Journal of Intelligent Systems and Internet of Things

♦ ISSN: 2690-6791 Vol: 19, No. 01, 2027 ♦ PP. 47–66

♦ Article DOI: https://doi.org/10.54216/JISIoT.190104



FULL LENGTH ARTICLE

IoT Device Vulnerabilities in the National Vulnerability Database: Latent Attack Profiles, Text-Based Recovery of Unenriched Records and Exploitation Hazards

Fadi Farha[1,*]

[1]The Faculty of Informatics Engineering, Aleppo University, Syria

Email: fadi_farha@alepuniv.edu.sy

Received: August 30, 2026 Revised: October 01, 2026 Accepted: November 07, 2026 ⋆ Corresponding author

ABSTRACT

Internet of Things (IoT) devices enter the vulnerability record through firmware advisories that are scored, catalogued and enriched by institutions under strain. We separate device CVEs from the rest of the National Vulnerability Database using product enumerations and characterise those published from January 2022 to June 2026 with five models. They are more severe. A quarter are Critical against 13.7% of other CVEs, memory-safety and command-injection flaws each carry an odds ratio above three, and a single router vendor accounts for one device CVE in eight. An expectation–maximisation latent class model finds eight attack profiles. A logistic text classifier trained on enriched records (AUC 0.98), combined with adjusted classify-and-count, recovers device CVEs that NIST never enriched and raises the 2025 count by about a third, although the decline in the device share survives the correction. Topic models are dominated by reporting templates. They also record a rise in end-of-life language, from 0.1% to 4.8% of device CVEs. With severity and weakness held fixed, device CVEs are less likely to reach the Known Exploited Vulnerabilities catalog (hazard ratio 0.61), which suggests that their high scores overstate observed exploitation.

Keywords: Internet of Things ▪ Firmware vulnerabilities ▪ Latent class analysis ▪ Text classification ▪ Survival analysis

1. INTRODUCTION

Routers, IP cameras, programmable logic controllers and the cellular modems inside phones and gateways run firmware that their owners rarely see and seldom update. That firmware is a large attack surface. The Mirai botnet assembled hundreds of thousands of such devices from default credentials and old flaws [1], measurements of home networks find long tails of vendors with weak update practices [2], and systematic reviews of IoT security keep returning to the same classes of firmware bug [3, 4, 5]. Regulation has followed. The European Union's Cyber Resilience Act now obliges manufacturers of products with digital elements to handle and report vulnerabilities for the whole support period [6], and ETSI EN 303 645 and NIST IR 8259 set baseline expectations for consumer and enterprise IoT devices [7, 8].

All of these instruments rely on a public record of device vulnerabilities, and the record is the CVE list as curated by the National Vulnerability Database (NVD). NVD adds what a CVE Numbering Authority (CNA) does not supply: product enumerations in the Common Platform Enumeration (CPE) syntax, which tell a reader that a CVE concerns a router firmware image and not a web application, and an independent severity score. That service has frayed. A backlog of unprocessed CVEs began in February 2024 [9], and in April 2026 NIST announced that NVD would enrich KEV entries within one business day while leaving most other records

© 2027 The Author(s). Published by ASPG. This article is licensed under CC BY 4.0.

without its own analysis [10]. A device CVE without CPE data is invisible to any analysis that filters by product type.

For IoT security the gap has a direct cost. Product type is the natural unit of analysis there, yet few studies have separated device vulnerabilities from the rest of the CVE stream at scale, and none that we know of has measured how many of them the enrichment backlog hides. Device-side studies examine firmware images and deployed devices [11, 12], while record-side studies treat the CVE list as one population [13, 14].

We address this gap with a five-stage pipeline. It runs on every CVE that NVD published from January 2022 to June 2026, and its first stage, a deterministic rule on CPE enumerations, labels device CVEs and assigns them to six categories (Algorithm 1). An expectation–maximisation (EM) latent class model then groups their CVSS vectors into attack profiles (Algorithm 2). Records that NVD never enriched need a different route. A text classifier trained on enriched records, combined with adjusted classify-and-count quantification, estimates how many device CVEs they contain (Algorithm 3). Non-negative matrix factorisation separates reporting templates from content topics (Algorithm 4), a Cox model relates device status to the time until exploitation is confirmed, and a beta-binomial empirical Bayes model ranks vendors by their share of Critical CVEs without rewarding small samples (Algorithm 5).

The paper makes four contributions:

- A reproducible labelling procedure and taxonomy for IoT and embedded device CVEs in NVD, with a comparison of their severity, attack vectors, weakness types and vendor concentration against all other CVEs.
- A latent class model of device CVSS vectors with eight attack profiles, each tied to its exploitation rate.
- A text-based quantification method, validated on later data, that recovers device CVEs missing from product-based queries and corrects yearly counts for the enrichment backlog, which in 2025 hid about a quarter of all device CVEs from such queries.
- Evidence from topic models and survival analysis that CNA templates shape the textual record, that end-of-life products form a growing share of device disclosures, and that device CVEs reach the KEV catalog less often than their severity predicts.

All results come from fixed public snapshots. Section 2 reviews related work. Section 3 describes the data and the labelling rule, Section 4 presents the models and Section 5 reports results, while Section 6 discusses what they mean for manufacturers, regulators and the operators of device fleets, along with the study's limitations, and Section 7 concludes.

2. RELATED WORK

2.1 Firmware and IoT device security

Large-scale firmware studies began with Costin et al. [11], who crawled 32,356 firmware images and found hard-coded credentials, backdoors and outdated components across vendors. Dynamic analysis came next. Chen et al. [12] emulated Linux-based firmware with FIRMADYNE and confirmed known exploits on a sizeable fraction of images. Feng et al. [15] built a rule engine that identifies IoT devices from application-layer banners, and Kumar et al. [2] measured 83 million devices in 16 million homes and found that a handful of vendors account for most devices in each category. Surveys organise the resulting knowledge by attack surface and device layer [3, 4, 16], and two recent reviews concentrate on firmware itself, one on analysis techniques [17] and one on vulnerability classes and auditing [5]. Measurements of gateway firmware report that memory-related weaknesses make up most of the vulnerability burden [18]. What these studies share is a device-side view. They start from images or deployments, so they measure what is present on devices, not what is disclosed about them.

2.2 The vulnerability record and its enrichment

The disclosure side has its own literature. Anwar et al. [13] documented missing and inconsistent fields across NVD. The Department of Commerce Inspector General went further in May 2026, reporting that independent evaluators matched NVD's scoring exactly for only 8 of 69 sampled vulnerabilities [9]. CISA's Known Exploited Vulnerabilities (KEV) catalog provides an exploitation outcome against which severity can be judged [19], and CVSS v3.1 defines the vector that both CNAs and NVD use to score [20]. Data-driven vulnerability assessment has turned these records into prediction targets, reviewed by Le et al. [14], but device CVEs have rarely been separated from the rest; analyses of the full CVE stream mix router firmware with WordPress plug-ins and cloud services.

2.3 Methods for mixed categorical data, text and counts

Four bodies of method are combined here. Latent class analysis models categorical response patterns as a finite mixture [21, 22], fitted by the EM algorithm [23] and compared across numbers of classes with the Bayesian information criterion [24, 25]. Text classification uses term frequency–inverse document frequency weights [26] with regularised logistic regression, support vector machines with Platt scaling [27] and naive Bayes, as implemented in scikit-learn [28]. Estimating the share of a class in an unlabelled set is a problem of its own, called quantification [29, 30]. Topics come from non-negative matrix factorisation [31, 32]. Coherence is scored with normalised pointwise mutual information [33, 34]. Survival analysis uses the Kaplan–Meier estimator and the Cox model [35, 36], with lifelines [37], and the empirical Bayes shrinkage of binomial proportions uses the beta-binomial model [38, 39] with Wilson intervals for single proportions [40].

3. DATA AND DEVICE LABELLING

3.1 Snapshot

NVD records come from the fkie-cad/nvd-json-data-feeds mirror at tag v2026.07.31-000014 (commit 734f9e2, 31 July 2026), which stores each CVE as published through the NVD application programming interface, and we parsed every record in the CVE-2022 to CVE-2026 folders and kept non-rejected CVEs published from 1 January 2022 to 30 June 2026: 167,579 CVEs. For each we extracted publication date, analysis status, CNA identifier, CVSS v3.x vector and base score (NVD's own Primary assessment where present and the CNA's Secondary assessment otherwise), CWE identifiers, reference tags, the English description and every CPE match

in the configurations, with its part (application, operating system or hardware), vendor, product and vulnerable flag. Exploitation status comes from the CISA KEV catalog at version 2026.07.29 [19], taken from commit 564b8c5 of the cisagov/kev-data repository. Table 1 lists the sources.

Enrichment is the first complication. A CVE is enriched if NVD has attached at least one CPE configuration; 129,500 of the 167,579 (77.3%) were, but the share fell from 100% for CVEs published in 2022 and 2023 to 80.2% in 2024 and 62.8% in 2025. Device labels from CPE data therefore exist only for enriched CVEs, and Section 4.3 deals with the rest.

3.2 Device labelling and taxonomy

NVD's CPE dictionary encodes device firmware in two ways. Hardware entries (part "h") describe the physical product, and operating-system entries (part "o") whose product name ends in "_firmware" describe its software. For enriched CVE i with CPE set C_i of (part, vendor, product) triples, the device indicator is

$$\delta_i = 1\Big[\exists (p,v,\pi) \in C_i : \; p = h \vee (p = o \wedge \pi \text{ ends in _firmware}) \; \wedge v \notin V_{\text{ex}}\Big] \tag{1}$$

where V_{ex} excludes 25 vendors whose hardware and firmware entries describe personal computers, servers, processors and their BIOS (Intel, AMD, Dell, Lenovo, HP and similar). The exclusion is a judgement. It removes server baseboard controllers and CPU microcode, which are embedded but are not what IoT policy addresses, and it keeps chipset vendors such as Qualcomm, MediaTek and Unisoc, whose modem and baseband firmware runs inside phones, gateways and cellular IoT modules.

Device CVEs are assigned to six categories by vendor lists and description keywords (Algorithm 1): network devices (routers, access points, switches, firewalls and cellular gateways), chipsets and modems, industrial control (programmable logic controllers, human–machine interfaces, industrial gateways), cameras and surveillance equipment, printers, and other embedded or consumer devices. Weakness families come from CWE identifiers in ten groups (memory safety; command or code injection; authentication and access control; cross-site scripting; SQL injection; path traversal, upload and request forgery; information exposure and cryptography; input validation and denial of service; other; unspecified).

Algorithm 1: Device labelling and taxonomy

Require: enriched CVEs with CPE sets C_i and descriptions d_i; excluded vendors V_{ex}; vendor lists $V_{\text{chip}}, V_{\text{ics}}, V_{\text{net}}$

Ensure: device indicator δ_i, category c_i, primary vendor v_i

1: for all enriched CVEs i do
2: $D_i \leftarrow \{v : (p,v,\pi) \in C_i, \; p = h \vee (p = o \wedge \pi \text{ ends in _firmware}), \; v \notin V_{\text{ex}}\}$
3: $\delta_i \leftarrow 1\,[D_i \neq \varnothing]$ ▷ Eq. (1)
4: if $\delta_i = 1$ then
5: $v_i \leftarrow$ alphabetically first element of D_i
6: if $D_i \cap V_{\text{chip}} \neq \varnothing$ then $c_i \leftarrow$ chipset / modem
7: else if $D_i \cap V_{\text{ics}} \neq \varnothing$ or d_i mentions PLC, SCADA, HMI or industrial then $c_i \leftarrow$ industrial control
8: else if d_i mentions camera, NVR or DVR then $c_i \leftarrow$ camera / surveillance
9: else if d_i mentions printer or a printer vendor is in D_i then $c_i \leftarrow$ printer
10: else if $D_i \cap V_{\text{net}} \neq \varnothing$ or d_i mentions router, gateway, switch, firewall or modem then $c_i \leftarrow$ network device
11: else $c_i \leftarrow$ other embedded / consumer
12: end if
13: end for
14: return δ_i, c_i, v_i

The rule labels 14,458 enriched CVEs (11.2%) as device CVEs. Table 2 gives the taxonomy. Network devices dominate with 52.6% of device CVEs, chipsets and modems follow with 20.1%, and cameras and printers form small categories, with 127 and 100 CVEs.

