scanner vendor of choice possibly showing different numbers from their tools and marketing materials. Third-party tools typically defer to the upstream as the basis for their analysis. Thus, they do not correctly account for the open source practice of backporting, nor account for compiler options and configurations applied during the software build process, which upstream code does not do.

For those readers interested in exploring data, Red Hat Product Security offers several options. Our security data is published publicly through multiple channels, including the Red Hat Customer Portal security metrics page and our Red Hat security data API. The data is intended to provide a clear and accurate picture of the vulnerabilities that may exist in a Red Hat solution. It is important to us that customers completely understand their risks.

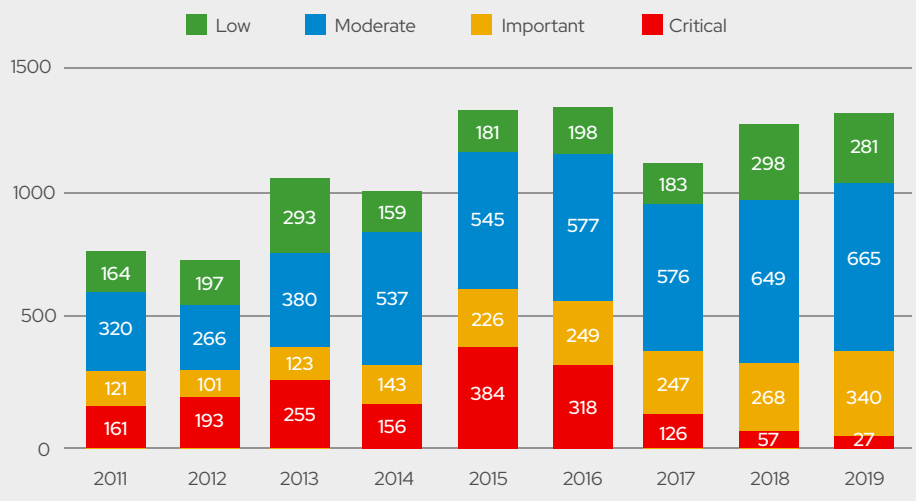

Figure 2. CVEs fixed by severity

Figure 3 shows that Red Hat Enterprise Linux provides the largest share of our vulnerability management work. With our partners in Red Hat Enterprise Linux Product Engineering, we fixed 1,066 CVEs in Red Hat Enterprise Linux in 2019.

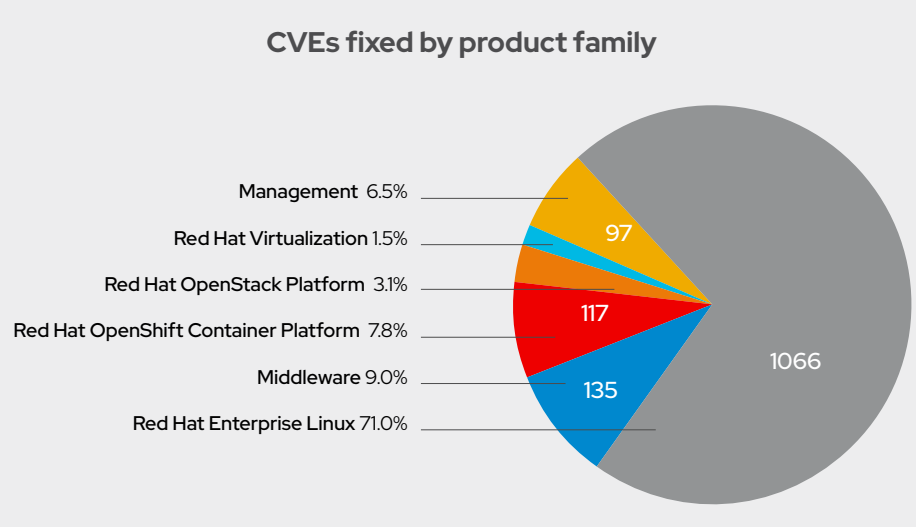

Figure 3. CVEs by product or product family

To illustrate this work over time, we can view the data like this:

