SUSE Solution Security Risk Report 2023

All SUSE Products

Stoyan Manolov, Head of Solution Security (SUSE)

SUSE Solution Security is committed to delivering best-in-class software security to its customers and to the open source community. The primary objectives are to treat software security as an ongoing and continual process.

The goal of this report is to provide a summary of all security vulnerabilities which affected SUSE products in calendar year 2023.

Disclaimer: This document is part of the SUSE Best Practices series. All documents published in this series were contributed voluntarily by SUSE employees and by third parties. If not stated otherwise inside the document, the articles are intended only to be one example of how a particular action could be taken. Also, SUSE cannot verify either that the actions described in the articles do what they claim to do or that they do not have unintended consequences. All information found in this document has been compiled with utmost attention to detail. However, this does not guarantee complete accuracy. Therefore, we need to specifically state that neither SUSE LLC, its affiliates, the authors, nor the translators may be held liable for possible errors or the consequences thereof.

Contents

1 Motivation

SUSE Solution Security is committed to delivering best-in-class software security to customers and to the open source community. The primary objectives are to treat software security as an ongoing and continual process that never ends. This implies to:

- promptly react to security incidents and deliver premium quality security updates.
- continuously improve the security-related functionality in SUSE products.
- continuously contribute to the rapidly growing maturity of open source software.
- respect the open source software security principles of openness, transparency and traceability.

The SUSE Security Team addresses all aspects of software security on an ongoing basis. Software security cannot be thought of as a state you can achieve at a specific point in time. Instead, it is a process that must be executed with professional expertise and continuous development. This persistent focus is what has given open source software, Linux and SUSE an excellent reputation for security. Our SUSE Solution Security team is responsible for handling all SUSE product-related security incidents. In that team, clear and well-defined roles are assigned for tracking new incidents and coordinating needed updates. The team closely collaborates with all SUSE software engineer specialists.

The objective of this report is to provide a summary of all security vulnerabilities which affected SUSE products in calendar year 2023. We will go into details on the high impact vulnerabilities and elaborate on how we responded to these incidents.

2 Background

A modern Linux operating system, such as SUSE Linux Enterprise Server for enterprise use or the openSUSE community distribution for home use, features a rich set of security programs and functions. Those range from access control, intrusion prevention and detection, flexible and trustworthy authentication mechanisms, encryption for files and network connections, file integrity checking utilities, to network analysis tools and monitoring/logging utilities for your system. To complement this, there are advanced tools that help you to securely configure and administer your system, and to securely download and install update packages. These utilities are standard in SUSE products. The update packages fix security bugs that have been found after your product has been released. The security features of your Linux system are waiting for you to explore them. SUSE encourages our customers to take advantage of them to further improve the level of privacy and security that is built into every system by default.

Programs are usually written by humans, and humans make mistakes. By consequence, all software can contain errors. Some of these errors appear as instabilities (the software or the entire system crashes), while others may not have any apparent, visible effect. However, some software errors may introduce a security risk. A local or a remote attacker may be able to feed specially drafted data to the software which takes advantage of the programming error. In the case of a remotely exploitable bug, the data comes from an attached network device, such as a cable or DSL modem, or a wireless network interface card. The application then either crashes, resulting in a Denial of Service (DoS) attack. Or it executes code that originates from the attacker, transferring control over the execution context from what the programmer intended to what the attacker has in mind for the exploitation of the error. Depending on the functionality of the software, the resulting security breach may pose a low or high security risk to your data and system, potentially giving an attacker the opportunity to delete, modify or even steal your data or exploit the system for their own purposes.

The SUSE Solution Security team is responsible for handling all SUSE product-related security incidents. In that team, clear and well-defined roles are assigned for tracking new incidents and coordinating needed updates. The team works with all SUSE engineering software specialists.

We use multiple sources to understand security incidents. These sources include the Mitre and NVD Common Vulnerabilities and Exposures (CVE) databases, various security mailing lists (OSS security, Linux distros, distros, bugtraq, and full-disclosure), direct reports, and other Linux vendors databases. We are also part of various pre-notification mailing lists for software components, like Xen, Samba, X.ORG. Confidential pre-notifications about vulnerabilities will be treated according to established responsible disclosure procedures.