4. METHODS

4.1 Comparisons and trend

Device and non-device CVEs are compared on proportions with Wilson score intervals [40],

$$\frac{\hat{p} + \frac{z^2}{2n} \pm z\sqrt{\frac{\hat{p}(1-\hat{p})}{n} + \frac{z^2}{4n^2}}}{1 + \frac{z^2}{n}}, \qquad z = 1.96, \tag{2}$$

and on weakness families with odds ratios from the 2×2 table of device status by family membership, with the Haldane correction,

$$\hat{OR} = \frac{\left(a + \frac{1}{2}\right)\left(d + \frac{1}{2}\right)}{\left(b + \frac{1}{2}\right)\left(c + \frac{1}{2}\right)}, \qquad \mathrm{SE}(\log \hat{OR}) = \sqrt{\frac{1}{a+1/2} + \frac{1}{b+1/2} + \frac{1}{c+1/2} + \frac{1}{d+1/2}} \tag{3}$$

The trend in the device share uses a negative binomial regression of monthly device counts Y_t on time τ_t in years, with the log of all enriched CVEs that month, N_t, as offset,

$$\log \mathbb{E}[Y_t] = \log N_t + \beta_0 + \beta_1 \tau_t, \qquad \mathrm{Var}(Y_t) = \mu_t + \alpha \mu_t^2, \tag{4}$$

so that $\exp(\beta_1)$ is the yearly multiplicative change in the device share; α is chosen by profile likelihood. Vendor concentration is summarised by the Gini coefficient of device CVE counts over vendors,

$$G = \frac{\sum_{i=1}^{V} (2i - V - 1)\, x_{(i)}}{V \sum_{i=1}^{V} x_{(i)}}, \tag{5}$$

with $x_{(i)}$ the counts in ascending order.

Table 1. Data sources and snapshot identifiers

Source	Snapshot	Fields used	Role
NVD JSON mirror (fkie-cad)	Tag v2026.07.31-000014	Publication date, status, CNA, CVSS v3.x, CWE, CPE configurations, description, reference tags	All analyses
CISA KEV catalog	Version 2026.07.29	CVE identifier, date added, ransomware field	Exploitation outcome

Table 2. Device categories, enriched CVEs published January 2022 – June 2026

Category	CVEs	Critical (%)	Network vector (%)	Memory safety (%)	Command injection (%)	Reported by VulDB (%)	In KEV
Network device	7,608	30.9	81.7	49.4	24.3	25.4	56
Chipset / modem	2,913	4.9	20.9	61.4	0.6	0.0	10
Other embedded / consumer	2,585	29.2	73.8	11.6	18.5	7.9	15
Industrial control	1,125	23.1	84.5	8.9	14.6	0.3	12
Camera / surveillance	127	19.7	71.7	11.8	21.3	1.6	3
Printer	100	48.0	86.0	38.0	5.0	0.0	0
All device CVEs	14,458	24.8	68.2	41.5	17.6	14.8	96

4.2 Latent attack profiles

Each device CVE with a CVSS v3.x vector supplies eight categorical items x_{ij}, $j = 1,\dots,8$ (attack vector, attack complexity, privileges required, user interaction, scope, and confidentiality, integrity and availability impact), with R_j levels each. The latent class model [21, 22] treats each vector as drawn from one of K unobserved classes, within which items are independent:

$$\Pr(x_i) = \sum_{k=1}^{K} \pi_k \prod_{j=1}^{8} \prod_{r=1}^{R_j} \theta_{jkr}^{1[x_{ij}=r]}. \tag{6}$$

The EM algorithm [23] alternates the posterior class probabilities

$$\tau_{ik} = \frac{\pi_k \prod_j \theta_{jkx_{ij}}}{\sum_l \pi_l \prod_j \theta_{jlx_{ij}}} \tag{7}$$

with the closed-form updates

$$\hat{\pi}_k = \frac{1}{n}\sum_i \tau_{ik}, \qquad \hat{\theta}_{jkr} = \frac{\sum_i \tau_{ik}\,1[x_{ij}=r] + \varepsilon}{\sum_i \tau_{ik} + R_j\varepsilon}, \tag{8}$$

where $\varepsilon = 10^{-6}$ keeps probabilities off the boundary. Twelve random Dirichlet starts are run for each K from 1 to 8 (eight starts for $K = 9, 10$), the best log-likelihood is kept, and models are compared with

$$\mathrm{BIC}_K = -2\log \hat{L}_K + \left[(K-1) + K\sum_j (R_j - 1)\right]\log n, \tag{9}$$

and with the relative entropy

$$E_K = 1 + \frac{\sum_i \sum_k \tau_{ik}\log \tau_{ik}}{n \log K}, \tag{10}$$

which approaches one when classes are well separated [25]. With more than 14,000 vectors the BIC keeps falling as classes are added, a known behaviour in large samples, so we fix the number of classes by a stated rule: K is the largest number for which adding one class still raises the log-likelihood by at least 0.5%. Algorithm 2 summarises the procedure.

Algorithm 2: EM estimation of the latent class model for CVSS vectors

Require: item matrix x_{ij} with R_j levels; candidate $K \in 1,\dots,10$; starts S; tolerance $\varepsilon = 10^{-8}$

Ensure: selected K^*, class weights $\hat{\pi}$, item-response probabilities $\hat{\theta}$, posteriors τ

1: for $K = 1$ to 10 do
  2: for $s = 1$ to S do
    3:draw $\theta_{jk\cdot} \sim Dirichlet(1)$; $\pi_k \leftarrow 1/K$
    4: repeat
      5:compute τ_{ik} by Eq. (7) in log space ▷ E-step
      6:update π_k, θ_{jkr} by Eq. (8) ▷ M-step
    7: until relative change in $\log L < \varepsilon$
  8: end for
  9:keep the start with the largest $\log L_K$; compute BIC_K and E_K by Eqs. (9)–(10)
10: end for
11: $K^* \leftarrow maxK : (\log L_K - logL_{K-1}) / |\log L_{K-1}| \geq 0.005$
12:assign each CVE to $argmax_k \tau_{ik}$ under K^*

13: return $K^*, \hat{\pi}, \hat{\theta}, \tau$

4.3 Recovering device CVEs from text

Unenriched CVEs have descriptions but no CPE data. We train a classifier on enriched CVEs, where δ_i is known, and apply it to the rest. Descriptions are represented by TF-IDF weights over word unigrams and bigrams [26],

$$w_{t,i} = (1 + \log f_{t,i}) \left(1 + \log \frac{1+n}{1+n_t}\right), \tag{11}$$

with $f_{t,i}$ the frequency of term t in description i and n_t the number of training descriptions containing it, followed by l_2 normalisation of each vector. The main model is l_2-penalised logistic regression,

$$\begin{aligned} \hat{w} = \operatorname*{argmin}_{w,b} & \frac{1}{2C}\|w\|_2^2 \\ & + \sum_i \log\left(1 + e^{-\bar{\delta}_i (w^\top x_i + b)}\right) \\ \bar{\delta}_i = 2\delta_i - 1 & \end{aligned} \tag{12}$$

with $C = 4$; a linear support vector machine calibrated by Platt scaling [27] and complement naive Bayes serve as comparisons. The split follows time. Training uses CVEs published January 2022 to June 2024, the decision threshold t^* maximises the F1 score

$$F_1(t) = \frac{2\,\mathrm{Prec}(t)\,\mathrm{Rec}(t)}{\mathrm{Prec}(t) + \mathrm{Rec}(t)} \tag{13}$$

on July–December 2024, and the test set is every enriched CVE published from January 2025 to June 2026.

Applying the classifier to unenriched CVEs and counting positives, the method known as classify and count, is biased whenever the true-positive rate falls short of one or the false-positive rate exceeds zero, and the adjusted classify-and-count estimator [29] removes that bias by inverting the misclassification,

$$\hat{p}_{ACC} = \frac{\hat{q} - \hat{\mathrm{FPR}}}{\hat{\mathrm{TPR}} - \hat{\mathrm{FPR}}}, \qquad \hat{p}_{PCC} = \frac{1}{m}\sum_{i=1}^{m} \hat{s}_i, \tag{14}$$

where $\hat{q}$ is the share of unenriched CVEs above t^* and both rates come from the test set. The probabilistic estimate $\hat{p}_{PCC}$ averages predicted probabilities $\hat{s}_i$ [30]. A joint bootstrap over test and unenriched sets (300 resamples) gives the interval for $\hat{p}_{ACC}$. The estimator assumes that the class-conditional distribution of descriptions is the same in enriched and unenriched records, which cannot be tested directly; its plausibility is examined by CNA in Section 5.5. Algorithm 3 gives the procedure.

Algorithm 3: Temporal text classification and adjusted classify-and-count quantification

Require: enriched CVEs with (d_i, δ_i, t_i); unenriched descriptions d_u; bootstrap size $B = 300$

Ensure: test metrics, threshold t^*, prevalence estimates $\hat{p}_{CC}, \hat{p}_{ACC}, \hat{p}_{PCC}$ with interval

1:split enriched CVEs into train ($t_i <$ Jul 2024), validation (Jul–Dec 2024) and test ($t_i \geq$ Jan 2025)

2:fit TF-IDF on training descriptions by Eq. (11); transform all sets

3:fit $\hat{w}$ by Eq. (12)

4: $t^* \leftarrow \mathrm{argmax}_t F_1(t)$ on validation ▷ Eq. (13)

5:compute AUC, average precision, F_1, TP̂R, FP̂R on test at t^*

6: $\hat{q} \leftarrow m^{-1}\sum_u \mathbb{1}[\hat{s}_u \geq t^*]$; $\hat{p}_{CC} \leftarrow \hat{q}$

7:compute $\hat{p}_{ACC}$ and $\hat{p}_{PCC}$ by Eq. (14), clipping $\hat{p}_{ACC}$ to $[0,1]$

8: for $b = 1$ to B do

9:resample test and unenriched sets; recompute $\hat{\mathrm{TPR}}_b$, $\hat{\mathrm{FPR}}_b$, $\hat{q}_b$ and $\hat{p}_{ACC,b}$

10: end for

11: return metrics, t^*, $\hat{p}_{CC}$, $\hat{p}_{ACC}$ with 2.5% and 97.5% percentiles, $\hat{p}_{PCC}$

4.4 Topics, templates and end-of-life language

Topic structure in device descriptions is extracted with non-negative matrix factorisation of the TF-IDF matrix $X \in R_{\geq 0}^{n \times m}$ into WH with K topics,

$$\min_{W \geq 0, H \geq 0} \|X - WH\|_F^2, \tag{15}$$

solved by the multiplicative updates of Lee and Seung [32],

$$H \leftarrow H \odot \frac{W^\top X}{W^\top W H}, \qquad W \leftarrow W \odot \frac{X H^\top}{W H H^\top}, \tag{16}$$

with element-wise products and quotients. The number of topics is chosen to maximise mean coherence, measured by the normalised pointwise mutual information of the ten top terms [33, 34],

$$\mathrm{NPMI}(w_a, w_b) = \frac{\log \frac{P(w_a, w_b)}{P(w_a)P(w_b)}}{-\log P(w_a, w_b)}, \tag{17}$$

with probabilities estimated from document co-occurrence. We fit the model twice. The first fit uses descriptions as published, and the second removes recurring boilerplate sentences inserted by specific CNAs (for instance, VulDB's note that "the vendor was contacted early about this disclosure but did not respond in any way" and MediaTek's "Patch ID" and "Issue ID" lines). The comparison tells us how much of the apparent topic structure reflects reporting templates. Separately, a regular expression flags descriptions that say a product is end of life, no longer supported or no longer maintained.