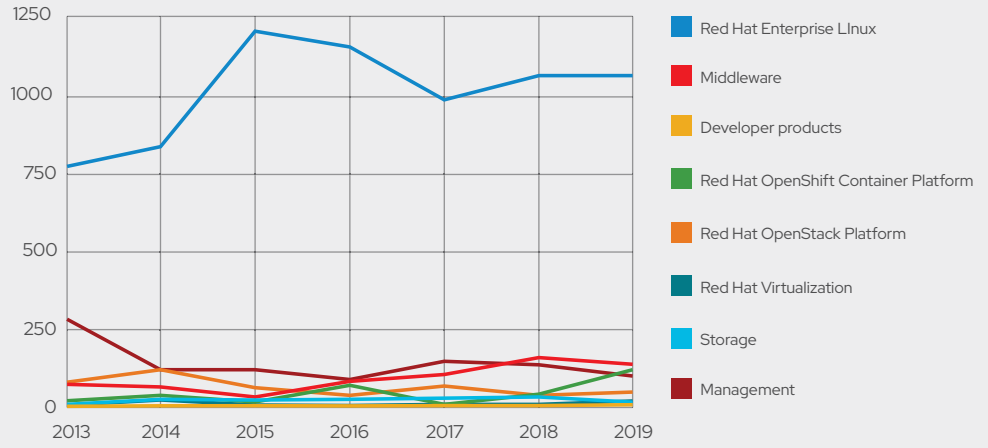

Figure 4. Red Hat portfolio view: CVEs fixed by year

The above information reviews the volume of work output in 2019. Now we will review the delivery times of those vital fixes.

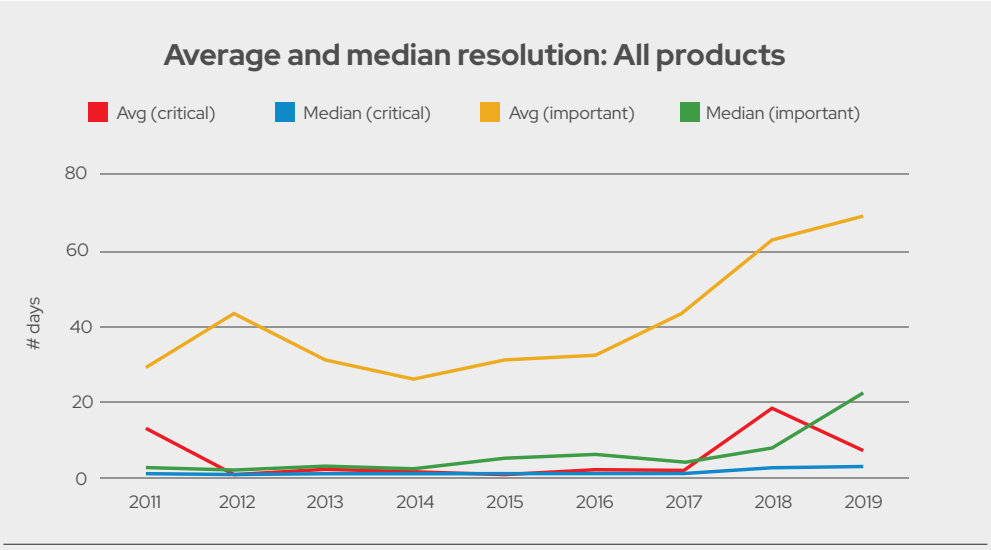

Figure 5. Average and median resolution

Risk is all about context.

Earlier we stated that across the whole portfolio we addressed Critical RHSAs within 7 days on average (with a median being 3 days and with 41% being addressed within 1 day of public disclosure). We were able to reduce our average number of days from 18 days in 2018 to 7 days in 2019. We did this through closely partnering with our Software Engineering and Quality Engineering teams throughout the organization.

Critical issues, in particular, should be addressed by subscribers as quickly as possible as they potentially carry very severe security defects. As we state on our classification page:

> *"This rating is given to flaws that could be easily exploited by a remote unauthenticated attacker and lead to system compromise (arbitrary code execution) without requiring user interaction. These are the types of vulnerabilities that can be exploited by worms. Flaws that require an authenticated remote user, a local user, or an unlikely configuration are not classed as Critical impact."*

Let's put this in context. In 2015, Red Hat's portfolio comprised approximately 100 streams of products/versions. In 2019, we oversaw nearly 150 unique products/versions and services, with 4 versions of Red Hat Enterprise Linux in active support or development (which includes more than 13 unique kernel versions that needed some level of support).

One of the most important benefits our customers gain from their subscriptions is the stability and engineering that goes into the delivered products and services. Take, for example, the Linux kernel. Red Hat Enterprise Linux Engineering and Quality Engineering typically will spend 48-96 hours testing each release candidate errata to ensure that it fixes the issue at hand and continues to function as expected. Adding to this team's burden has been the explosion of new layered products that depend upon that kernel. Each of these products needs to be considered as lower-level changes are made. Even if the errata works as expected in Red Hat Enterprise Linux, there could still be a downstream impact to Red Hat OpenStack Platform or Red Hat Storage. This thorough, methodical testing helps ensure that when a subscriber gets a newly hardened package, it will not come with unintended surprises. No one likes a regression, especially on a production system.