3 Incident rating and tracking

We rate the severity of incidents with two different systems, a simplified rating system and the Common Vulnerability Scoring System (CVSS) v3.1 scoring system. The CVSS is an open framework for communicating the characteristics and severity of software vulnerabilities. It is being developed by the US-based non-profit organization FIRST.org: Its main goal is to assign the right score to a vulnerability to help security administrators prioritize responses and resources to specific threats. CVSS v3.1 scoring consists of three metric groups: Base, Temporal, and Environmental. The Base group represents the intrinsic qualities of a vulnerability that are constant over time and across user environments. The Temporal group reflects the characteristics of a vulnerability that change over time. The Environmental group represents the characteristics of a vulnerability that are unique to a user's environment. The Base metrics produce a score ranging from 0 to 10, which can then be modified by scoring the Temporal and Environmental metrics. A CVSS score is also represented as a vector string, a compressed textual representation of the values used to derive the score. Today, SUSE uses the Base score methodology to evaluate vulnerabilities throughout the support life cycle of our products. SUSE keeps the right to adjust the final score of the vulnerability as more details become known and available throughout the analysis. The most current CVSS resources can be found at https://www.first.org/cvss/ . The CVSS v3.1 calculator used by SUSE could be found at https://www.first.org/cvss/calculator/3.1 . The framework is measuring the severity of a given vulnerability, not the associated risk alone. The scoring of any vulnerability may vary with different analysts hence the final score could be slightly different between vendors impacted by that vulnerability. For a more accurate assessment of the impact, vendors and application owners must always consider factors outside of CVSS such as exposure or threat.

The security incidents are tracked in our own workflow system. Technical details are tracked in the SUSE bug-tracking system, and the updated software package is built, processed, and published by our internal “Open Build System”. Internal Service Level Agreements (SLAs) corresponding to the severity rating are monitored and reviewed regularly. Our packagers backport the required security fixes to our version of the software. To protect the stability of our customer setups, we only rarely do minor version upgrades. After receiving fixes for the affected software, four eye reviews cross-check the source patches. Several automated checks verify source and binary compatibility and the completeness of patch meta information. They also check whether patches can be installed without problems. Dedicated QA teams provide integration, bugfix, and regression testing for all updates before they are released to our customers. After the release of an update, automated processes publish the updates, update notices, and cross reference information on our CVE index pages and machine-readable OVAL and CVRF XML information.

For a better understanding of our classification mechanisms, we have described our rating system along with the equivalency of each rating to the CVSS v3.1 scoring calculator:

TABLE 1: INCIDENT RATING AND CVSS SCORE

Rating	CVSS Score	Definition
Critical	9.0 and above	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

Rating	CVSS Score	Definition
		exploited by worms. Flaws that require an authenticated remote user, a local user, or an unlikely configuration are not classed as critical impact.
Important	7.0 to 8.9	This rating is given to flaws that can easily compromise the confidentiality, integrity, or availability of resources. These are the types of vulnerabilities that allow local users to gain privileges, allow authenticated remote users to execute arbitrary code, and finally allow unauthenticated remote users to view resources that should otherwise be protected by authentication and to cause a denial of service without user interaction.
Moderate	4.0 to 6.9	This rating is given to flaws that may be more difficult to exploit but could still lead to some compromise of the confidentiality, integrity, or availability of resources, under certain circumstances. These are the types of vulnerabilities that could have had a critical impact or important impact but are less easily exploited based on a technical evaluation of the flaw, or affect unlikely configurations. Local, persistent (service needs to be restarted) denial of service conditions for basic system services (kernel, systemd, polkit, dbus, etc.) with and without user interaction should also be rated “moderate”.
Low	up to 3.9	This rating is given to all other issues that have a security impact. These are the types of vulnerabilities that are believed to require unlikely circumstances to be able to be exploited, or where a successful exploit would give minimal consequences.

4 When to prefer version upgrades over backports

It is a general policy rule that no new upstream versions of a package are introduced into our enterprise products. This rule is not an absolute rule however. For certain types of packages, in particular antivirus software, security concerns weigh heavier than the conservative approach that is preferable from the perspective of quality assurance. For packages in that class, occasionally newer versions are introduced to a released version of an enterprise product line.

Sometimes also for other package types the decision is made to introduce a new version instead of a backport. This happens when producing a backport is not economically feasible or if there is a very relevant technical reason for introducing the newer version.