Algorithm 4: NMF topic extraction with boilerplate removal and coherence-based selection

Require: device descriptions d_i; boilerplate patterns B; candidate $K \in 6, 8, \ldots, 16$; iterations $T_{\max} = 800$

Ensure: selected K^*, topic–term matrix H, document–topic matrix W

1: for all d_i do

2:delete every sentence matching a pattern in B; delete identifiers, version strings and vendor names

3: end for

4:build X from TF-IDF weights of unigrams and bigrams ▷ Eq. (11)

5: for all K do

6: initialise W, H by NNDSVD with zeros set to the mean of X

7: for $t = 1$ to $T_{\max}$ do

8:update H and W by Eq. (16)

9: end for

10: $\bar{c}_K \leftarrow$ mean NPMI of the ten top terms per topic ▷ Eq. (17)

11: end for

12: $K^* \leftarrow \arg\max_K \bar{c}_K$; refit and assign each CVE to $\arg\max_k W_{ik}$

13: return K^*, W, H

4.5 Exploitation timing

For every enriched CVE, the time T_i from NVD publication to KEV listing is observed or censored at 29 July 2026, plus half a day so that same-day listings fall after time zero; CVEs listed before NVD publication receive a lag of zero. Severity in this model is the disclosure score. It takes the CNA's assessment where one exists and NVD's otherwise, because NVD's own score is often added after listing. The cumulative probability of listing is one minus the Kaplan–Meier estimate [35]. Covariate effects come from the Cox model [36],

$$h(t \mid z_i) = h_0(t) \exp\left(\gamma \delta_i + z_i^{\top} \beta\right), \tag{18}$$

where δ_i is the device indicator and z_i contains the severity band, three CVSS vector components, five weakness families, the log of one plus the issuing CNA's earlier KEV listings, and the publication year. The hazard ratio e^{γ} is the quantity of interest. It tells us whether, with severity and weakness held fixed, a device CVE is more or less likely to be listed.

4.6 Vendor-level empirical Bayes

Raw shares of Critical CVEs are noisy for vendors with few records, so for vendor v with n_v scored device CVEs, of which k_v are Critical, we assume

$$k_v \mid \theta_v \sim \text{Binomial}(n_v, \theta_v), \qquad \theta_v \sim \text{Beta}(a, b), \tag{19}$$

estimate (a, b) by maximising the beta-binomial marginal likelihood over the vendors with $n_v \geq 10$ [38], and report the posterior mean and 95% interval [39],

$$\bar{\theta}_v = \frac{k_v + a}{n_v + a + b}, \qquad \theta_v \mid k_v \sim \text{Beta}(k_v + a,\ n_v - k_v + b). \tag{20}$$

The intra-vendor correlation $\rho = 1/(a + b + 1)$ measures how much of the variation in Critical shares is between vendors, and a likelihood-ratio test against a common binomial rate checks whether the hierarchy is needed.

Algorithm 5: Beta-binomial empirical Bayes for vendor severity profiles

Require: device CVEs with vendor v and base score s; minimum size $n_{\min} = 10$

Ensure: $(\hat{a}, \hat{b})$, $\hat{\rho}$, LR, posterior means and intervals $\bar{\theta}_v$

1: $P \leftarrow (k_v, n_v) : n_v \geq n_{\min}$ with $k_v = \#s \geq 9.0$

2: $(\hat{a}, \hat{b}) \leftarrow argmax \sum_v \log \Pr(k_v \mid n_v, a, b)$ ▷ Nelder–Mead on $\log a, \log b$

3: $\hat{\rho} \leftarrow 1/(\hat{a} + \hat{b} + 1)$; $\hat{p} \leftarrow \sum k_v / \sum n_v$

4: $LR \leftarrow 2\left[l_{BB}(\hat{a}, \hat{b}) - l_{Bin}(\hat{p})\right]$

5: for all $v \in P$ do

6: $\bar{\theta}_v \leftarrow (k_v + \hat{a}) / (n_v + \hat{a} + \hat{b})$ ▷ Eq. (20)

7:interval $\leftarrow$ 2.5% and 97.5% quantiles of $Beta(k_v + \hat{a}, n_v - k_v + \hat{b})$

8: end for

9: return $\hat{a}, \hat{b}, \hat{\rho}, LR, \bar{\theta}_v$

5. RESULTS

5.1 Volume, trend and reporting institutions

Device CVEs are a steady but shrinking slice of the record. Among enriched CVEs the device share was 13.4% in 2022 and 14.2% in 2023, then fell to 10.1% in 2024, 10.7% in 2025 and 7.4% in the first half of 2026 (Figure 1). The negative binomial model of Eq. (4) puts the yearly multiplicative change at 0.897 (95% CI 0.839–0.958, $p = 0.001$), a decline of about a tenth per year. Panel (b) of Figure 1 carries the warning that goes with this number. Enrichment coverage fell over the same years, from 100% to 63%, so a share measured on enriched records alone could fall merely because NVD stopped enriching the kinds of CVE that concern devices while it kept enriching the others. Section 5.5 tests that possibility. It explains part of the drop in counts but almost none of the drop in share.

The composition of the device record changed faster than its size. Network devices rose from 50.6% of device CVEs in 2022 to about 60% in each year from 2024, while the chipset and modem category, fed by the monthly bulletins of Qualcomm, MediaTek and Unisoc, peaked at 1,153 CVEs (29.8%) in 2023 and fell back to 14.4% in 2026 (Figure 2). The reporting institutions moved too. MITRE, which assigns identifiers to researchers whose target has no vendor CNA of its own, issued 33.1% of device CVEs over the period, and the three chipset vendors together issued another 18.2% through their own monthly security bulletins. VulDB is the change that stands out. Its share of device CVEs went from 0.4% in 2022 to 16.5% in 2024 and 32.4% in 2026, and 90.2% of VulDB's device CVEs concern network devices, mostly low-cost routers from Tenda, TOTOLINK, D-Link and similar vendors. In 41.5% of them the description records that the vendor was contacted and did not respond.

5.2 Severity, attack vectors and weakness types

Device CVEs score higher than the rest (Table 3). A quarter of them are Critical, against 13.7% of non-device CVEs. High or Critical scores cover 71.2% against 51.1%, and the mean base score is 7.79 against 6.97. Two vector components carry most of the gap. No user interaction is needed for 93.6% of device CVEs but for only 67.0% of other CVEs, which reflects the absence of browser-borne and document-borne attacks on headless devices, and adjacent-network vectors, typical of Wi-Fi, Bluetooth and cellular baseband flaws, are eight times as common (10.3% against 1.3%). Network attack

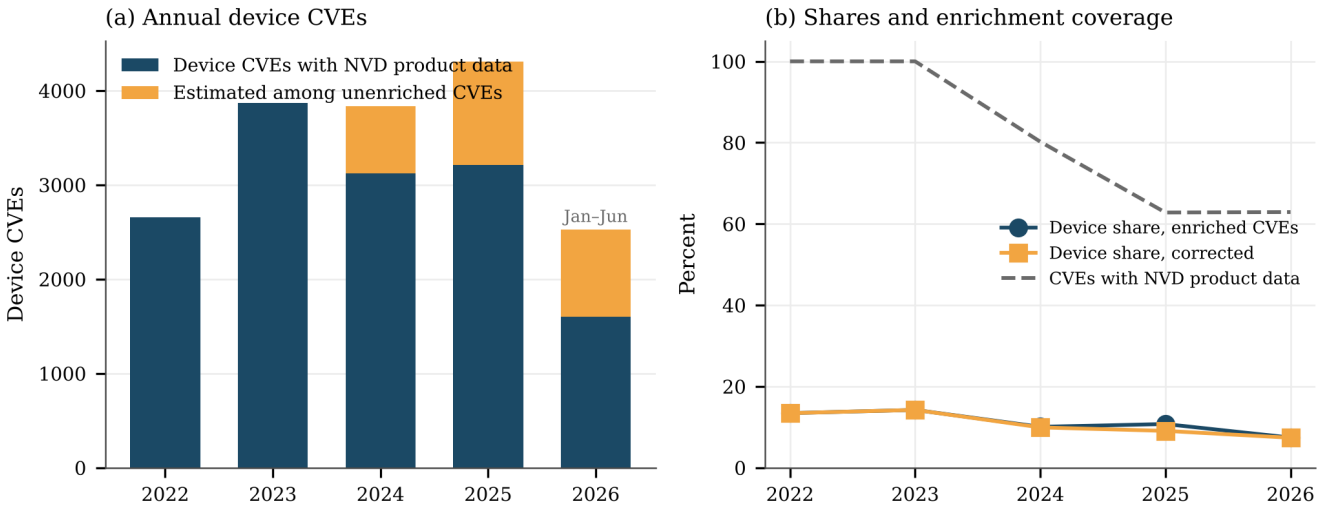


Figure 1. Device CVEs by publication year. (a) Device CVEs observed among enriched records and the corrected total that adds the adjusted classify-and-count estimate for unenriched records; 2026 covers January–June. (b) Device share of enriched CVEs, corrected device share and the share of all CVEs that NVD enriched.

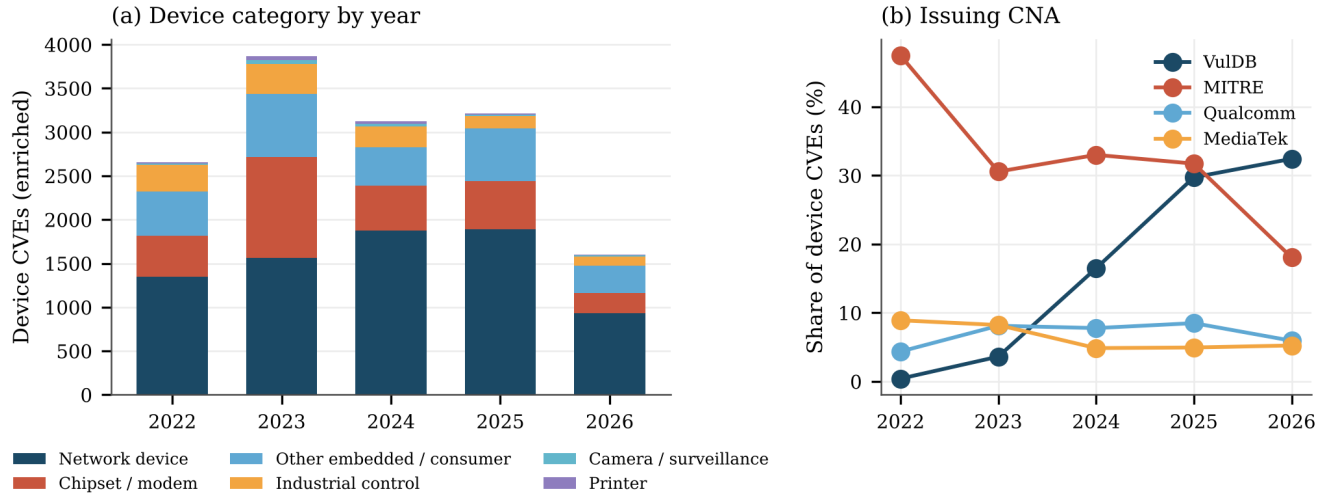


Figure 2. (a) Enriched device CVEs by category and publication year. (b) Share of device CVEs issued by each CNA, by year; "Other" groups all remaining CNAs.

vectors are, if anything, slightly less frequent among device CVEs. NVD attached an "Exploit" reference tag to 40.5% of device CVEs and 28.5% of others, a sign that researchers who report router flaws often publish working proof-of-concept code with them.

Figure 3 puts the same comparison on the full score distribution. Device scores pile up at 9.8, the score of an unauthenticated network flaw with high impact on all three properties, which alone accounts for 23.0% of scored device CVEs, with further peaks at 8.8, the score the same vector receives when low privileges or adjacency are added, and at 7.5, its availability-only variant. Non-device scores are spread more evenly. Their sixth most frequent value is 5.4, the typical score of stored cross-site scripting.