In summary, delivery times have risen over the years due to the increase in number of CVEs, the number of packages covered by our support life cycles, and the expansion of our Extended Update Support offerings.

Now, let's look at the packages that had the most CVEs remediated in 2019:

Table 3. CVEs remediated by package

Component	Number of CVEs (this includes multiple affected version numbers of a product)	CWE counts (included if 15+ for top 5)
kernel	216	cwe-200(19), cwe-203->cwe-385(47), cwe-400(22), cwe-284(20), cwe-416(16)
thunderbird	156	cwe-120(24), cwe-416(27), cwe-829(24), cwe-843(15)

Component	Number of CVEs (this includes multiple affected version numbers of a product)	CWE counts (included if 15+ for top 5)
firefox	152	cwe-120(25), cwe-416(27), cwe-829(24), cwe-843(16)
chromium-browser	131	cwe-416(2), cwe-20(1), cwe-125(1)
jackson-databind	123	cwe-502(93), cwe-502->cwe-200(18)
kernel-rt	112	cwe-200(13), cwe-385->cwe-203(13), cwe-416(13)
mysql:8.0/mysql	95	n/a
rh-mysql80-mysql	95	n/a
java-1.8.0-ibm	69	cwe-20(8)
qemu-kvm-rhev	59	cwe-122(13), cwe-203->cwe-385(24)
qemu-kvm	44	cwe-203->cwe-385(32)
libvirt	39	cwe-203->cwe-385(32)
rh-mariadb102-mariadb	37	n/a
redhat-virtualization-host	35	cwe-203->cwe-385(10)
ansible	34	cwe-117->cwe-532(12), cwe-20(10), cwe-200(10)
rh-php71-php	29	cwe-122(6), cwe-125(7)
kernel-alt	23	cwe-416(3), cwe-476(4)
exiv2	21	cwe-125(7)
rh-php72-php	20	cwe-122(6)

It has been long known that web browsers are security-challenged. 2019's insights into Chromium and Firefox as two of the top four most patched packages should help illustrate to system administrators that if they do not need to browse the web from their infrastructure, those two packages might be better left not installed.

If we remove the Linux kernel, Chromium, and Firefox packages, the list makes much more sense. The Linux kernel is one of the most important packages of the operating system. Red Hat Engineering spent a lot of time incorporating security fixes throughout the year and ultimately fixed more than 216 kernel CVEs across all supported versions of Red Hat Enterprise Linux. Doing this helped Red Hat systems continue to remain stable. The kernel is the central hub of "operations" of a system, and Red Hat Quality Engineering spends 48-96 hours testing each kernel release to ensure that we do not have regressions or other unexpected consequences from fixing these security defects. Doing that more than 200 times takes a lot of care and effort.

The Common Weakness Enumeration (CWE) data helps Red Hat Product Security provide our kernel developers with a list of coding patterns to avoid and can help a system administrator and security practitioner understand the root coding problems that created that vulnerabile condition.

Looking back at Table 3, readers can see that CWE-203/CWE-385 was found to be the root cause of 47 kernel CVEs. CWE-203 is Information Disclosure Through Discrepancy, with CWE-385 being a more precise way to express that condition (Covert Timing Channel).

Readers will also see packages like jackson-databind and assorted flavors of Java™ highly seated on the list. These middleware tools help run many enterprise applications, so they are of great interest to our customer base. Jackson-databind is especially helpful to have around because it is used to transform data from various formats (like JSON or Java objects). However, users must weigh the potential risk, particularly if that sort of operation is not being performed. Readers will also note that 93 out of the 127 CVEs related to jackson-databind were due to CWE-502: Deserialization of Untrusted Data. This issue and the kernel issues highlighted above relate to how developers are accessing, manipulating, and storing data with their code. Education supporting good data handling and sanitization processes can help reduce the frequency of those coding errors in the future.

We will reiterate here that if you do not need a package or daemon running on your system, you should remove or disable it to reduce your attack surface and reduce the amount of future maintenance and testing you will have to do by applying security patches.

Which issues were branded, and which issues really mattered in 2019

Where were you in early April 2014 when CVE-2014-0160, aka OpenSSL's "Heartbleed," came into all of our lives? Many of us can tell you precisely where we were when we got the news about OpenSSL's "little" problem. The Heartbleed vulnerability (CVE-2014-0160) was a wakeup call that greatly impacted the security landscape for the technology industry. This was the first time a computer flaw was "branded." This new naming practice drew substantial media and consumer interest and has shaped vulnerability responses across the industry since. The effectiveness of branding is highly debatable (a report from July 2019 cites that more than 90,000 systems worldwide were still exploitable to Heartbleed), but that has not slowed the branding trend.

