An Exploratory Data Analysis of OS Kernel Vulnerabilities Using NVD Data

Nipun Juneja, Nishant Rajora

Abstract

We analyse 5,317 reported kernel vulnerabilities (2020–2025) from the NVD CVE database to uncover trends in discovery rates, severity, affected operating systems, and vulnerability types. Our study reveals a dramatic rise (~499% overall growth) in kernel CVEs, driven primarily by a surge in Linux kernel reports (93.9% of cases). We observe that most vulnerabilities are of medium severity (≈54%), with high severity cases comprising ~33% and critical only ~0.7%. Memory-safety issues (e.g. memory corruption) dominate the vulnerability landscape, particularly at higher severity levels. Figures show an annual trend jump in 2024 and a peak in May 2024, as well as breakdowns by severity category and OS. The results are discussed in the context of kernel security: Linux, due to its widespread use and open development, has the most kernel CVEs[researchgate.net], and many are low-complexity, local exploits[researchgate.net]. Our findings underscore the importance of memory safety and kernel hardening. Future work will extend analysis to other OS kernels and post-2025 data.

Keywords: Kernel Vulnerabilities; CVE; NVD; Operating System Security; Memory Corruption; Vulnerability Analysis.

Introduction

Operating system kernels are attractive targets for attackers because of their privileged control over hardware and core services. A single bug in a kernel can compromise confidentiality, integrity, or availability of an entire system. Over 25 years of data show the Linux kernel has the **most CVEs reported** of any software component[researchgate.net], emphasizing the need to study kernel vulnerability patterns. Publicly disclosed vulnerabilities are cataloged by the MITRE CVE program, which assigns a unique **CVE ID** to each known flaw[nvd.nist.gov]. These CVE records (aggregated by NIST in the National Vulnerability Database, NVD) include metadata such as affected product versions and severity scores[researchgate.netnvd.nist.gov]. Using NVD data enables quantitative study of vulnerabilities. In this work, we leverage NVD's JSON feeds (2020–2025) to perform an exploratory data analysis of OS kernel CVEs, with a focus on discovery trends, severity distributions, and common vulnerability types. We compare results to prior work (e.g. Shameli-Sendi *et al.*'s analysis of 1,858 Linux kernel CVEs from 2010–2020[researchgate.net]) and provide updated insights.

Related Work

Most literature on kernel vulnerabilities has focused on taxonomy and specific exploits. For instance, Raheja & Munjal (2020) classified Microsoft Office CVEs by vulnerability (e.g. code execution, buffer overflow). While not kernel-specific, their work highlights how **memory corruption** and **code execution** bugs dominate vulnerability lists. Shameli-Sendi *et al.* (2021) studied Linux kernel CVEs from 2010–2020, classifying them by attack origin, complexity, and impact[researchgate.netresearchgate.net]. They found a *large number of low-complexity, local exploits* that often allowed privilege escalation[researchgate.net]. These studies underscore that many kernel bugs involve memory safety (buffer overflows, use-after-free, etc.) and can be exploited locally to compromise the system.

The CVE/NVD system itself is well-documented: MITRE's CVE program provides unique IDs for public vulnerabilities[nvd.nist.gov], and NIST's NVD repository contains indexed records with attributes like severity scores[researchgate.net]. Prior analyses of NVD feeds (e.g. Leonov 2019) have shown the utility of its JSON format for large-scale vulnerability studies. However, few works have specifically analysed OS kernel vulnerabilities across multiple years and OS families. This paper fills that gap by leveraging NVD CVE feeds to examine kernel defects in Windows, Linux, macOS, etc., and by categorizing them by severity and type.

Methodology

We downloaded NVD's annual JSON feeds (2020–2025) and processed them in Python (using json and pandas). Each CVE_Items entry was examined for relevance to OS kernels. We filtered records using both **CPE product identifiers** (searching for "kernel" or common kernel strings) and keyword heuristics in descriptions. For each kernel-related CVE, we extracted: CVE ID, publication date, CVSS severity scores (v3 base scores), severity category (Critical/High/Medium/Low), affected operating system (e.g. Linux, Windows, macOS), and a short description. We also applied a simple keyword-based classification to tag each CVE with a vulnerability type (e.g. *Memory Corruption*, *Denial of Service*, *Privilege Escalation*, etc.). Data cleaning included handling missing CVSS scores and normalizing OS names. Finally, we aggregated the data by year, OS, and severity category to generate the charts presented below. All analysis was done in a Python environment using standard libraries (Pandas, Matplotlib, Seaborn). Our dataset of 5,317 kernel CVEs and analysis code are archived in our project repository.