Weakness types separate the two groups more sharply than scores do (Table 4 and Figure 4). Memory-safety flaws, chiefly out-of-bounds writes (CWE-787), stack-based buffer overflows (CWE-121) and classic buffer copies without size checks (CWE-120), make up 41.5% of device CVEs and only 18.4% of other CVEs; the odds ratio is 3.15. OS command injection (CWE-78) and generic command injection (CWE-77), which in router firmware typically means a web parameter passed unchecked to a shell call, push the injection family to 17.6% against 6.8%, an odds ratio of 2.91. These two families account for nearly six in ten device CVEs. Web-application weaknesses go the other way. Cross-site scripting has an odds ratio of 0.19. SQL injection has 0.09, and the path-traversal, upload and request-forgery family 0.35. Authentication and access-control flaws, which policy documents on IoT security usually put first [7, 8], are about equally common in both groups (odds ratio 0.95, interval 0.90–1.00), so the weakness that the baselines stress most is the one that sets devices apart least.

The vendor distribution is heavily concentrated. Device CVEs name 871 distinct primary vendors, and the Gini coefficient of Eq. (5) is 0.85 (Figure 5). Ten vendors hold 54.0% of all device CVEs. Tenda alone accounts for 1,791 (12.4%), followed by D-Link (1,194), Qualcomm (1,044), TOTOLINK (1,036) and MediaTek (950). Three of the top five are makers of consumer routers sold mostly at low prices. Part of the concentration reflects how flaws are counted. Of Tenda's device CVEs, 82.3% describe a stack or buffer overflow, usually in one HTTP handler of one firmware image, and every affected handler or parameter receives its own identifier,

Table 3. Device and non-device CVEs compared, enriched CVEs (percentages with 95% Wilson intervals)

Measure	Device CVEs	Non-device CVEs
Critical (base score $\geq$ 9.0)	24.8 (24.1–25.5)	13.7 (13.5–13.9)
High or Critical ($\geq$ 7.0)	71.2 (70.5–71.9)	51.1 (50.8–51.4)
Mean base score	7.79	6.97
Attack vector network	68.2 (67.5–69.0)	71.1 (70.9–71.4)
Attack vector adjacent network	10.3 (9.8–10.8)	1.3 (1.2–1.3)
No privileges required	56.6 (55.8–57.4)	51.4 (51.1–51.7)
No user interaction	93.6 (93.2–94.0)	67.0 (66.7–67.3)
NVD "Exploit" reference tag	40.5 (39.7–41.3)	28.5 (28.2–28.7)
In KEV catalog	0.66 (0.54–0.81)	0.55 (0.51–0.60)
KEV with known ransomware use	0.08 (0.04–0.14)	0.11 (0.09–0.13)

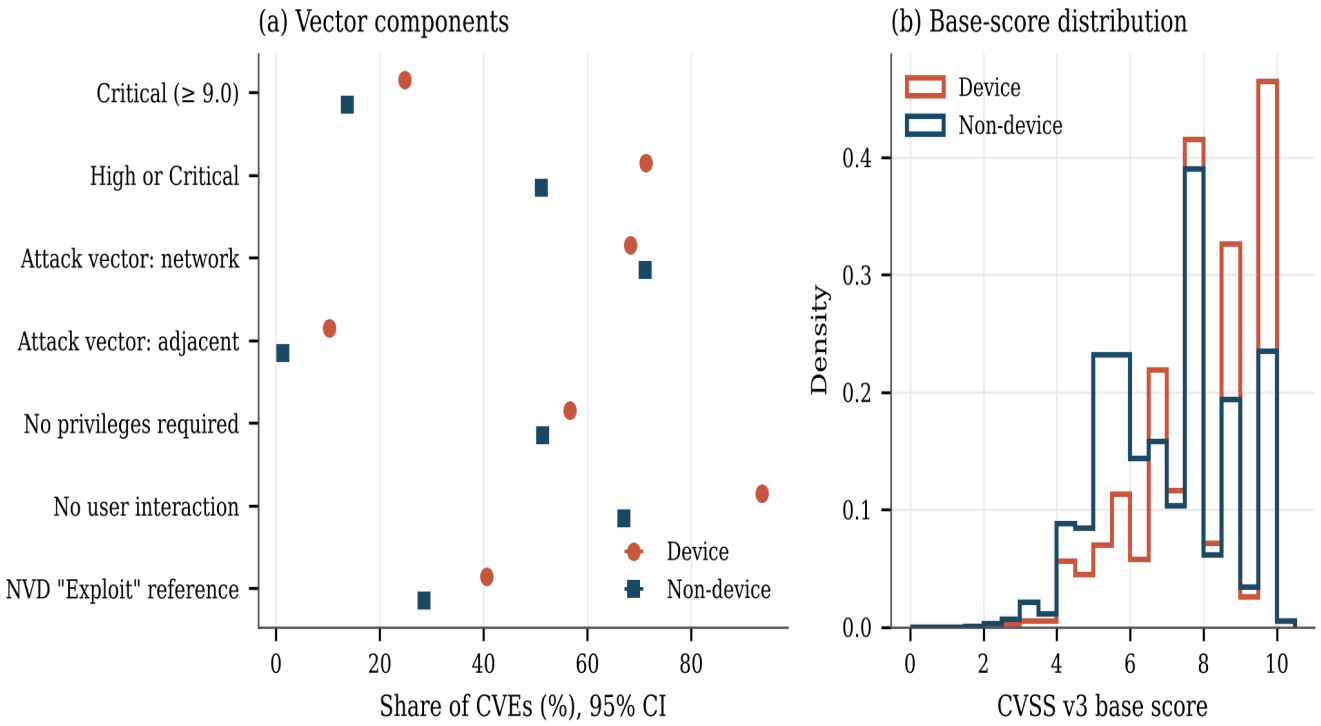


Figure 3. (a) CVSS v3.x vector components for device and non-device CVEs, with 95% Wilson intervals. (b) Density of base scores.

Table 4. CWE families among device and non-device CVEs, with Haldane-corrected odds ratios (Eq. 3)

CWE family	Device (%)	Non-device (%)	Odds ratio	95% CI
Memory safety	41.5	18.4	3.15	3.04–3.26
Command / code injection	17.6	6.8	2.91	2.77–3.05
Authentication / access	13.5	14.0	0.95	0.90–1.00
Input validation / DoS	3.6	4.6	0.77	0.70–0.84
Information / crypto	2.8	3.4	0.83	0.75–0.92
Path / upload / SSRF	2.2	6.1	0.35	0.31–0.39
Cross-site scripting	3.0	14.3	0.19	0.17–0.21
SQL injection	0.6	5.9	0.09	0.07–0.11
Unspecified	2.2	7.6	0.28	0.25–0.31
Other	13.0	18.8	0.65	0.61–0.68

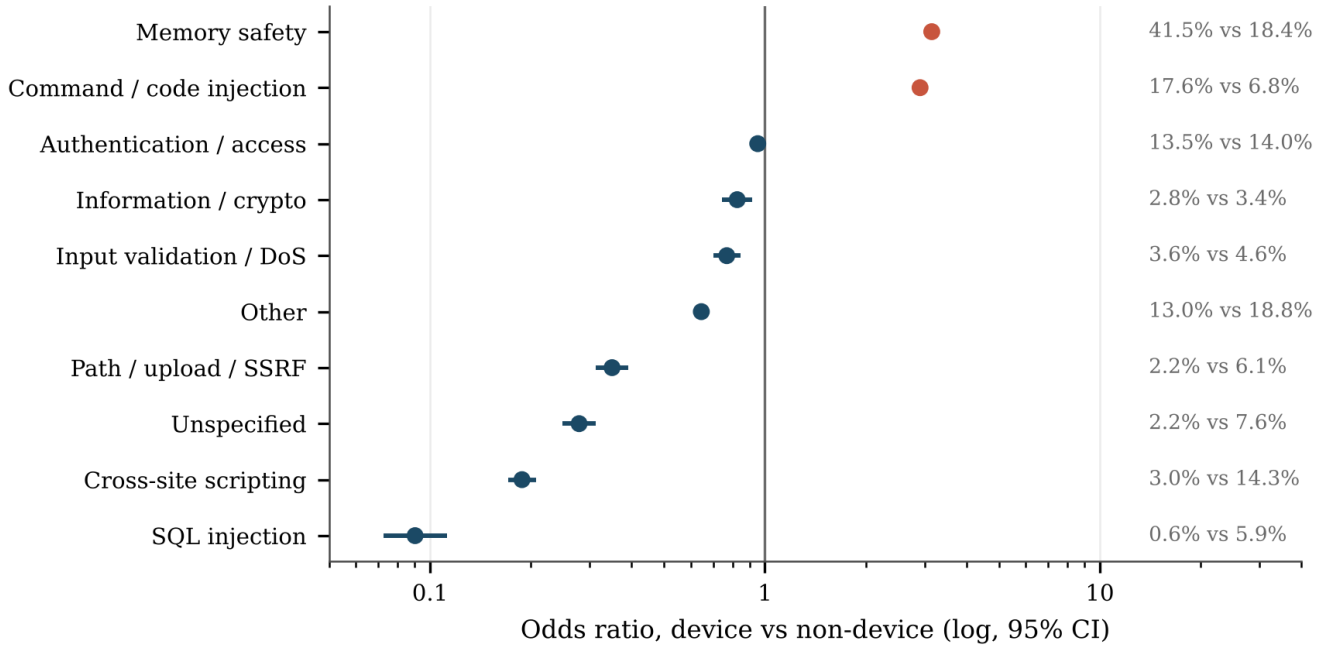


Figure 4. Odds ratios (log scale) for each CWE family, device against non-device CVEs, with 95% confidence intervals.

so a single weak code path can account for many entries in the vendor's count.

5.3 Latent attack profiles

Table 5 reports the fit of latent class models with one to ten classes, estimated on the 14,457 device CVEs that carry a CVSS v3.x vector, and BIC falls at every step (Figure 6), which is expected when the sample is large and the item space is coarse. The log-likelihood gains tell a clearer story: 11.2% from one class to two, 5.8% from two to three, and then a slow decay to 1.06% from seven to eight and 0.50% from eight to nine. The ninth class misses the 0.5% threshold by a hair (0.495%). Eight is the rule's verdict, and a close one. Relative entropy for the eight-class model is 0.968. The mean maximum posterior probability is 0.978, so almost every vector is assigned to one class with near certainty.

The eight classes are easy to name from their item-response probabilities (Table 6). The largest class, C1, holds a third of device CVEs and is the textbook firmware vulnerability: a network or adjacent attacker, no privileges, no user interaction and high impact on confidentiality, integrity and availability, with a mean score of 9.40. C2 differs mainly in needing privileges. Its mean is 7.95, and 38% of its vectors are local. C3 is an unauthenticated availability-only class, and 57.3% of its members are memory-safety flaws whose reporters claimed a crash and no code execution. The five smaller classes cover confidentiality-only disclosures, partial-impact web flaws, local disclosures, cross-site-scripting-like vectors with changed scope and required interaction, and local denial of service.

Class membership differs by device category (Figure 7, panel a). Printers fall into C1 most often (57%), followed by other embedded and consumer devices (42%) and cameras and network devices (38% and 37%). Chipset and modem CVEs are concentrated in C2 (45%) and in the two local classes C6 and C8 (15% and 13%), as one would expect for flaws that are reached through an app or a radio stack on the handset. Exploitation follows the classes closely (panel b). C1 has a KEV rate of 1.17% and C2 of 0.68%. The remaining six classes together have 12 listings among 5,448 CVEs (0.22%). That memory-safety share is a warning about scoring practice. The same stack overflow can be filed as a denial of service or as code execution depending on how much effort the reporter put into exploitation, and the class then sets its score at 7.5 or 9.8.

5.4 A text classifier for device CVEs

Table 7 compares three classifiers on the 2025–2026 test set, where 9.3% of the 51,459 enriched CVEs are device CVEs. All three discriminate well. Logistic regression reaches an AUC of 0.983 (bootstrap 95% interval 0.982–0.985) and an average precision of 0.930, and at the validation-selected threshold of 0.37 it attains an F1 of 0.871 with precision 0.918, recall 0.828 and a false-positive rate of 0.77%. The calibrated support vector machine is indistinguishable (F1 0.873), and complement naive Bayes trails on precision. Performance did not decay over the test period: the AUC was 0.982 for CVEs published in 2025 and 0.986 for those published in the first half of 2026, with F1 of 0.877 and 0.857. Figure 8 gives the logistic model's ROC and precision–recall curves for each test year; the ROC panel uses a symmetric-log horizontal axis so that the region of low false-positive rates, where the threshold sits, can be read.