Given all the threats that could potentially occur within your enterprise, it is important to have calm, clear guidance about which vulnerabilities to prioritize. An ominous name does not mean that a vulnerability is a pressing problem for you or your organization. Managing vulnerabilities is managing risk: something that might happen, given certain circumstances. No two companies have the same risks, even though they might be susceptible to the same vulnerabilities. What greatly concerns a major bank in the Czech Republic might not make the top 10 most concerning things to a retailer out of New Zealand. The conversation about risk is a conversation about perspective.

We have seen it all over the years: melting ghosts, tenacious zombies, vulnerabilities with theme songs, and the ever-popular T-shirt sales. We also have seen counter-marketing campaigns when the hyped-up bug really did not matter. Not all flaws are equal, so do your homework and find trusted sources that can provide you the vital details about how they might or might not affect the solutions you created and support. Manage your risks. Don't let your risks manage you.

The following issues impacted Red Hat offerings throughout 2019, and many of them were handled through our Customer Security Awareness program:

runc malicious container escape (11Feb2019) CVE-2019-5736

Severity rating: IMPORTANT **CVSSv3 score:** 7.7 **CWE:** CWE-672: Operation on a Resource after Expiration or Release

A flaw in the runc and docker packages could have allowed an attacker to escape from a container and gain root access to the container host. Unlike upstream, Red Hat OpenShift implements SELinux in enforcing mode (in fact, the product does not work correctly if it ever gets turned off), so it was immune to this threat. Default SELinux profiles for Red Hat Enterprise Linux also protected against this attack unless they were otherwise disabled or modified. Red Hat subscribers could wait to deploy package updates at their own cadence with the peace of mind that the attack could not be executed, unlike other docker/runc environments that had to scramble to get updates out.

Red Hat had patches available for impacted versions of Red Hat Enterprise Linux and on-premise Red Hat OpenShift Container Platform on the same day the issue went public. Red Hat OpenShift Online and Red Hat OpenShift Dedicated hosted environments were highly resistant to this vulnerability.

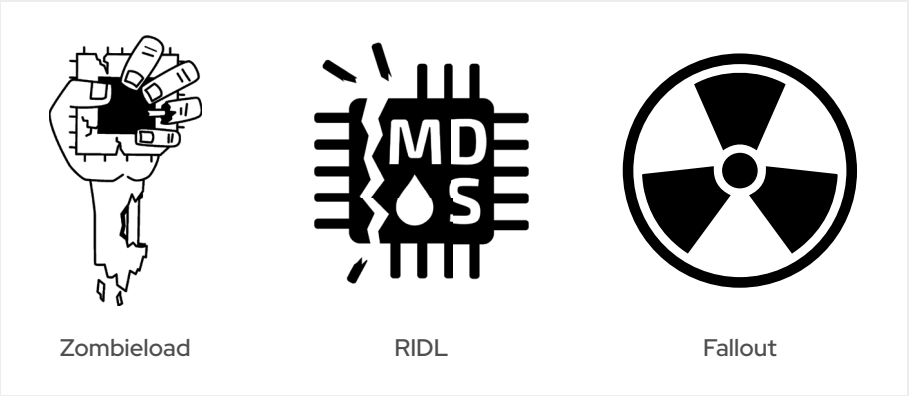

Figure 6. Some vulnerabilities from 2019

MDS - Microarchitectural Data Sampling (14May2019) CVE-2018-12130 (Zombieload & RIDL), CVE-2018-12126 (Fallout), CVE-2018-12127 (RIDL) & CVE-2019-11091 (RIDL)

Severity rating: IMPORTANT (CVE-2018-12130) and MODERATE (the rest) **CVSSv3 score:** 6.2
CWE: CWE-203->CWE-385: Operation on a Resource after Expiration or Release

In May, we saw several Intel CPU side-channel vulnerabilities, relatives to 2018's Spectre, Meltdown and others. Researchers referred to these particular flaws as RIDL, Fallout, and ZombieLoad, and they are collectively referred to as Microarchitectural Data Sampling (MDS). At a very high level, they all work the same way and yield similar results: An attacker can train a CPU to share data in the CPU memory that is on the same physical core as where the attacker has access. The gadgets and memory regions changed each vulnerability, but these are incredibly difficult attacks that could expose small amounts of data over very long periods of time to an attacker that is already resident on the system.

The likelihood of being targeted by such a well-funded and sophisticated attacker is astronomically small. Disabling certain features of the system (hyperthreading/simultaneous multithreading, HT/SMT) eliminates the threat, as does understanding where your critical data runs and what other tenants might co-locate on that hardware. Isolating critical data systems by not allowing untrusted workloads to share the same CPU removes any chance of data exposure. Updates should be applied, but the speed should be dictated by the risk tolerance of the organization and the threat actors you feel are your true adversaries.