5 Major security vulnerabilities in 2023

5.1 CVE-2023-38408: Remote code execution in OpenSSH's forwarded ssh-agent

Overview

In July 2023, The Qualys Threat Research Unit (TRU) has discovered a remote code execution vulnerability in OpenSSH's forwarded `ssh-agent`. This vulnerability allows a remote attacker to potentially execute arbitrary commands on vulnerable OpenSSH's forwarded `ssh-agent`.

Attackers must be able to access a host via SSH to escalate privileges on that host by exploiting a flaw in the `pkcs11` module loading of the SSH agent. As the `pkcs11` agent helper allowed loading of system dynamic libraries, certain loading patterns and problems in system libraries could be used to gain code execution as the `pkcs11` helper.

Solution

Installing the updated packages provided by SUSE is sufficient to fix the problem. Use

```
zypper lp -a --cve=CVE-2023-38408
```

to search for the specific patch information. A restart of the service is not required.

Note that for any SPx (Service Pack level) which is no longer in general support, you might need an LTSS or ESPOS subscription to obtain the update. See the SUSE “CVE Page” link in the “References” paragraph below for more details about each SPx.

Workaround

In case PKCS11 smartcards are not used for SSH agent support, remove `/usr/lib/ssh/ssh-pkcs11-helper` from the system until maintenance updates have been released.

The workaround prevents exploitation and might be the right thing to do given how easy the exploit it, but customers must be aware that this will break functionality until the update is installed.

References

- SUSE Web page for CVE-2023-38408:
  https://www.suse.com/security/cve/CVE-2023-38408.html
- Blog article:
  https://blog.qualys.com/vulnerabilities-threat-research/2023/07/19/cve-2023-38408-remote-code-execution-in-opensshs-forwarded-ssh-agent

5.2 CVE-2023-20593: AMD CPU: "ZenBleed" - VZEROUPPER does not clear upper bits under certain conditions

Overview

Researchers at Google have discovered Zenbleed, a hardware bug causing corruption of the vector registers.

When a VZEROUPPER instruction is discarded as part of a bad transient execution path, its effect on internal tracking is not unwound correctly. This manifests as the wrong micro-architectural state becoming architectural, and corrupting the vector registers.

Note: While this malfunction is related to speculative execution, this is not a speculative side-channel vulnerability.

The corruption is not random. It happens to be stale values from the physical vector register file, a structure competitively shared between sibling threads. Therefore, an attacker can directly access data from the sibling thread, or from a more privileged context.

Solution

Packages containing a fix for this security issue were made available quickly. To apply the fixes, install the new packages with the following command:

```
zypper patch --cve=CVE-2023-20593
```

References

- SUSE Web page for CVE-2023-20593:
  https://www.suse.com/security/cve/CVE-2023-20593.html ↗
- AMD Security Bulletin:
  https://www.amd.com/en/resources/product-security/bulletin/amd-sb-7008.html ↗
- GitHub security research article:
  https://github.com/google/security-research/security/advisories/GHSA-v6wh-rxpg-cmm8 ↗

5.3 CVE-2023-44487: HTTP/2 **Rapid Reset** attack

Overview

In August 2023, Amazon Web Services, Cloudflare and Google noticed a new type of distributed denial-of-service (DDos) attacks on their networks. These attacks had record breaking sizes, three times bigger than previous attacks.

As it turned out, the problem that was exploited was not an implementation bug, but an issue inside the internal design of the HTTP/2 protocol itself. The principle of the **Rapid Reset** attack is quite simple.

With HTTP/1.1, all requests to the server are processed serially on one connection. The client is sending a request, the server will read and process it and send a response. Then the next request is processed. The newer HTTP/2 protocol allows multiple bidirectional streams via a single TCP connection. A client can therefore send several requests at the same time, which are then answered by the server. This results in a much higher utilization of each connection.

The **Rapid Reset** attack now uses the fact that each of those inner streams can be canceled at any point in time via an RST_STREAM frame. This can be done even before data has been transmitted back to the client. The problem that arises is as follows: The request is processed by the server, and for this purposes, resources are allocated per stream. These resources must be deleted again a moment later when the RST_STREAM frame has arrived.

This comes at almost no cost for the attacker, but depending on the server implementation, it can result in significant resource utilization for the victim.