What the classifier learns is part product vocabulary and part reporting template (Figure 9). Vendor and product terms ("tenda", "totolink", "netgear", "router", "modem", "camera", "goform", "cgi") carry the largest positive weights, but so do phrases that appear only because particular CNAs use fixed wording: "patch id" and "issue id" from MediaTek bulletins, "memory corruption" and "transient dos" from Qualcomm's. The negative side is dominated by web and desktop software ("plugin", "wordpress", "windows", "sql", "linux kernel"). Template terms make the classifier accurate on records written by the same CNAs. They also tie its accuracy to those CNAs' house style, which matters for the quantification below.

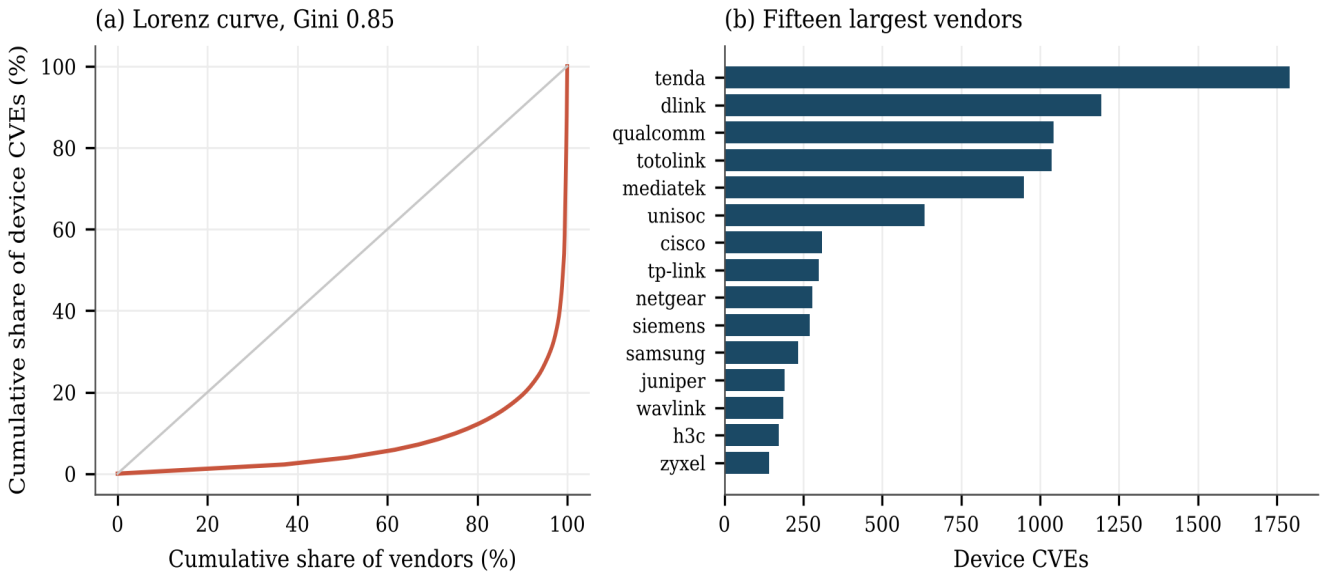


Figure 5. (a) Lorenz curve of device CVEs over primary vendors. (b) The 15 vendors with the most device CVEs (CPE vendor names).

Table 5. Fit of latent class models for CVSS v3.x vectors of device CVEs

Classes K	Log-likelihood	Parameters	BIC	Gain in log-likelihood (%)	Relative entropy
1	-67,441.4	14	135,016.9	–	1.000
2	-59,878.5	29	120,034.9	11.21	0.945
3	-56,424.2	44	113,270.0	5.77	0.977
4	-54,265.2	59	109,095.6	3.83	0.978
5	-53,051.3	74	106,811.4	2.24	0.985
6	-52,028.7	89	104,909.8	1.93	0.976
7	-51,412.9	104	103,822.0	1.18	0.964
8	-50,866.6	119	102,873.1	1.06	0.968
9	-50,614.6	134	102,512.7	0.50	–
10	-50,464.8	149	102,356.9	0.30	–

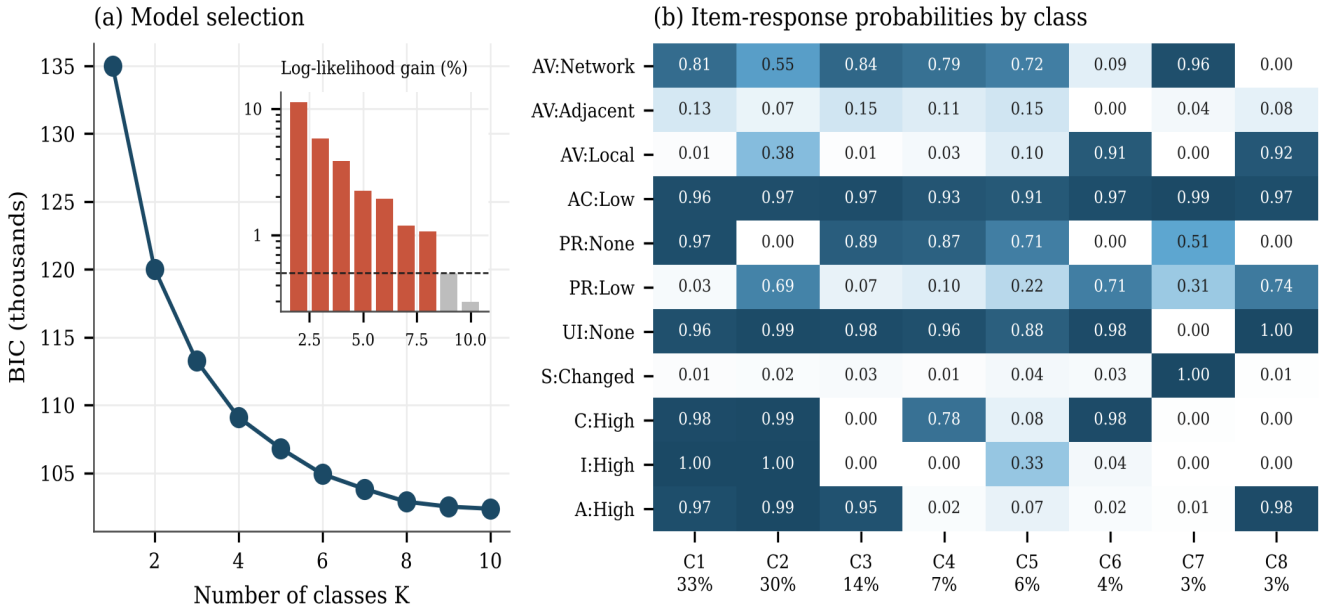


Figure 6. (a) BIC against the number of latent classes; the inset gives the percentage gain in log-likelihood from each added class, with the 0.5% rule as a dashed line. (b) Item-response probabilities of the eight-class model; columns are classes with their weights.

Table 6. Profiles of the eight latent classes (classes ordered by weight)

Class	Profile	Share (%)	CVEs	Mean base score	Memory safety (%)	Command injection (%)	In KEV
C1	Remote unauthenticated, full impact	32.6	4,604	9.40	40.3	25.9	54 (1.17%)
C2	Privileged, full impact	29.6	4,405	7.95	52.0	25.9	30 (0.68%)
C3	Unauthenticated availability only	14.0	2,004	7.01	57.3	0.3	1 (0.05%)
C4	Unauthenticated, confidentiality only	7.2	1,074	6.39	6.7	0.7	3 (0.28%)
C5	Partial impact, web interface	6.4	901	6.12	18.7	16.7	7 (0.78%)
C6	Local information disclosure	3.9	534	5.31	26.8	2.1	1 (0.19%)
C7	Scope changed, user interaction	3.3	481	5.67	0.0	7.1	0
C8	Local denial of service	3.0	454	5.21	70.9	0.4	0

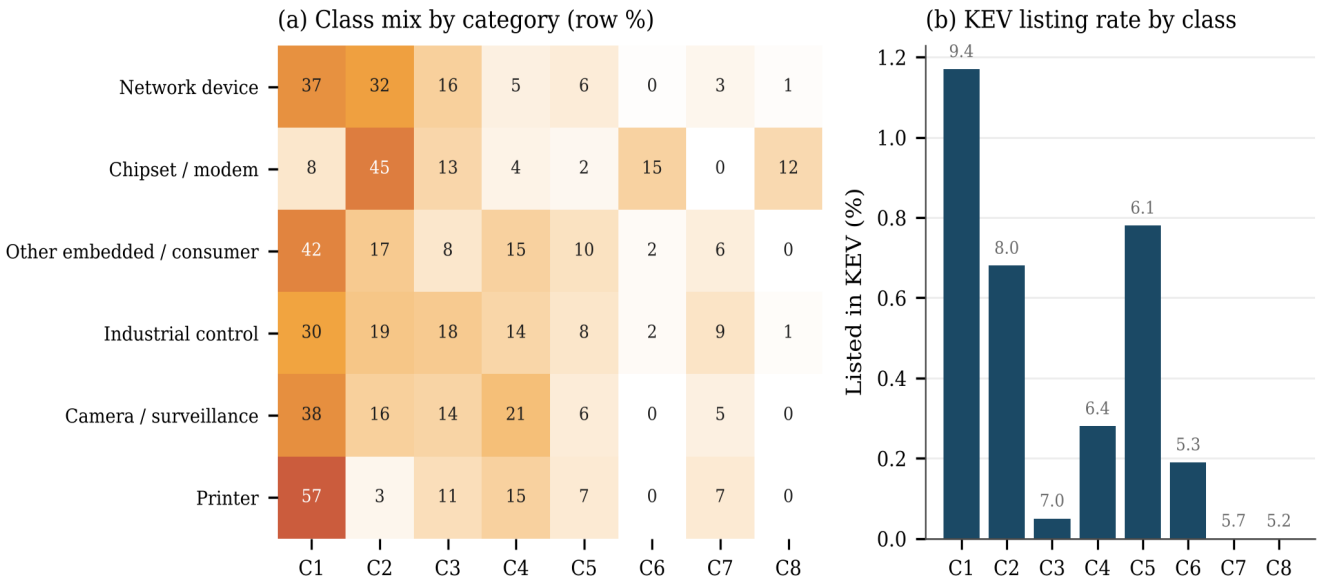


Figure 7. (a) Distribution of each device category over the eight latent classes (row shares). (b) Share of each class listed in the KEV catalog.