Red Hat worked with Intel and a cross-industry group on this and all the side-channel issues. Patches were available starting at day zero, when the flaw was announced to the world. All of these Spectre-like issues required updated CPU microcode as well as kernel and virtualization patches.

TCP SACK Panic (17June2019) CVE-2019-11477, CVE-2019-11478, & CVE-2109-11479

Severity rating: IMPORTANT (CVE-2019-11477) and MODERATE (the rest) **CVSSv3 score:** 7.5 **CWE:** CWE-190->CWE-400: Integer Overflow or Wraparound leads to Uncontrolled Resource Consumption

June brought us back to the realm of "classic" problems with some good old-fashioned kernel networking bugs. All three issues revolved around how the Linux kernel handled TCP Selective Acknowledgement (SACK) packets. A remote attack could exploit these flaws to perform a denial-of-service attack against a machine and trigger a kernel panic, thus causing the system to be unavailable.

Red Hat provided mitigations and patches when the issue was made public.

Libvirt privilege escalation (20June2019) CVE-2019-10161, CVE-2019-10166, CVE-2019-10167, & CVE-2019-10168

Severity rating: IMPORTANT **CVSSv3 score:** 8.8 **CWE:** CWE-284: Improper Access Control

Another traditional vulnerability was released in late June. The libvirt maintainers discovered several vulnerabilities in the libvirt management API that allowed for four escalation of privilege attacks, one that allowed for escalation to root in Red Hat Enterprise Linux 8. Libvirt is a crucial component to Red Hat's virtualization products.

This issue is interesting in that Red Hat feels this vulnerability is more severe than what NVD did. Due to the impacts to our portfolio and our recognition that the scope of the attack changed, we gave this issue a CVSSv3 score of 8.8 (CVSS:3.0/AV:L/AC:L/PR:L/UI:N/**S:C**/C:H/I:H/A:H), while NVD gave it

a CVSSv3 score of 7.8 (CVSS:3.0/AV:L/AC:L/PR:L/UI:N/**S:U**/C:H/I:H/A:H). This again highlights that intimate knowledge of the solution is superior to generalized understanding and allows us to better depict what might happen with a vulnerability.

Red Hat released fixes for all affected versions of Red Hat Enterprise Linux starting the same day as public notification, with Red Hat Virtualization following close behind several days later.

Spectre SWAPGS gadget vulnerability (6August2019) CVE-2019-1125

Severity rating: MODERATE **CVSSv3 score:** 5.9 **CWE:** CWE-284: Improper Access Control

SWAPGS was a new way to exploit the Spectre v1 attack. With it, a local attacker could circumvent existing security controls and be able to read parts of privileged memory within a CPU. Side-channel attacks have become one of the most researched areas within information security in the last year, and it is expected that novel variations of these types of exploits will continue to trickle out for the foreseeable future.

To date, this exploit has not been proven to work within a Linux environment.

Red Hat began releasing updates to protect against this potential problem within one business day of the public disclosure.

VHOST-NET Guest-to-Host Escape (17Sept2019) CVE-2019-14835

Severity rating: IMPORTANT **CVSSv3 score:** 7.2 **CWE:** CWE-120: Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')

In September, a new vulnerability emerged in the Linux kernel's virtualization functionality. If a privileged attacker had access to a guest machine within a Red Hat Enterprise Linux-based or Red Hat Virtualization-based environment, during a live migration event (when a guest machine is moved to a different system or CPU within the virtual environment), the attacker could trigger a buffer overflow and escalate their privileges on the host system.

Virtualization and container escapes are increasingly concerning due to the large industry shift from physical systems to virtualization or cloud-based deployments. Red Hat Product Security works to bring awareness to our customers as these scenarios arise because so many of our subscribers are affected by these situations. Problems like these are why we created our Customer Security Awareness program. Early warnings when your systems may be at risk can be the difference between having a quiet day or a very bad one. Forewarned is forearmed.

Red Hat provided mitigations as soon as the issue went public and started providing patches to resolve the issue within the same business week.

sudo: Privilege escalation via 'Runas' (14Oct2019) CVE-2019-14287

Severity rating: IMPORTANT **CVSSv3 score:** 7.0 **CWE:** CWE-267: Privilege Defined With Unsafe Actions

Sudo is one of the cornerstones of *nix security. When CVE-2019-14287 came out, it rightly caused major anxiety within the open source world. A flaw was introduced in the way sudo implemented running commands from arbitrary users (especially if a command was allowed to be run by anyone and excluded the root account) so that no additional commands were allowed to be executed with escalated privileges.