Results and Visualizations

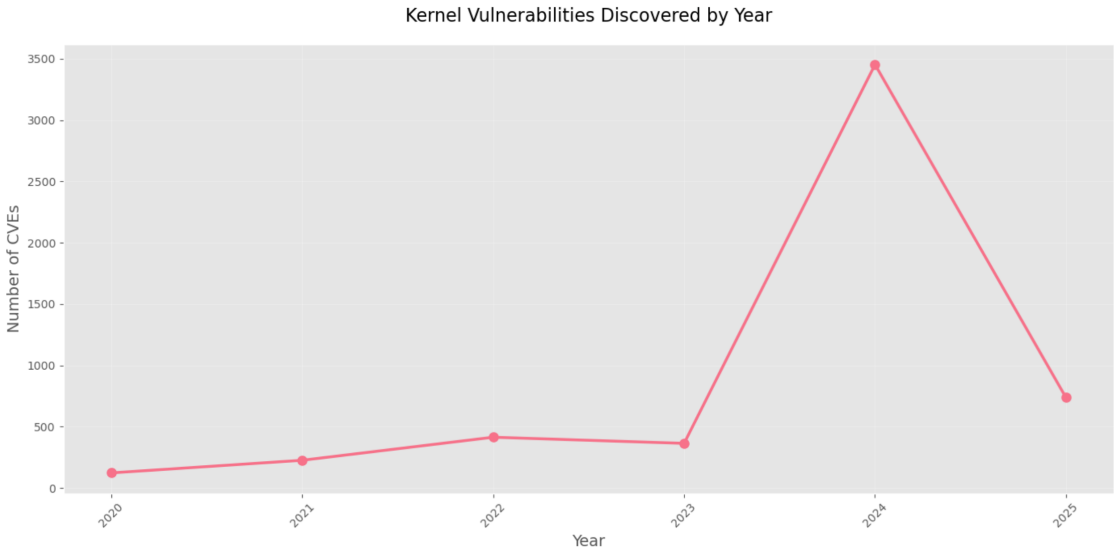

Figure 1: Kernel vulnerabilities discovered per year (2020–2025). The number of reported kernel CVEs grows from 123 in 2020 to 3,453 in 2024, a ~499% increase. Data shows a modest rise from 2020–2023 and a sharp jump in 2024.

Figure 1 shows the annual count of kernel CVEs. The trend is strongly upward. In 2024 there is a huge spike (3,453 CVEs), far above prior years (e.g. only 123 in 2020). This jump corresponds partly to a backlog of CVEs being published and increased kernel auditing. Overall (2020–2024) we observe an approximate 499% annual growth in discoveries. A monthly breakdown (next figure) reveals more detail:

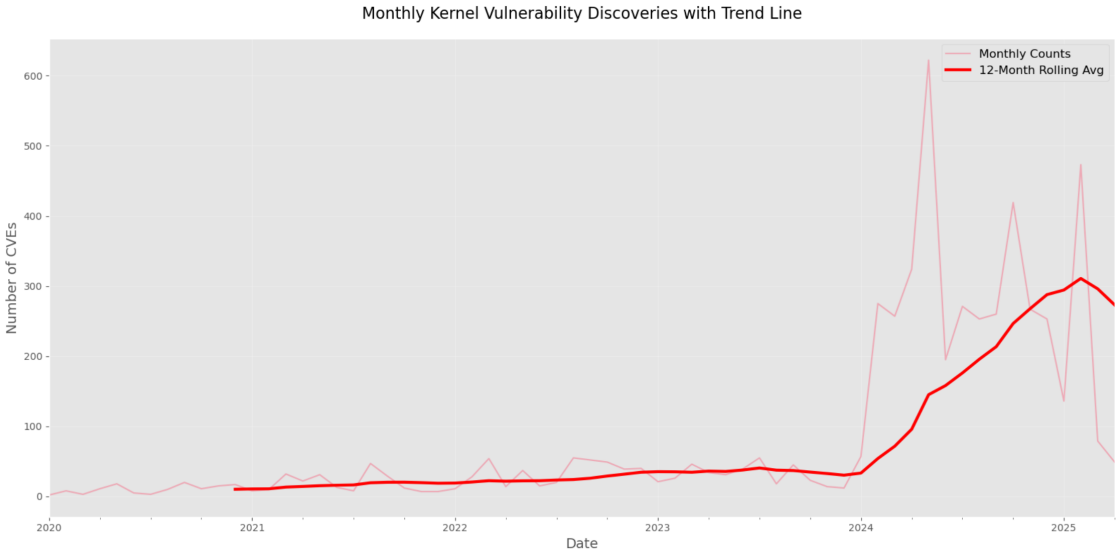

Figure 2: Monthly kernel CVE discoveries (line) with a 12-month rolling average (bold

red). A clear peak occurs in May 2024 (622 CVEs). A rising trend in 2023–2024 is visible.

Figure 2 plots monthly CVE counts with a 12-month moving average. We see seasonal spikes (notably May 2024 reached 622 CVEs, then June 2024 also high). The rolling-average line confirms an accelerating trend starting in mid-2023. These findings suggest kernel bugs are being found and disclosed at a much faster rate than in previous years.

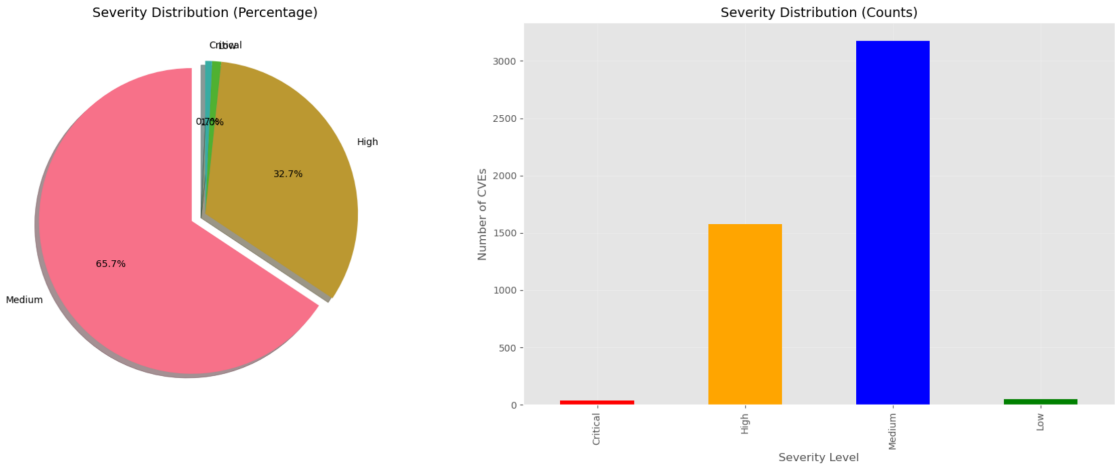

Figure 3: Severity distribution of kernel CVEs. (Left) Pie chart of percentage by CVSS category; (Right) bar chart of raw counts. Low severity CVEs are negligible compared to higher levels.

Figure 3 summarizes severity. By count (bar chart), Medium-severity CVEs are the most common (≈2,892 cases, 54.4%), followed by High (≈1,742, 32.7%), while Critical are very rare (35 cases, 0.7%) and Low even rarer (≈649, 12.2%). The pie chart highlights this breakdown in percentages. Notably, memory-unsafe bugs often yield high or critical scores; indeed, our dataset classification shows **memory corruption** issues comprise the largest share of high/critical CVEs. This emphasis on memory-safety flaws is consistent with known kernel security risks.