Table 7. Classifier performance on the temporal test set (enriched CVEs published January 2025 – June 2026)

Classifier	Threshold	AUC	Average precision	F1	Precision	Recall	False-positive rate (%)
Logistic regression	0.37	0.983	0.930	0.871	0.918	0.828	0.77
Linear SVM (Platt-calibrated)	0.52	0.983	0.930	0.872	0.907	0.840	0.89
Complement naive Bayes	0.63	0.978	0.887	0.804	0.796	0.812	2.14

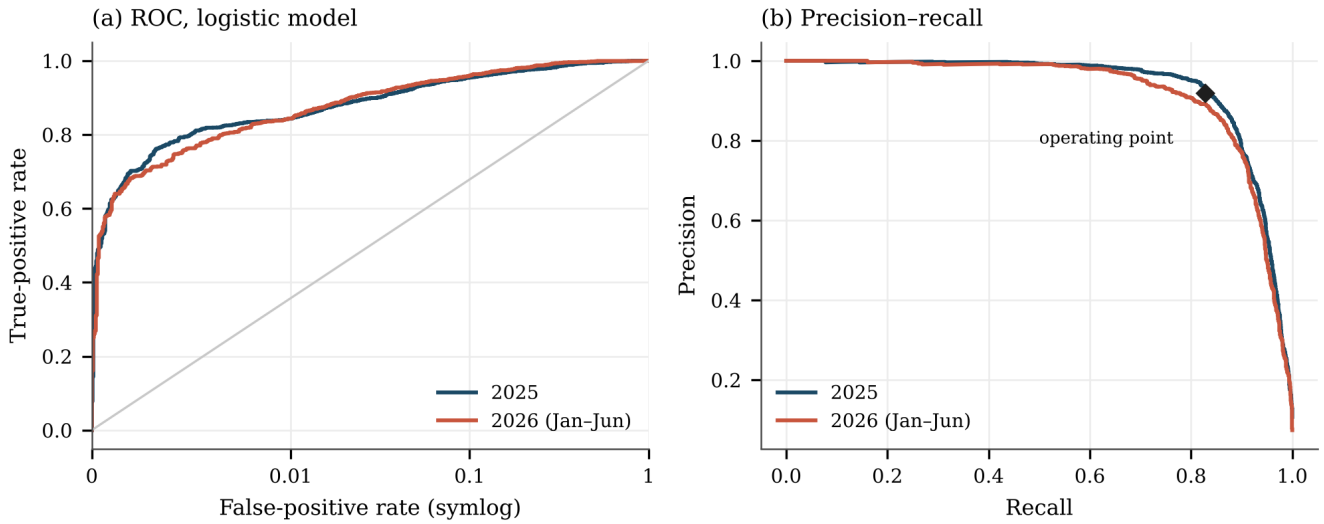


Figure 8. Logistic regression classifier on the temporal test set, CVEs published in 2025 and in January–June 2026. (a) ROC curves (symmetric-log horizontal axis). (b) Precision–recall curves; the diamond marks the operating point of the whole test set at the threshold chosen on validation data.

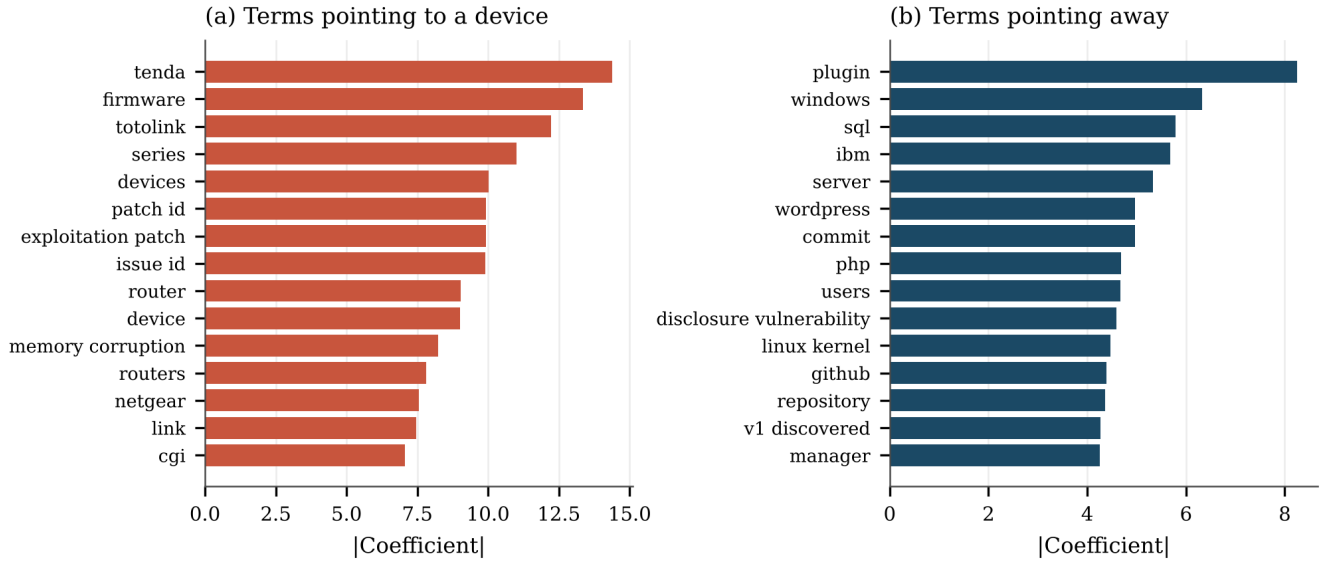


Figure 9. Terms with the largest positive (device) and negative (non-device) coefficients in the logistic regression classifier.

5.5 Device CVEs hidden among unenriched records

There are 38,079 unenriched CVEs in the cohort, all published from 2024 onward. Classify-and-count puts 6.67% of them above the threshold; adjusted classify-and-count, which corrects for the 82.8% true-positive rate and the 0.77% false-positive rate, raises the estimate to 7.20% (bootstrap interval 6.90–7.48%), or about 2,741 device CVEs, and the probabilistic estimate gives 7.08%. The three estimators agree within half a percentage point. Table 8 breaks the estimate down.

The corrected totals change the count of device CVEs a good deal. For 2025 the estimate is 4,311, about a third more than the 3,211 visible from CPE data, and for the first half of 2026 it is 2,527 against 1,600, so a reader who filters NVD by device CPEs misses roughly one device CVE in four for 2025 and more than one in three for 2026. The share is a different matter. Corrected shares of 9.9%, 9.1% and 7.4% for 2024, 2025 and 2026 sit close to the observed 10.1%, 10.7% and 7.4% (Figure 1), because unenriched CVEs are on average less likely to concern devices than enriched ones. The decline in the device share found in Section 5.1 is not an artefact of the enrichment gap.

The breakdown by CNA (Figure 10) locates the hidden records. Nearly half of the unenriched CVEs issued by CISA's industrial control systems team are estimated to be device CVEs, as are about a fifth of the unenriched records from VulDB and MITRE. All three CNAs are heavy reporters of device flaws, so these estimates are plausible, and they also bear on the assumption behind Eq. (14). The estimator requires that the classifier behave on unenriched records as it does on enriched ones. The CNAs that dominate the hidden device records are those whose templates the classifier learned, which supports the assumption for them, while the 31,185 unenriched records from all other CNAs contribute a low estimated share (3.8%) where a modest change in the false-positive rate would move the estimate most. Manual labelling of a random sample of unenriched records would settle how far the assumption holds; we did not have the resources for it.

Table 8. Estimated device share among unenriched CVEs and corrected yearly totals

Group	Unenriched CVEs	Classify and count (%)	Adjusted (%)	Probabilistic (%)	Observed device CVEs	Corrected total
2024	7,657	8.42	9.33	8.60	3,123	3,838
2025	17,703	5.86	6.21	6.19	3,211	4,311
2026 (Jan–Jun)	12,719	6.75	7.29	7.41	1,600	2,527
CNA: CISA ICS	406	–	48.3	–	–	–
CNA: VulDB	2,846	–	19.1	–	–	–
CNA: MITRE	3,203	–	18.2	–	–	–
CNA: all others	31,185	–	3.8	–	–	–
All unenriched	38,079	6.67	7.20	7.08	–	–

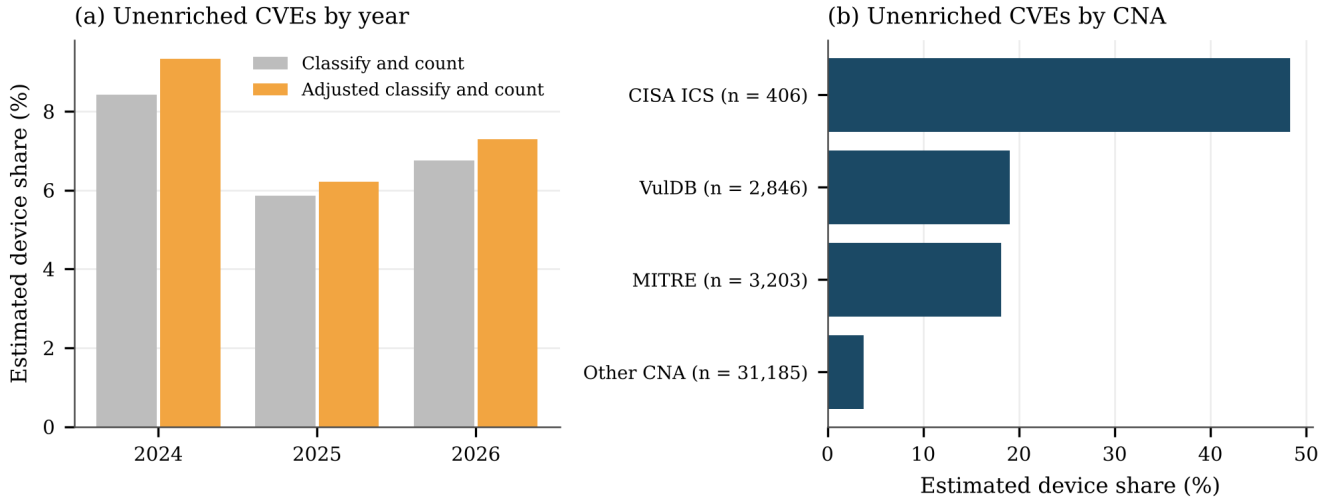


Figure 10. (a) Estimated device share among unenriched CVEs by publication year, by classify and count and by adjusted classify and count. (b) Adjusted estimate by issuing CNA.

5.6 Templates, topics and end-of-life language

Topic models on raw descriptions are dominated by reporting templates. In the six-topic model, the coherence-maximising choice on raw text (mean NPMI 0.576), three topics are essentially CNA signatures. One collects VulDB's sentence "The vendor was contacted early about this disclosure but did not respond in any way" (NPMI 0.979; 96% of its 1,485 CVEs come from VulDB), one MediaTek's "User interaction is not needed for exploitation" with its patch and issue identifiers, and one Unisoc's "missing permission check" bulletins. Six recurring phrases from these CNAs appear in 18.8% of all device descriptions. Coherence of this kind measures the uniformity of CNA prose and says little about the flaws themselves.

After the boilerplate, identifiers, version strings and vendor names were removed, coherence fell to between 0.32 and 0.37 for all candidate numbers of topics. The sixteen-topic model scored highest (0.374), only marginally ahead of fourteen topics (0.374) and eight (0.360), so the number of topics is weakly identified and we read the topics as a descriptive partition. Table 9 lists them. Three are chipset topics, each drawing about 90% or more of its CVEs from that category (bounds checks, missing permission checks, local privilege escalation), six draw more than 70% of theirs from network devices, and one, web interface access control, is the largest (3,333 CVEs) and the most mixed across categories. Two of the sixteen have negative coherence (the command-injection topic around "setting" parameters and the stack-overflow topic around router web forms), because their top terms are parameter and page names that rarely co-occur.

Figure 11 follows the topics over time. Most are stable, but two moved in opposite directions. The stack-overflow topic built on router web forms (T4) fell from 13.2% of device CVEs in 2022 to 0.3% in 2026, while the end-of-life topic (T13) rose from 0.2% to 16.3% and became the second-largest topic of 2026. T13 is 90% VulDB, and its defining terms ("longer supported", "maintainer") come from a sentence VulDB adds when the affected product is past its support period.

The rise of end-of-life language holds without the topic model. A regular expression for "no longer supported", "end of life" and similar phrases matches 2.19% of device CVEs (316) against 0.31% of non-device CVEs, and the device share rises from 0.11% in 2022 to 1.03% in 2023, 1.54% in 2024, 4.61% in 2025 and 4.81% in the first half of 2026 (Figure 12, panel b; panel a gives the frequency of each template phrase). Among vendors with at least 100 device CVEs, TRENDnet (21.6%) and D-Link (18.3%, 219 of 1,194) lead by a wide margin; DrayTek follows at 5.7%. Seven device CVEs whose descriptions say the product is unsupported are in the KEV catalog. The phrase is a fair proxy for a vulnerability that will never be fixed, and its growth means that a rising share of the public record on devices describes flaws that owners can only remove by replacing the device.