Red Hat did not ship any sudo configurations with the defective configuration options being used by default. By default, neither Red Hat Enterprise Linux nor Red Hat Virtualization, while including the impacted package, was susceptible to attacks using this vulnerability due to default configuration and hardening changes made to the product. While there is no way to stop administrators from creating or deploying bad configurations, Red Hat quickly provided updated packages to use in case anyone did.

Red Hat released patches to remediate this issue starting within 10 days of public disclosure.

Machine Check Error on Page Size Change (12Nov2019) CVE-2018-12207

Severity rating: IMPORTANT **CVSSv3 score:** 6.5 **CWE:** CWE-805: Buffer Access with Incorrect Length Value

A flaw in Intel CPUs again allowed an unprivileged local attack to cause mayhem, this time by causing a denial-of-service attack. Like Meltdown, Spectre, POP SS, SSB, L1TF, and MDS before them, Machine Check Error on Page Size Change was something worth addressing, but the complexity of the attack limited the types of attackers that could execute an exploit using this vulnerability as well as the probability it might actually be used over other simpler flaws that could exist on a system.

Red Hat provided kernel and microcode packages to address this issue starting at the public disclosure time when the issue went public.

Transactional Synchronization Extensions (TSX) Asynchronous Abort (12Nov2019) CVE-2019-11135

Severity rating: MODERATE **CVSSv3 score:** 6.5 **CWE:** CWE-385->CWE-203: Covert Timing Channel leads to Information Exposure Through Discrepancy

Alongside the Machine Check Error on Page Size Change flaw came the Transactional Synchronization Extensions Asynchronous Abort. This was an MDS-style attack that allowed a local user to circumvent security and potentially read sensitive data. This particular technique exploited the SMT/HT and TSX implementations of the CPU, and disabling either or both drastically reduced the attack surface for this vulnerability to be exploited.

Red Hat provided kernel and microcode packages to address this issue starting at the time when the issue went public.

i915 Graphic Driver flaws (12Nov2019) CVE-2019-0155 & CVE-2019-0154

Severity rating: IMPORTANT (CVE-2019-0155) and MODERATE (CVE-2019-0154) **CVSSv3 score:** 8.8 **CWE:** CWE-284: Improper Access Control

Two issues impacting embedded i915 GPUs were revealed in November. The more severe of the two issues allowed an attacker to gain write access to privileged memory, and the lesser allowed a denial-of-service attack through the graphics card.

Red Hat started releasing updates to both of these flaws starting within one business day of public disclosure.