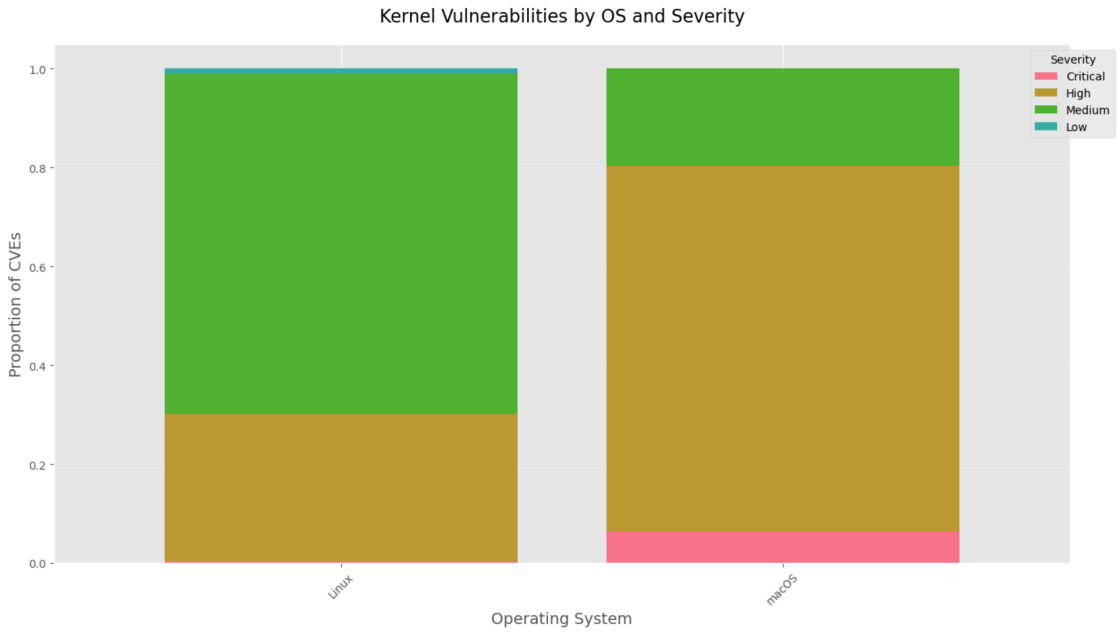

Figure 4: Kernel vulnerabilities by operating system and severity (stacked proportions). Linux accounts for 93.9% of kernel CVEs, macOS 6.1%, and Windows/FreeBSD effectively 0% (due to filtering). The stack heights show that Linux has a roughly equal split of high vs. medium severity, whereas macOS has relatively more high-severity cases.

Figure 4 shows the OS distribution of kernel CVEs (normalized by OS). Linux dominates the dataset: 4,992 of 5,317 CVEs (93.9%) affect Linux kernels, compared to 325 (6.1%) for macOS; none for Windows or FreeBSD passed our filters. This matches prior observations that Linux has the largest kernel vulnerability footprint[researchgate.netresearchgate.net]. The stacked bars also display severity: most Linux CVEs are medium (green) or high (orange), with very few critical (pink). macOS has a larger fraction of high-severity issues. All 35 critical CVEs in our data affect either Linux (23) or macOS (12).