Table 9. Content topics after boilerplate removal (16-topic NMF model)

Topic	CVEs	NPMI	Critical (%)	Network vector (%)	VulDB (%)	In KEV	Main vendors
T1 Stack-based buffer overflow	848	0.47	29.6	94.8	75.5	1	Tenda, TOTOLINK, D-Link
T2 Out-of-bounds write (chipset)	702	0.71	6.1	15.2	0.1	0	MediaTek, Unisoc, Qualcomm
T3 Command injection in settings	1,120	-0.02	46.4	84.0	26.6	13	TOTOLINK, Tenda, D-Link
T4 Stack overflow in web forms	862	-0.12	61.5	85.6	0.0	0	Tenda, H3C, TOTOLINK
T5 Code execution (ZDI reports)	533	0.78	3.4	12.6	0.0	2	D-Link, Qualcomm, NETGEAR
T6 Denial of service	1,314	0.28	5.4	70.9	1.8	3	Tenda, Juniper, Unisoc
T7 Missing permission check (chipset)	675	0.62	1.9	21.2	4.1	0	Unisoc, Qualcomm, MediaTek
T8 Buffer overflow in HTTP daemon	867	0.05	33.1	84.5	25.0	3	Tenda, D-Link, TOTOLINK
T9 Web interface access control	3,333	0.23	21.2	67.4	2.8	30	Qualcomm, Cisco, Siemens
T10 Arbitrary command execution	1,092	0.33	40.9	76.7	0.0	23	TOTOLINK, D-Link, Cisco
T11 HTTP POST handler flaws	427	0.40	15.7	81.0	40.0	2	TOTOLINK, Tenda, Qualcomm
T12 Local privilege escalation (chipset)	527	0.53	4.0	11.0	0.4	2	MediaTek, Qualcomm, Unisoc
T13 End-of-life products	657	0.50	31.8	90.9	90.0	4	D-Link, Tenda, UTT
T14 Cross-site scripting	475	0.55	1.7	93.9	12.8	0	TOTOLINK, Tenda, NETGEAR
T15 Remote code execution in routers	433	0.19	55.9	79.0	0.0	12	TOTOLINK, D-Link, Tenda
T16 Crafted HTTP request trigger	592	0.48	24.8	88.3	2.2	1	Siretta, Milesight, Wavlink

5.7 Exploitation timing

Device CVEs are listed in the KEV catalog at about the same raw rate as other CVEs, 0.66% against 0.55%, and the unadjusted Cox hazard ratio is 1.17 (95% CI 0.95–1.45). The comparison changes once severity and weakness are held fixed. In the adjusted model of Eq. (18), the hazard ratio for device CVEs is 0.61 (0.49–0.77; Table 10): a device CVE is listed at about three-fifths the rate of a non-device CVE with the same severity band, vector components, weakness family, CNA history and year. The model discriminates well (concordance 0.807, 732 listings among 129,500 CVEs). Critical disclosure scores raise the hazard tenfold and High scores fourfold, injection and code-execution weaknesses double it, and SQL injection and cross-site scripting cut it to a fraction.

The reversal from 1.17 to 0.61 has a simple source. Device CVEs are concentrated in exactly the strata that predict listing, Critical scores and injection flaws, so their raw listing rate looks normal; within each stratum they are listed less often than non-device CVEs with the same profile. Memory-safety flaws, the largest device family, carry no extra hazard at

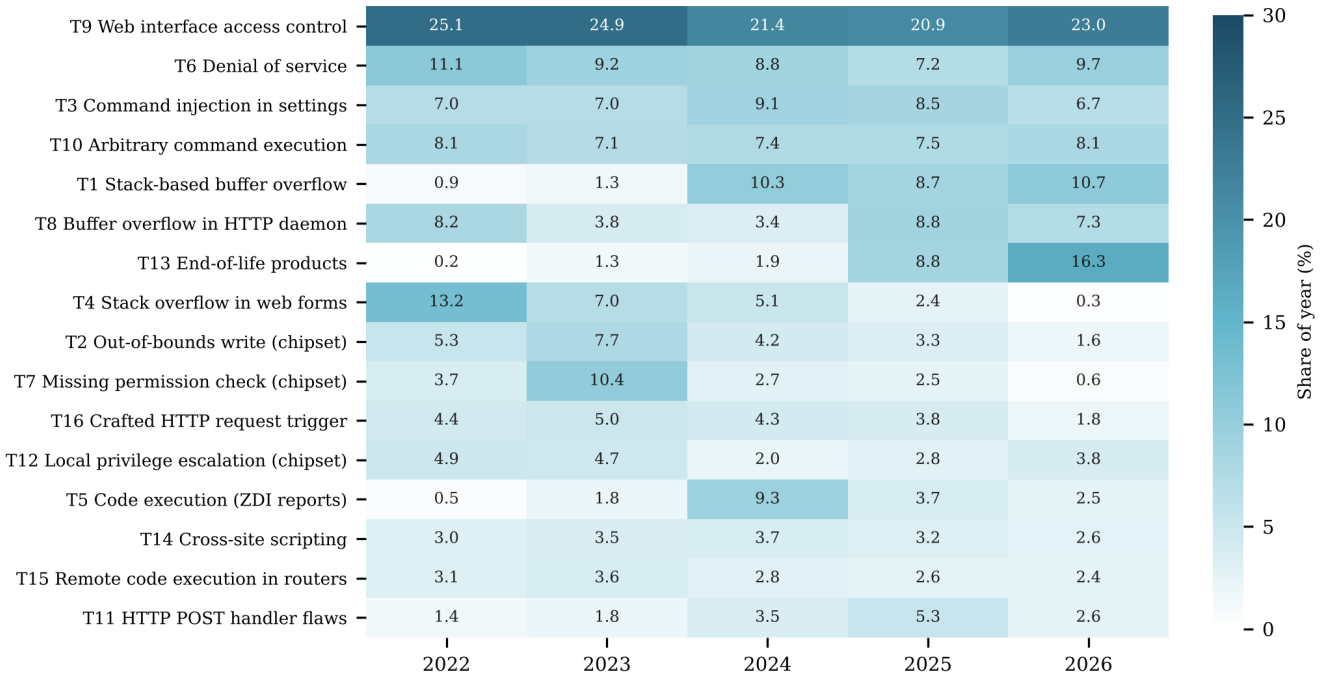


Figure 11. Share of each content topic among device CVEs published in each year (percent of the year's device CVEs). Topics are ordered by mean share.

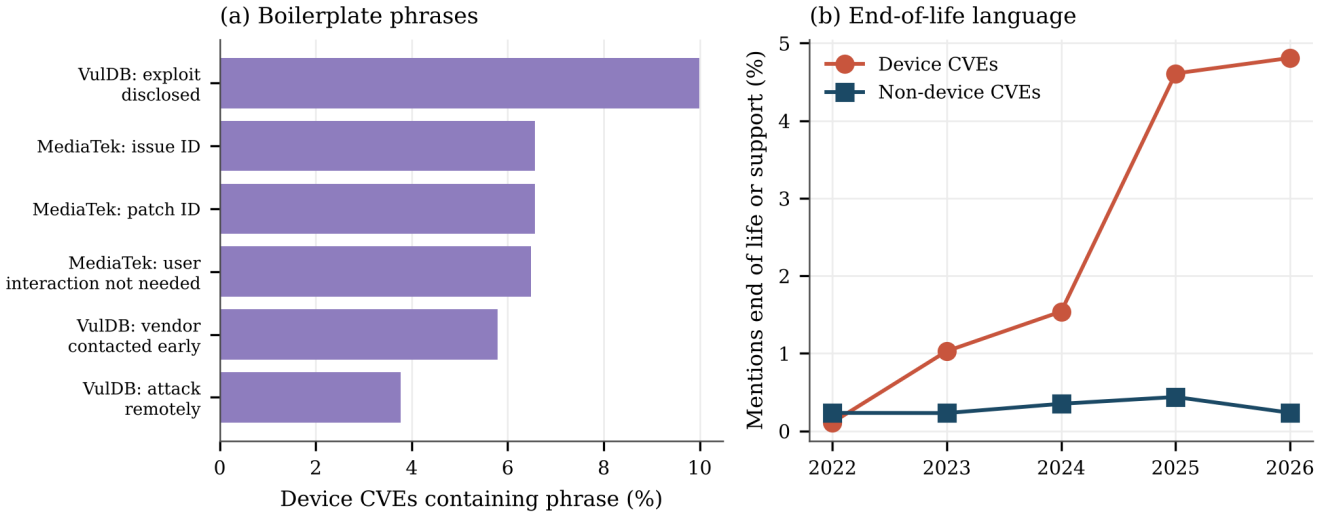


Figure 12. (a) Share of device CVEs containing six recurring CNA template phrases. (b) Share of device and non-device CVEs whose description states that the product is end of life or no longer supported, by publication year.

all (1.01). The categories differ as well (Figure 13). By one year after publication, 1.07% of industrial control CVEs and 0.58% of network device CVEs had been listed, against 0.54% of non-device CVEs and only 0.27% of chipset and modem CVEs. The camera category has the highest one-year figure (2.4%), but it rests on three listings among 127 CVEs. Device listings also come later. The median lag from NVD publication to listing is 21 days for device CVEs and 7 days for others, and 19.8% of device listings happened on or before the day of publication, against 28.1% of other listings, which fits a pattern in which device exploitation is discovered in the wild after disclosure more often than it is disclosed together with an attack.

5.8 Vendor severity profiles

Raw Critical shares of vendors with few CVEs range from zero to one, and most of that range is noise. The beta-binomial model of Eq. (19), fitted to 159 vendors with at least ten scored device CVEs, estimates $\hat{a} = 1.09$ and $\hat{b} = 3.08$, which correspond to a prior mean Critical share of 0.261 and an intra-vendor correlation of 0.193. The likelihood-ratio statistic against a common rate is 1,876, so vendors differ far more than binomial sampling allows. Figure 14 plots the shrinkage, and Table 11 lists the vendors at both ends.

Yifan Wireless, with 13 Critical CVEs out of 13, keeps a posterior of 0.82. Shrinkage costs it 18 points but leaves it first. TCL, IP-COM and Canon follow with posteriors near three-quarters, and the building-automation vendor Johnson Controls comes fifth. At the other end, the chipset vendors are almost never Critical. Unisoc has one Critical CVE in 634 and MediaTek 31 in 950, because their bulletins score most flaws as local with low privileges. The gap between the chipset vendors and the router makers is a difference in attack surface as much as in code quality: a modem bug reached

Table 10. Cox model for time from NVD publication to KEV listing, enriched CVEs

Covariate	Hazard ratio	95% CI	p
Device CVE (unadjusted model)	1.17	0.95–1.45	0.149
Device CVE (adjusted model)	0.61	0.49–0.77	< 0.001
Critical (disclosure score)	10.42	7.95–13.66	< 0.001
High (disclosure score)	3.93	3.13–4.95	< 0.001
Attack vector network	0.94	0.77–1.14	0.514
No privileges required	1.13	0.93–1.36	0.216
No user interaction	0.83	0.67–1.02	0.074
log(1 + CNA's prior KEV listings)	1.26	1.20–1.32	< 0.001
Publication year (2022 = 0)	0.95	0.90–1.01	0.111
CWE: memory safety	1.01	0.82–1.24	0.914
CWE: injection / code execution	2.02	1.63–2.50	< 0.001
CWE: authentication / access control	1.48	1.18–1.84	< 0.001
CWE: SQL injection	0.17	0.10–0.29	< 0.001
CWE: cross-site scripting	0.34	0.20–0.57	< 0.001

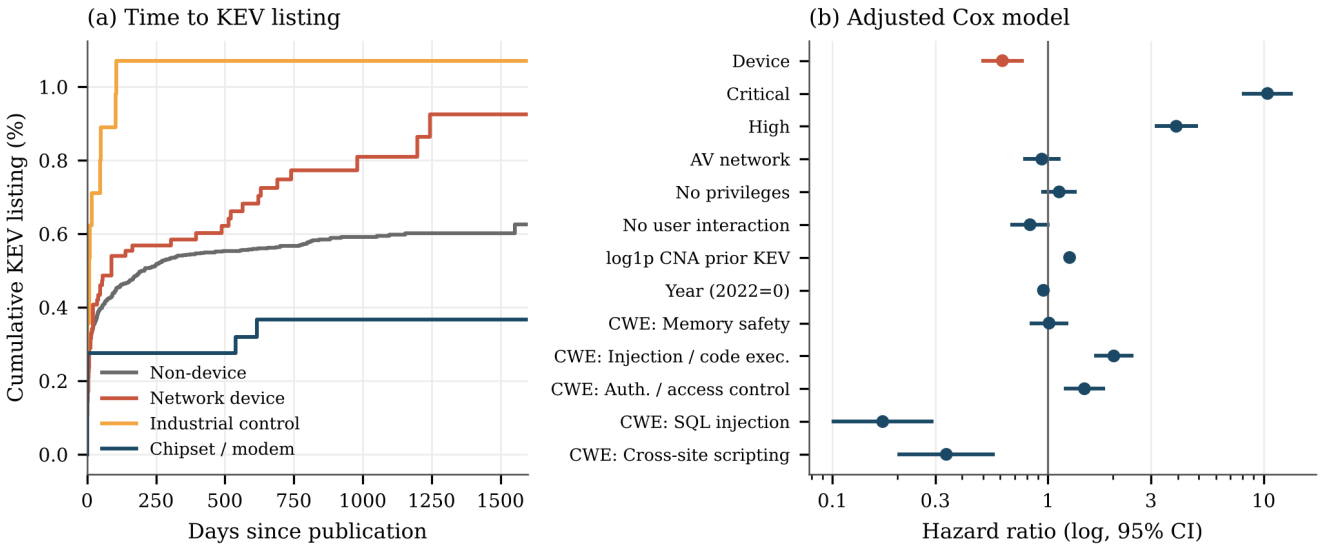


Figure 13. (a) Cumulative probability of KEV listing (one minus Kaplan–Meier) for non-device CVEs and three device categories, enriched CVEs. (b) Hazard ratios from the adjusted Cox model with 95% confidence intervals (log scale).