Solution

Several upstream projects updated their code to implement or extent the mitigation mechanisms that prevent or lower the impact of those attacks. This is usually done by setting a reset rate limit.

To install the respective patches, use:

```
zypper patch --cve=CVE-2023-44487
```

References

- SUSE Web page for CVE-2023-44487:
  https://www.suse.com/security/cve/CVE-2023-44487.html ↗
- Google Cloud blog article:
  https://cloud.google.com/blog/products/identity-security/how-it-works-the-novel-http2-rapid-reset-ddos-attack ↗
- Cloudflare blog article:
  https://blog.cloudflare.com/technical-breakdown-http2-rapid-reset-ddos-attack/ ↗
- Qualys blog article:
  https://blog.qualys.com/vulnerabilities-threat-research/2023/10/10/cve-2023-44487-http-2-rapid-reset-attack ↗

5.4 CVE-2023-48795: SSH prefix truncation attack (aka Terrapin Attack)

Overview

Security researchers from the Ruhr University Bochum have published a new attack on the SSH v2 protocol, which allows active person-in-the-middle attackers to impact SSH connections by removing initial encrypted SSH packets.

This can lead to protocol security downgrades or similar problems. Changing SSH packages or injecting new encrypted SSH packages is not possible with this attack.

Software on all SUSE Linux Enterprise versions were originally affected. The problem is inherent to the existing SSH v2 protocol, so new protocol addition(s) and enforcement of them are needed to avoid the problem.

The protocol vulnerability needs to be exploited in tandem with specific SSH ciphers. The `chacha20-poly1305` SSH cipher is the one that was shown to be exploitable most easily, also other ciphers using Encrypt-Then-MAC Message Authentication Codes (MACs) might be exploitable under certain conditions.

Note that the ciphers themselves are not problematic. They could lead to exploitable scenarios only in combination with the SSH v2 protocol weakness.

Find below a list of SSH v2 implementations that are shipped by SUSE and their exploitability status:

- `openssh`: is affected in all shipping versions up to 9.5p1. All versions of SUSE Linux Enterprise Server are affected.
- `putty`: is affected. It is shipped via SUSE PackageHub 15.
- `libssh.org` (aka libssh): supports `chacha20-poly1305` since 0.8.0. SUSE Linux Enterprise Server 12 SP5 and SUSE Linux Enterprise Server 15 SP1 and newer are affected.
- `libssh2.org` (aka libssh2_org): does not implement the `chacha20-poly1305` cipher in the newest release 1.11.0. ETM MACs were only implemented in 1.11.0, therefor versions before 1.11.0 are not affected. SUSE Linux Enterprise Server 12 SP5 and SUSE Linux Enterprise Server 15 come with version 1.11.0 and are affected by this problem.

- `jsch` (Java SSH): `chacha20-poly1305` was added with version 0.1.66, ETM MACs in 0.1.58. Versions from 0.1.58 to current 0.2.9 are considered affected. SUSE currently does not ship affected versions of `jsch`.
- `proftpd`: its `mod_sftp` module, which is shipped via SUSE PackageHub 15, is affected. It supports ETM MACs, but not `chacha20-poly1305`. The module, however, is not enabled by default.
- `golang.org/x/crypto/ssh`: The Golang SSH module is also affected. The SSH module is used and/or included by a long list of software written in GO.s of an issue, as in scenarios relying on secure boot, like public systems.

Solution

The solution is to install respective updates on server and client machines.

`openssh` updates were provided on December 18th. Other SSH software was also updated after backporting upstream security fixes. Note that both SSH clients and servers must be adjusted for the protocol adjustments to be effective.

Mitigations like removal of ciphers can be done on either side to be effective. Bacuase rollout may take some time and not all clients are under administrative control, configuration adjustments such as removing ciphers should be made to avoid using affected ciphers (see the References section below).

References

- SUSE Web page for CVE-2023-48795:
  https://www.suse.com/security/cve/CVE-2023-48795/
- SUSE TID 000021295:
  https://www.suse.com/support/kb/doc/?id=000021295
- Web page:
  https://terrapin-attack.com/

5.5 TTY injection via sudo

Overview

The Linux TTY subsystem allows pushing back keypresses into the TTY stack. This is a not well-known feature and may lead to unexpected consequences or potential attacks. As an example, in cases where untrusted code using a command such as **`sudo`** is executed, the untrusted code could push keypresses into the TTY stack, which could then be executed after the **`sudo`** execution completes.