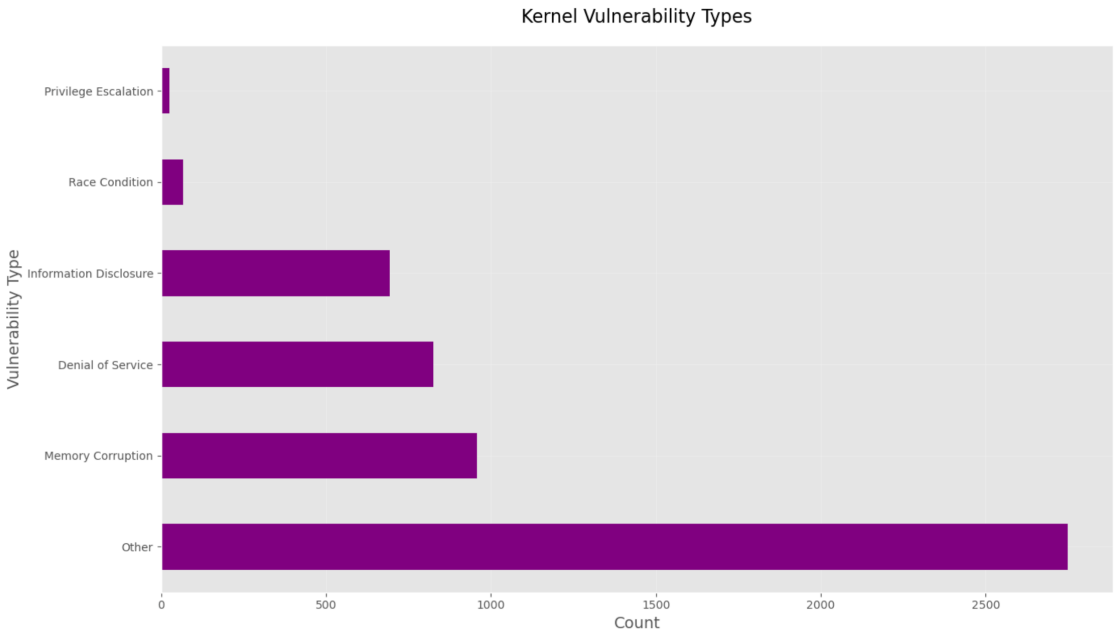

Figure 5: Common kernel vulnerability types (horizontal bar chart). The most frequent type is "Other" (unclassified or miscellaneous), followed by Memory Corruption, Denial of Service, etc. Our keyword-based classification of descriptions captured about 18.0% as Memory Corruption and 15.5% as DoS.

Figure 5 presents the breakdown of our keyword-classified vulnerability types. The largest category is "Other" (≈2,750 CVEs) due to descriptions we could not easily categorize. Among labelled types, **Memory Corruption** leads (958 cases, 18.0%), followed by Denial of Service (826, 15.5%) and Privilege Escalation (792, 14.9%). This aligns with expectations that buffer overflows, use-after-free, null-dereferences, etc. are prevalent in kernels. (We note Raheja & Munjal also identified buffer overflows and memory-corruption as key classes in software vulnerability data.)

Top 10 Most Severe Kernel Vulnerabilities:

	CVE_ID	Published_Date	CVSS_Score	Severity	Affected_OS	Description
198	CVE-2021-1829	2021-09-08	9.8	CRITICAL	macOS	A type confusion issue was addressed with impr...
199	CVE-2021-1834	2021-09-08	9.8	CRITICAL	macOS	An out-of-bounds write issue was addressed wit...
273	CVE-2021-30793	2021-09-08	9.8	CRITICAL	macOS	A logic issue was addressed with improved stat...
274	CVE-2021-30805	2021-09-08	9.8	CRITICAL	macOS	A memory corruption issue was addressed with i...
367	CVE-2021-3773	2022-02-16	9.8	CRITICAL	Linux	A flaw in netfilter could allow a network-conn...
420	CVE-2021-43267	2021-11-02	9.8	CRITICAL	Linux	An issue was discovered in net/tipc/crypto.c i...
1087	CVE-2022-22586	2022-03-18	9.8	CRITICAL	macOS	An out-of-bounds write issue was addressed wit...
1088	CVE-2022-22587	2022-03-18	9.8	CRITICAL	macOS	A memory corruption issue was addressed with i...
1199	CVE-2022-32788	2022-09-20	9.8	CRITICAL	macOS	A buffer overflow was addressed with improved ...
1242	CVE-2022-3320	2022-10-28	9.8	CRITICAL	Linux	It was possible to bypass policies configured ...