A tour of the most significant vulnerabilities in 2019

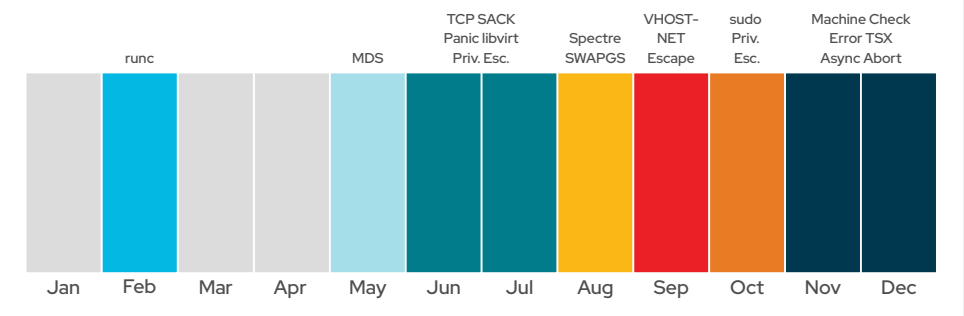

Figure 7. Significant vulnerabilities by month

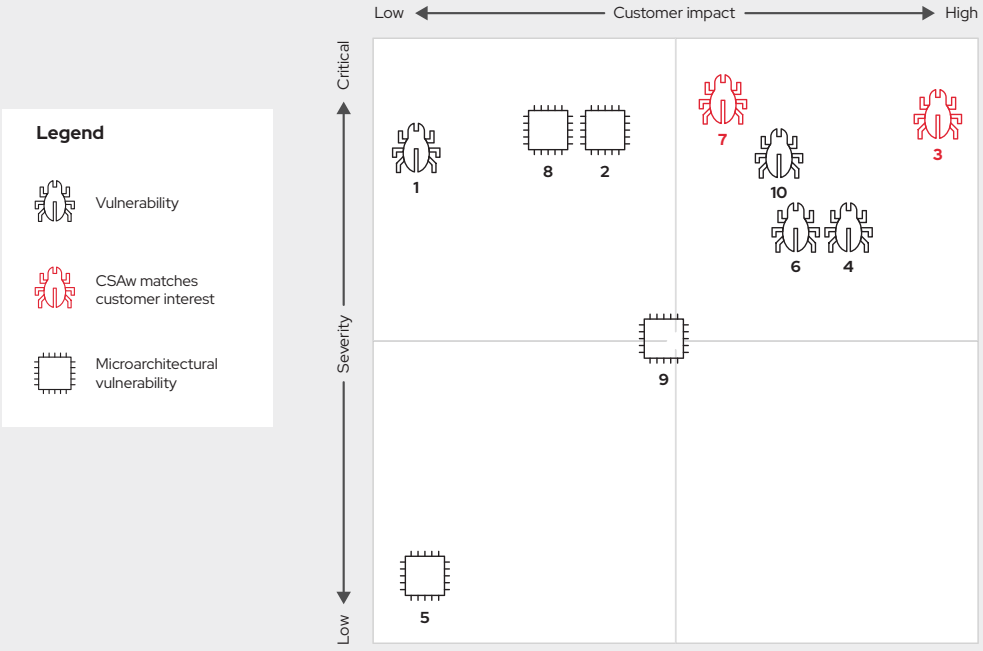

Figure 8. Vulnerabilities by severity and impact

Table 4. Significant vulnerabilities in 2019

Issues denoted in red were branded flaws. Issues denoted by a computer chip were related to microprocessor issues

CVE	Name	Severity
CVE-2019-5736	runc malicious container escape	IMPORTANT
CVE-2018-12130, CVE-2018-12126, CVE-2018-12127 & CVE-2019-11091	MDS - Microarchitectural Data Sampling	IMPORTANT / MODERATE
CVE-2019-11477, CVE-2019-11478, & CVE-2109-11479	TCP SACK Panic	IMPORTANT / MODERATE
CVE-2019-10161, CVE-2019-10166, CVE-2019-10167, & CVE-2019-10168	libvirt privilege escalation	IMPORTANT
CVE-2019-1125	Spectre SWAPGS gadget vulnerability	MODERATE
CVE-2019-14835	VHOST-NET Guest-to-Host Escape	IMPORTANT
CVE-2019-14287	sudo: Privilege escalation via 'Runas'	IMPORTANT
CVE-2018-12207	Machine Check Error on Page Size Change	IMPORTANT
CVE-2019-11135	Transactional Synchronization Extensions (TSX) Asynchronous Abort	MODERATE
CVE-2019-0155 & CVE-2019-0154	i915 Graphic Driver	IMPORTANT / MODERATE

In Figure 8 above, we have plotted the 10 most "interesting" vulnerabilities based on two data factors: the severity of the CVE(s) and the impact to our customers. Each customer is unique. Customers might not use a particular package, or they might have other controls in their environment that decrease the residual risk of a given flaw. Conversely, a system affected by any of the 1,313 vulnerabilities addressed in 2019 could support mission-critical systems or have other factors that increase the risk for the affected organization.

Another way we gauge customer interest around an issue is to measure web traffic, specifically views for each CVE page in the Red Hat Customer Portal.