While **`sudo`** is commonly used for transitioning to “root” where this would not be an issue, other **`sudo`** scenarios are possible where this could be used to escalate privileges. The untrusted code would then be able to execute commands as “root” using this attack.

To avoid this injection, **`sudo`** has the option to create a new pseudo terminal device (PTY), which is discarded after running **`sudo`**, and would not be affected by this keypress push back attack. However, note that this might impact terminal output or the use of interactive keys like backspace or delete.

Solution

Add the line

```
Default use_pty
```

to the `/etc/sudoers` configuration file using the **`visudo`** command. Starting with `sudo` version 1.9.14p1, this setting is the default for new installations.

References

- SUSE TID 000021241: https://www.suse.com/support/kb/doc/?id=000021241

5.6 CVE-2022-40982: CPU transient information leakage from GATHER instructions aka “Gather Data Sampling” aka “DOWNFALL”

Overview

Security researcher Daniel Moghimi has identified a transient information leakage from GATHER instructions on modern Intel CPUs (**Skylake** to **Tiger Lake** generations).

This can be used to reveal secret data contained in vector registers, which can essentially be any information due to the use of these instructions in memory copy operations. This information leak can cross process and privilege boundaries.

Solution

Intel has released CPU microcode to mitigate these issues. The CPU microcode is mandatory for mitigation, the mitigation is enabled by default.

Updated `ucode-intel` packages released by SUSE, version 20230808 or later, contain the mitigations. If there is no CPU Microcode available, the **`avx`** instructions can be hidden from CPUID reporting by the kernel.

In addition, kernel and XEN changes are being applied to:

- allow disabling the mitigation.
- report affectedness of the CPU and state of the mitigation.

SUSE will release kernel and XEN updates.

Mitigation reporting

The mitigation state is reported via the sysfs file `/sys/devices/system/cpu/vulnerabilities/gather_data_sampling`. The file can report the following states:

- Not affected
  This processor is not vulnerable.
- Vulnerable

This processor is vulnerable and the mitigation is disabled.

- Vulnerable: No microcode
  This processor is vulnerable and the microcode is missing mitigation.
- Mitigation: Microcode
  This processor is vulnerable and the mitigation is in effect.
- Mitigation: Microcode (locked)
  This processor is vulnerable and the mitigation is in effect and cannot be disabled.
- Unknown: Dependent on hypervisor status
  Running on a virtual guest processor that is affected but with no way to know if the host processor is mitigated or vulnerable. This can happen if the hypervisor does not expose necessary MSR registers to guests.

Kernel command line options:

- `gather_data_sampling=off`
  Switch off this specific mitigation (the default is "on").
- `mitigations=off`
  Switch off all CPU transient execution mitigations, including the `gather_data_sampling` one.
- `clearcpuid=avx`
  This option hides the AVX instructions from CPUID flags. Therefore, existing optimized code will not use AVX instructions, or fallback to other variants, and thus not expose the vulnerability.

Performance considerations

The mitigation impacts performance of the GATHER parts of the AVX2 and AVX512 instructions, and potentially operations that are translated within the CPU to use these vector instructions (for example potentially REP MOVS, or other instruction sets like SSE).

According to the technical paper linked from the Intel Product Security Center Advisory listed below, the impact might be as follows:

"When the mitigation is enabled, there is additional latency before results of the gather load can be consumed. Although the performance impact to most workloads is minimal, specific workloads may show performance impacts of up to 50%."

References

- SUSE Web page for CVE-2022-40982:
  https://www.suse.com/security/cve/CVE-2022-40982.html

- Downfall attacks Web page:
  https://downfall.page/

- Intel Product Security Center Advisory INTEL-SA-00828:
  https://www.intel.com/content/www/us/en/security-center/advisory/intel-sa-00828.html

5.7 CVE-2023-24932: BlackLotus bootkit

Overview

A UEFI bootkit called BlackLotus has been spotted in the wild, which uses buggy secure boot binaries to bypass the UEFI secure boot chain. While this component is not shipped by SUSE, to ensure the integrity of the UEFI secure boot chain, the affected components will need to be excluded from the UEFI secure boot space. Any UEFI secure boot-enabled installation could be impacted by this vulnerability.