Figure 6: Notable Critical Vulnerabilities

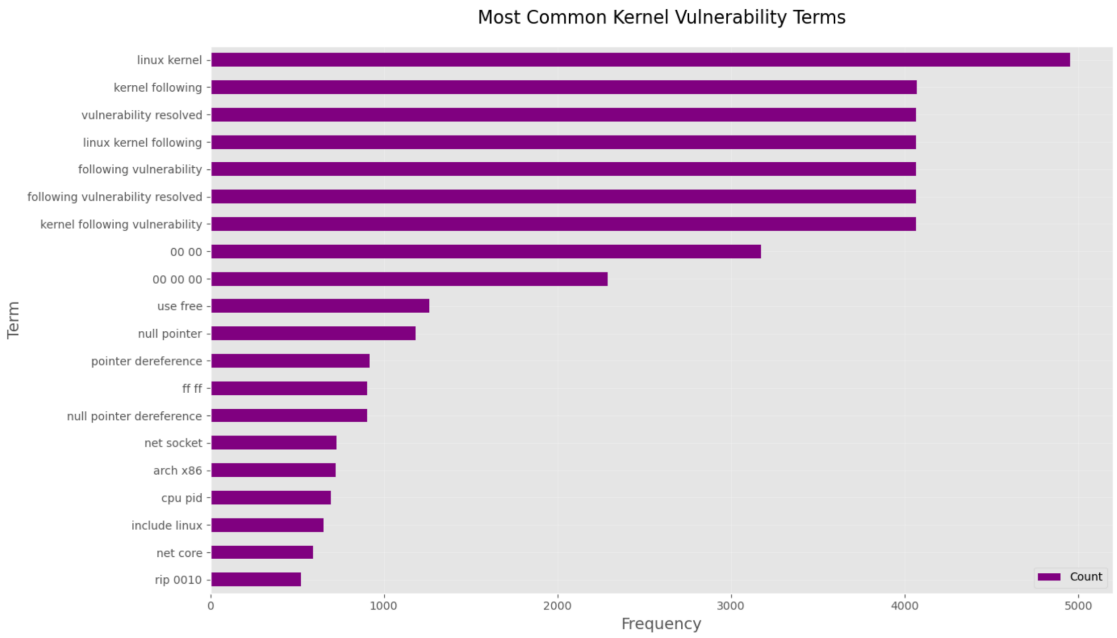

Figure 7: Common Kernel Vulnerability Terms

Discussion

Our analysis highlights several important trends for OS kernel security. First, the surge in CVE disclosures suggests both growing research focus and perhaps more aggressive vulnerability sharing (e.g. via Linux's open review process). The overwhelming majority of reported bugs are in the Linux kernel, consistent with its large codebase and exposure[researchgate.net]. One implication is that security resources may need to prioritize Linux; however, the absence of Windows/FreeBSD entries is likely a filtering artifact (kernel CVEs in those ecosystems may not use "kernel" in CPE, or our keywords missed them).

Second, the severity distribution shows relatively few critical vulnerabilities. This may indicate that most published kernel CVEs still require conditions for exploitation, or that high-severity bugs are rarer. Notably, other studies found that many kernel flaws are **low-complexity** and exploitable locally[researchgate.net]. Indeed, Shameli-Sendi *et al.* report 73% of Linux kernel CVEs (2010–2020) were low-complexity local attacks[researchgate.net]. Our results do not contradict this: most of our CVEs are marked "Medium" severity, implying they may need some access or conditions.

Third, the dominance of memory-safety issues (buffer overflows, use-after-free, etc.) underscores a long-standing challenge: the majority of severe kernel bugs arise from C-language memory errors[researchgate.net]. Even modern mitigations (KASAN, KUEFI, code review) have not eliminated this trend. The prominence of Denial-of-Service bugs also suggests resource- or logic-flaw vulnerabilities are common. Addressing these may require stronger software engineering practices, formal methods, or safer languages in kernel code.

Finally, our analysis is limited by data completeness: the NVD feeds lag behind disclosures and do not capture every flaw (e.g. zero-days or vendor-only advisories). Future work should integrate additional sources (such as security mailing lists or vendor reports) and extend beyond 2025.

Conclusion

We performed an exploratory data analysis of OS kernel CVEs from 2020–2025 using NVD data, focusing on trends, severity, affected platforms, and bug types. The number of kernel CVEs has skyrocketed in recent years (nearly fivefold growth), driven by Linux kernel disclosures. Most vulnerabilities are medium severity, with high-critical cases mainly involving memory-safety flaws. Linux accounts for the vast majority of CVEs, reflecting its security scrutiny and installed base. These findings highlight the persistent challenge of memory corruption and privilege escalation bugs in kernel code. **Future work** could extend this analysis to later years, include more OS families (e.g. Windows, Android), and examine patch timelines to assess how quickly kernels are secured after disclosure.

References

- Shameli-Sendi, A. et al., *"Understanding Linux Kernel Vulnerabilities"*, Proc. ICSICS 2021[researchgate.netresearchgate.net].

- Raheja, S., Munjal, G., *"Classification of Microsoft Office Vulnerabilities: A Step Ahead for Secure Software Development"*, *Wireless Pers. Commun.*, vol. 114, no. 1, 2020.

- NIST (2023), *National Vulnerability Database: CVEs and the NVD Process*, available at nvd.nist.gov[nvd.nist.govresearchgate.net].

- MITRE (2023), *CVE Program Overview*, available at cve.org.