Gauging customer concern

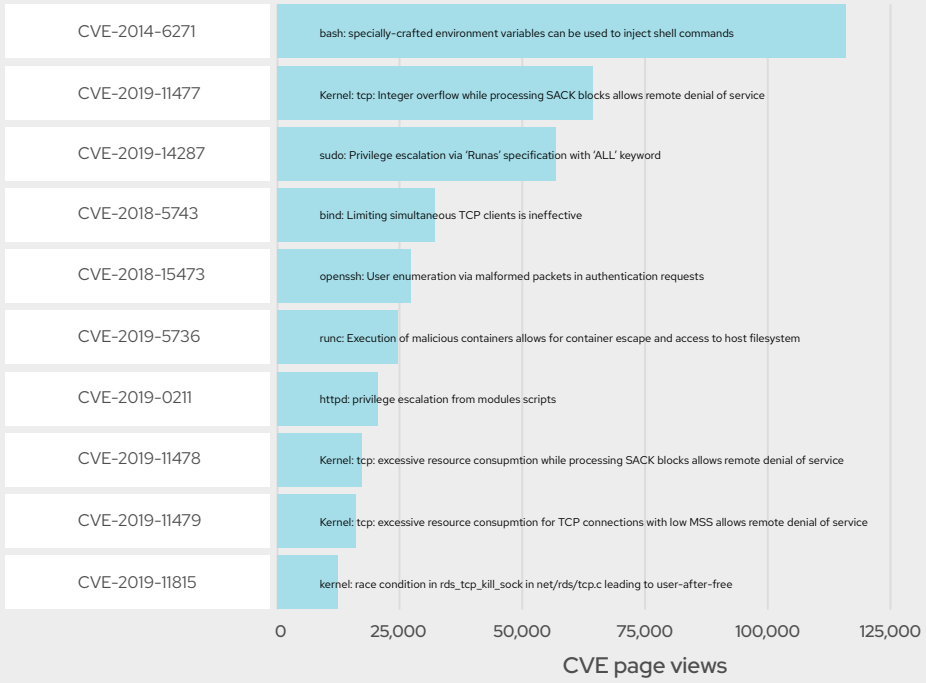

Figure 9. Most viewed vulnerabilities in 2019

These issues were the most viewed vulnerabilities from the Red Hat CVE database during the 2019 calendar year. Customer interest in a 5-year-old bash vulnerability occupied a significant amount of attention over the year, bringing in nearly twice the page views of the next most visited CVE page, TCP SACK.

2019 was a year of high-profile hardware issues that dominated the headlines alongside several Important issues in virtualized and containerized platforms that impacted a broad spectrum of Red Hat subscribers.

The open source supply chain

Red Hat is an enterprise software company that uses an open source development model to compose our products. The source code is derived from upstream open source projects, and any changes we make are made public for both upstream and downstream consumers of the software. We feel strongly about placing the open source community first to ensure that development is done in the open and so that we can take advantage of all of the creativity and expertise of the global community of developers. Each Red Hat-branded solution comprises hundreds to thousands of open source packages and projects. These communities provide vital contributions that become the foundation for our enterprise-class products and services.

Red Hat Product Security's mission is twofold:

- Help oversee the productization of these solutions that ultimately are deployed in companies around the world, ensuring that they are composed, managed, and delivered in a security-focused manner.
- Monitor the components of the solutions we provide and – as vulnerabilities are discovered – document, describe, and work with Product Engineering to address those vulnerabilities appropriately.

We work closely with internal Red Hat teams (Product Engineering, Quality Engineering, Support, etc.) and external peers and security researchers.

In 2019, Red Hat Product Security investigated 2,714 vulnerabilities that potentially affected components of our products and services. From this large list, we determined that 1,313 of these reports were vulnerabilities that required us to take action. We recorded each of these reports in our public Bugzilla system and shared them externally once any embargoes were over. Each issue that impacts our products is assigned a CVE, a Red Hat severity score, and CVSS score. All of this data is available through multiple streams for anyone to review:

- Metrics webpage
- Red Hat Security Vulnerability Data API
- OVAL and CVRF data feeds
- RHSA announcements

We use this data to create metrics and review trends with Product Engineering to help improve future releases and the whole open source ecosystem.

Red Hat does not wait for problems to come to us – we proactively seek out problems that could affect our offerings and our subscribers, and we address these vulnerabilities as early as possible. Approximately 30% of critical issues we addressed came to us directly from peers, Red Hat employees, or Red Hat customers. This is slightly up from 2018, but the number of total flaws reported grew 3.2% over the previous year. Whenever possible, we share these issues with the upstream and our industry peers. Additionally, Red Hat may also find and report flaws in software that are not part of our currently shipped products. When it comes to fixing issues in third-party software, relationships matter. Red Hat Product Security and Product Engineering have deep ties with upstream communities and the technology industry at large. We are constantly communicating and collaborating with our peers on issues that impact all of our shared customers and communities.

If an upstream community is willing to share information about flaws with us in advance, we feel a responsibility to give value back for that shared trust. We do this by reviewing advisories, checking patches, and feeding data back from our quality or performance testing groups. Ultimately we are all focused on providing remediation to the flaws, and we all try to contribute positively to the solution as it is evolving.

Conclusion

Risk is a topic that is different for every organization and one we have discussed through our Security in the modern IT world blog series. It manifests itself differently based on the data, system, or personnel that are in danger. How a retailer reacts to a particular problem could be different from how a medical provider reacts. In general, both types of users want to eliminate vulnerabilities, but the retailer might be most concerned about point-of-sale terminals, whereas the medical security team might be valiantly trying to overcome problems in embedded medical devices used by patients.

Risk is all about context. Worry about the threats that pertain to you and that impact your most sensitive things. Having a solid vulnerability management program will allow you to know what threats could impact you, where your most important things are, and where and when you have to react most quickly.

No control is perfect and protects against every vulnerability. Think holistically about your security controls, and have layered or overlapping protections in place. Know where your critical data and systems are, and focus your security efforts there for maximum effectiveness. Ideally, you will stop all your attackers, but if not, these blended controls should alert you of some maleficence and allow you to react quickly before the incident gets out of your control.

We hope that this report has provided useful information. For the latest in what is going on with security for Red Hat products and services, see the Red Hat Product Security Center.

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

facebook.com/redhatinc
@RedHat
linkedin.com/company/red-hat

redhat.com
#F21332_0320