Solution

You can exclude affected components via a so-called `DBX` file which blocks UEFI binaries using `signatures/has`. `DBX` file updates can be delivered and deployed in different ways:

1. Via BIOS or Windows updates
   The DBX file can be delivered via BIOS updates. In case of a dual boot system, it is delivered with Windows updates. If it is installed via BIOS or Windows updates, no additional deployment on Linux is required.

2. Via the firmware update tool (`fwupd`)
   Download the current DBX revocation list from https://uefi.org/revocationlistfile for the respective hardware platform. Use the command **`fwupdmgr install x64_DBXUpdate.bin`** and follow the prompts. This update method only works starting with SUSE Linux Enterprise 15 SP3 based products and newer.
   Note: This is a manual deployment and is only required if there are concerns about UEFI bootkits, as they are not a common attack scenario.

References

- SUSE Web page for CVE-2023-24932:
  https://www.suse.com/security/cve/CVE-2023-24932.html ↗
- Microsoft Support KB5025885:
  https://support.microsoft.com/en-us/topic/kb5025885-how-to-manage-the-windows-boot-manager-revocations-for-secure-boot-changes-associated-with-cve-2023-24932-41a975df-beb2-40c1-99a3-b3ff139f832d ↗
- UEFI Revocation List File:
  https://uefi.org/revocationlistfile ↗

5.8 CVE-2023-29552: New SLP-based traffic amplification attack

Overview

Security researchers Pedro Umbelino at Bitsight and Marco Lux at Curesec have published a new network traffic amplification attack using the Service Location Protocol (SLP, RFC 2608). This new attack allows for up to 2200x amplification, enabling very effective distributed denial of service attacks.

This flaw is a network protocol design flaw, a software fix is not easily possible.

Solution

The SLP protocol IP ports 427, both UDP and TCP, outside of your network boundary should be blocked or filtered.

References

- SUSE Web page for CVE-2023-29552:
  https://www.suse.com/security/cve/CVE-2023-29552.html ↗
- SUSE TID 000021051:
  https://www.suse.com/support/kb/doc/?id=000021051 ↗

6 Vulnerability management in 2023

The SUSE Solution Security team constantly monitors all the software components used in our products for security issues. More and improved tools are now available to find zero-day vulnerabilities and scan for existing vulnerabilities. Such tools can check and report whether the application code written conforms to standard security best practices or whether there are major holes in potential attack vectors such as buffer overflow, denial of service, or unwanted elevated access. It is clear that developers are becoming more and more security-aware and that the quality of the code being developed has improved significantly, both in quantity and quality. While we are seeing an increasing number of important vulnerabilities, the number of critical vulnerabilities is decreasing year over year.

TABLE 2: VULNERABILITIES WITH A UNIQUE CVE IDENTIFIED, IMPACTING SUSE PRODUCTS IN 2023

Low	Moderate	Important	Critical
123	597	387	9

TABLE 3: SECURITY UPDATES AND PATCHES RELEASED TO FIX THESE VULNERABILITIES IN 2023

Low	Moderate	Important	Critical
49	718	2137	37

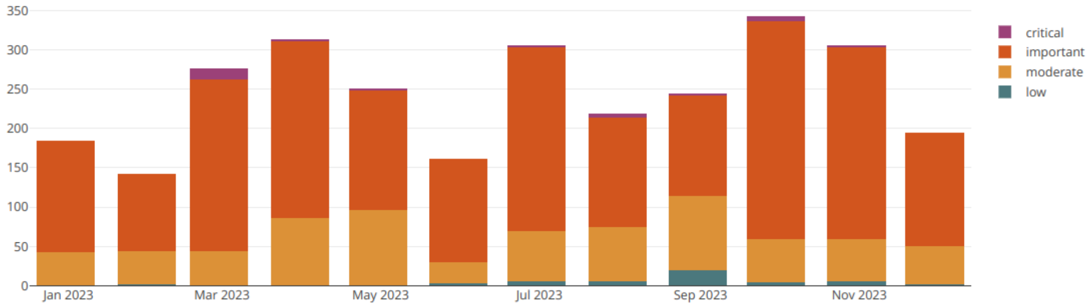

FIGURE 1: SECURITY UPDATES AND PATCHES RELEASED PER MONTH IN 2023

7 Securing the software supply chain

Securing our software supply chain is a top priority for SUSE to protect our customers from security risks, known and zero-day vulnerabilities. Ensuring that no threat actor can inject malicious code into our build service systems is certified by industry leading security certifications. Our teams continually work to certify all SUSE products, and develop security solutions to offer our customers the highest level of trust and reliability.

7.1 Proactive review of code

Our skilled team of security engineers regularly reviews the code we ship. We harden our products and conduct security audits on selected packages. Packages and products are continually reviewed to ensure we protect our customers' systems with the latest state of the art technology. To this end, we have introduced an automation setup that notifies our team of potentially dangerous changes in existing or new packages (for example important permission changes) and triggers an audit if necessary.

Over the last year, our proactive Security team has found and assigned CVE IDs to a number of vulnerabilities which document the more important findings. Find more information on our blog page (https://security.opensuse.org/) ↗.

7.2 Software Bill of Materials (SBOM) available

Following recent supply chain attacks and increasing security automation, software inventory management in particular is becoming increasingly important. Governments and other regulated industries now require the publication of a so-called Software Bill Of Materials (SBOM) for software products.

Various SBOM formats have appeared on the market. SUSE has begun releasing SBOM in the two formats most commonly used by operating system vendors: SPDX 2.0 and CycloneDX. SPDX 2.0 has even been standardized in ISO/ICE 5962:2021.

SUSE Linux Enterprise product media:

For our product media (ISO images), the SBOM materials are available from our download Web site in both SPDX 2.0 and CycloneDX formats. The granularity of this data is currently at RPM level.

SUSE Linux Enterprise BCI container images:

For our BCI container images, the SBOM data is delivered in the sigstore attestation blob in SPDX 2.0 format. The data is supposed to be processed automatically, but manual retrieval is also possible. Because of multi-arch container manifests it is a two step operation.

Example call:

Use the external `crane` tool to extract the x86_64 container part, the `cosign` tool to retrieve and verify the attestation, and the `jq` tool to extract the SPDX 2.0 SBOM data.

```
crane digest --platform linux/amd64 registry.suse.com/suse/sle15:15.4
```

Example result:

```
sha256:c8aeb5a7662c38716d303fb854c5baa2329afccb4637c0f3c7c44b971181fdbb
```

Then run the following on the command line:

```
 cosign verify-attestation --type spdxjson –key
 /usr/share/pki/containers/suse-container-key.pem
 registry.suse.com/suse/
sle15@sha256:c8aeb5a7662c38716d303fb854c5baa2329afccb4637c0f3c7c44b971181fdbb | jq
 '.payload | @base64d | fromjson
 | .predicate'
```

7.3 Security information in CSAF format

The Common Security Advisory Format (CSAF) is an industry-standard format for publishing security advisories in machine readable form. It is the descendant of the CVRF format and standardized by the OASIS foundation.

It differs from the OVAL format, whose goal is to be able to machine-check the health of a system for security, while the CVRF and CSAF formats are intended for machine import into ticketing systems and bug trackers to respond to vulnerabilities.

SUSE currently offers:

- CSAF data indexed by Security Advisory in CSAF 2.0 format.
- CSAF VEX data indexed by CVE in CSAF 2.0 VEX format.

SUSE has started generating CSAF data for SUSE security update notifications and CVEs in February 2023, including all previous security advisories and CVEs.

The CSAF 2.0 security advisory data can be downloaded from this SUSE ftp site (https://ftp.suse.com/pub/projects/security/csaf/) ↗.

The CSAF 2.0 VEX data indexed by CVE can be downloaded from this SUSE ftp site (https://ftp.suse.com/pub/projects/security/csaf-vex/) ↗.

The data is available under the Creative Commons license, with attribution CC-BY-4.0. The CSAF format is a verbose and simple JSON format, so it can be hooked into other tools pretty easily even without additional libraries.

A list of reference tools (https://oasis-open.github.io/csaf-documentation/tools.html) ↗ is also provided by OASIS.

7.4 Open Source Vulnerabilities (OSV) data

OSV is a Google-based vulnerability database and triage infrastructure for open source projects aimed at helping both open source maintainers and consumers of open source.

SUSE is publishing regularly updated OSV data, which is available in beta status at https://ftp.suse.com/pub/projects/security/osv/ ↗.

8 Securing our product portfolio

January 17th 2023

SUSE Linux Enterprise Micro 5.2 (SLE Micro) has received the PSA Certified Security Assurance Certificate (PSA Certified Level 1) on ARM. PSA (Platform Security Architecture) Certified is a security certification scheme for Internet of Things (IoT) hardware, software and devices. You can view our certification at https://www.psacertified.org/products/suse-linux-enterprise-micro/ ↗.

April 26th 2023

IST has validated 17 new algorithms for SUSE Linux Enterprise Server (SLES) 15 SP4. You can view these certifications at https://csrc.nist.gov/projects/cryptographic-algorithm-validation-program/validation-search? ↗ when you search by **Implementation** and **Vendor Name SUSE**.

June 5th 2023

The University of New Hampshire Interoperability Laboratory (UNH-IOL) has certified SUSE Linux Enterprise Server (SLES) 15 SP4 under the USGv6 R1 technical requirements. UNH-IOL provides testing and evaluation services of IPv6 technologies and certifies compliance.

August 23rd 2023

The University of New Hampshire Interoperability Laboratory (UNH-IOL) has certified SUSE Linux Enterprise Micro (SLE Micro) 5.3 under the IPv6 Ready Logo Program.

September 29th 2023

The University of New Hampshire Interoperability Laboratory (UNH-IOL) has certified SUSE Linux Enterprise Micro (SLE Micro) 5.4 under the USGv6 R1 technical requirements. UNH-IOL provides testing and evaluation services of IPv6 technologies and certifies compliance.

November 1st 2023

SUSE Rancher Hosted completed its annual SOC2 audit and achieved SOC2 Type 2 Compliance. SUSE also achieved the AICPA System and Organization Controls (SOC2) Type 1 certification. The audits were conducted by Armanino LLP, one of the largest independent accounting and business consulting firms in the United States. SUSE is proud to provide yet another level of compliance to its clients.

December 15th 2023

SUSE Linux Enterprise Server 15 SP4 is now Common Criteria certified with the BSI scheme. This guarantees that our operating system meets all the requirements of the NIAP Protection Profile General Purpose Operating System along with Functional Package for Secure Shell (SSH).

9 About SUSE

SUSE is a global leader in innovative, reliable and secure enterprise open source solutions, including SUSE Linux Enterprise, Rancher and NeuVector. More than 60% of the Fortune 500 rely on SUSE to power their mission-critical workloads, enabling them to innovate everywhere – from the data center to the cloud, to the edge and beyond. SUSE puts the “open” back in open source, collaborating with partners and communities to give customers the agility to tackle innovation challenges today and the freedom to evolve their strategy and solutions tomorrow. For more information, visit https://www.suse.com ↗.

10 Forward-looking statements

Any statements in this document about future expectations, plans and prospects for the company, including statements containing the words “aims”, “targets”, “will”, “believes”, “anticipates”, “plans”“, expects”, and similar expressions, may constitute forward-looking statements and should be read with caution.

Actual results may differ materially from those indicated by such forward-looking statements as a result of various important factors, including competitive landscape, development of customer deals, reliance upon customer relationships, management of growth and acquisitions, the possibility of undetected software issues, the risks of impacts of the COVID-19 pandemic and economic downturns, pricing pressures and the viability of the Internet. In addition, any forward-looking statements included herein represent views as of the date of this document and these views could change. The Company does not have any obligation to update its forward-looking statements. These forward-looking statements are subject to change and should not be relied upon as representing the Company's views as of any date other than the publication date of this document.

11 Legal notice

Documents published as part of the **SUSE Best Practices** series have been contributed voluntarily by SUSE employees and third parties. They are meant to serve as examples of how particular actions can be performed. They have been compiled with utmost attention to detail. However, this does not guarantee complete accuracy. SUSE cannot verify that actions described in these documents do what is claimed or whether actions described have unintended consequences. SUSE LLC, its affiliates, the authors, and the translators may not be held liable for possible errors or the consequences thereof.

Below we draw your attention to the license under which the articles are published.

GNU Free Documentation License

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

```
Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.2
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled "GNU
 Free Documentation License".
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts". line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.