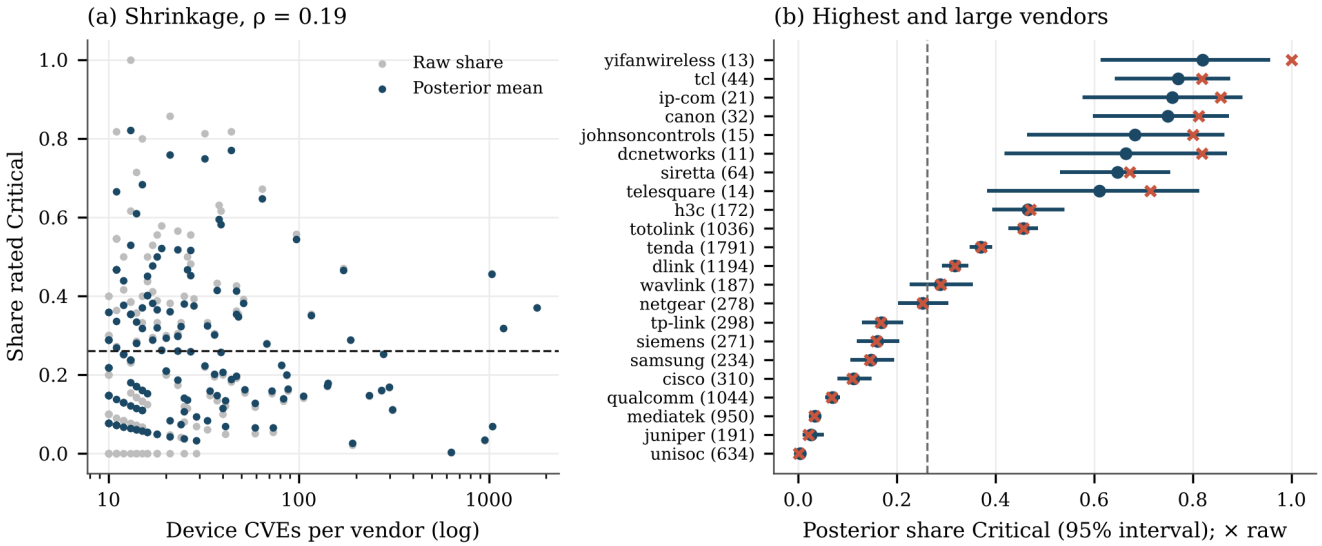


Figure 14. (a) Raw and posterior Critical shares by vendor against the number of scored device CVEs (log scale); posterior means are pulled toward the prior mean, most strongly for small vendors. (b) Vendors with the highest and lowest posterior Critical shares, with 95% posterior intervals; crosses mark the raw shares.

Table 11. Posterior Critical shares for vendors at both ends of the distribution (CPE vendor names)

Vendor	Critical / scored	Raw share	Posterior mean	95% interval
yifanwireless	13 / 13	1.000	0.820	0.614–0.958
tcl	36 / 44	0.818	0.770	0.643–0.876
ip-com	18 / 21	0.857	0.758	0.577–0.901
canon	26 / 32	0.812	0.749	0.598–0.874
johnsoncontrols	12 / 15	0.800	0.683	0.465–0.865
siretta	43 / 64	0.672	0.647	0.530–0.755
mediatek	31 / 950	0.033	0.034	0.023–0.046
jvckenwood	0 / 29	0.000	0.033	0.001–0.114
juniper	4 / 191	0.021	0.026	0.009–0.053
unisoc	1 / 634	0.002	0.003	0.000–0.009

through an app on the handset cannot score as high as an unauthenticated flaw in a router's web server.

6. DISCUSSION

6.1 What the record says about IoT devices

Table 12 gathers the main findings of the five analyses. Three of them carry practical weight.

The first concerns weakness types. The policy baselines for IoT devices [7, 8] and the Cyber Resilience Act's essential requirements [6] lead with credentials, authentication and update mechanisms, and those are real problems: Mirai spread through default passwords [1], and home-network measurements find many devices exposing weak services [2, 41]. The disclosed flaws point somewhere else. Authentication weaknesses are no more common among device CVEs than among other CVEs, whereas memory-safety bugs and command injection are three times as common and together make up most of the device record. This agrees with firmware analyses that find memory corruption at the centre of gateway vulnerabilities [18] and with reviews of firmware auditing [5, 17], and it points to two concrete engineering remedies: memory-safe languages or hardened allocators for network-facing daemons, and the removal of shell calls from web handlers. Neither appears by name in the current baselines.

The second concerns the enrichment gap. NVD's backlog [9] and its decision to concentrate on KEV entries [10] mean that a product-based query of NVD now misses roughly a quarter of the 2025 device CVEs and more than a third of those from 2026. Manufacturers and regulators who count device vulnerabilities from CPE data, as the Cyber Resilience Act's reporting regime will encourage, will undercount, and the undercount will be largest for industrial control advisories, which CISA itself issued and which NVD often left unenriched. A text classifier recovers the count well enough for aggregate use. It is less suitable for labelling single records, where its 17% miss rate would matter.

The third concerns severity. Device CVEs score higher than other CVEs, yet with severity held fixed they reach the KEV catalog less often. Two readings fit the hazard ratio of 0.61. The first is that scores for device CVEs are inflated: reporters who file twenty overflows against one router's web server, each with a 9.8, describe the same exploitable condition twenty times, and attackers need only one of them. The second is that exploitation of devices is under-observed. KEV listing requires evidence of exploitation that CISA can verify, and botnet operators who exploit routers en masse rarely produce the incident reports that enterprise intrusions do. The data cannot separate the two. A comparison of KEV listings with exploitation telemetry from honeypots, or with the Exploit Prediction Scoring System [42], would, because botnet traffic is visible to sensors even when it never reaches a CISA advisory. Either way, the practical conclusion for a fleet operator is the same: a 9.8 on a router CVE carries less information about imminent attack than a 9.8 elsewhere, and the latent classes together with the injection family are better guides. Within the device record, C1 holds 56% of KEV listings, and command-injection flaws double the hazard while memory-safety flaws leave it unchanged.

6.2 Templates as a measurement problem

Reporting templates turned up in every text model we fitted. They inflate topic coherence, they give the classifier some of its strongest features, and they account for the fastest-growing topic. None of this is wrong in itself. A VulDB sentence saying that the vendor did not respond is information about the disclosure process, and its rise over 2024–2026 records a real change: a growing share of device flaws are reported through a third-party CNA to vendors who do not engage. The measurement problem is that text-based analyses of the CVE record, from topic models to severity predictors [14], partly learn who wrote a description and only partly what it describes. We stripped known boilerplate before topic modelling. The coherence penalty that followed, from 0.58 to 0.37, gives a rough measure of how much of the apparent structure was template. Studies that train on CVE text should report how they handled CNA templates, and data publishers could help by separating structured fields such as patch identifiers and vendor response status from the free-text description.

6.3 Implications

For manufacturers, the evidence supports investment in memory-safe network services and in the removal of command interpreters from configuration handlers, and it supports answering vulnerability reports: about a third of 2026 device CVEs came through VulDB, and many of those carry the note that the vendor never responded. For regulators, the rise of end-of-life language matters for the Cyber Resilience Act's support-period requirement [6], because devices whose vendors have stopped issuing fixes account for a growing share of new device CVEs. Most come from a few consumer router

Table 12. Summary of findings by analysis

Analysis	Finding	Evidence
Volume and reporters	About one enriched CVE in nine concerns a device; the share falls about 10% a year; VulDB issued a third of device CVEs in 2026	Figures I–II
Severity and weakness	Critical share almost twice the rest; memory safety and command injection have odds ratios near 3; Gini of vendors 0.85	Tables III–IV, Figures III–V
Attack profiles	Eight classes; one in three is a remote, unauthenticated, full-impact flaw, and that class holds 56% of device KEV listings	Tables V–VI, Figures VI–VII
Hidden device CVEs	7.2% of unenriched CVEs are devices; the 2025 count rises by a third; the share trend survives the correction	Tables VII–VIII, Figures VIII–X
Exploitation and vendors	Adjusted listing hazard 0.61; end-of-life language rises to 4.8%; strong between-vendor variation in Critical shares	Tables IX–XI, Figures XI–XIV

brands. For fleet operators, the results argue for prioritising device CVEs by attack profile and weakness family alongside severity. An end-of-life note should be read as a replacement order.

6.4 Limitations

The device label depends on NVD's CPE practice. Hardware and firmware entries are not applied uniformly, so some device CVEs are filed only under application CPEs and are missed, and the vendor exclusion list is a judgement that moves server and PC firmware out of scope. A different list would change the totals but, given how few excluded vendors appear among routers, cameras and controllers, probably not the comparisons. Categories rest on vendor lists and keywords and were not validated against manual labels. The camera and printer estimates are small and fragile. The quantification assumes that descriptions of device CVEs look the same in enriched and unenriched records, which is plausible for the CNAs that dominate the hidden device records and untested for the others; a labelled audit sample of unenriched records would test it directly. The Cox model is restricted to enriched CVEs. Because NVD enriches KEV entries first, every listed CVE is enriched while many unlisted CVEs from 2024–2026 are not, which raises the apparent listing rate among enriched records in those years for device and non-device CVEs alike; the year term absorbs part of this, but the hazard ratio is conditional on enrichment. Finally, KEV listing is a conservative proxy for exploitation, and Section 6.1 has argued that it may be more conservative for devices than for other products.

7. CONCLUSION

We separated IoT and embedded device vulnerabilities from the rest of the National Vulnerability Database with a reproducible rule on product enumerations and studied them with five models, from descriptive comparisons to a survival model of exploitation. Device CVEs are more severe on paper than other CVEs. They are dominated by memory-safety and command-injection flaws in a few router and chipset vendors. Eight latent attack profiles describe their CVSS vectors. One of them, the remote unauthenticated full-impact flaw, holds most of their exploitation. A text classifier with adjusted classify-and-count recovers device CVEs that NVD never enriched and finds that product-based counts now miss a quarter or more of them, while the decline in the device share is real. Text models are shaped by CNA templates. They also record a fast rise in flaws reported against products their vendors no longer support. With severity held fixed, device CVEs reach the KEV catalog less often than other CVEs, and whether that reflects inflated scores or under-observed botnet exploitation is the question that follow-up work with sensor data should answer.

REFERENCES

[1] M. Antonakakis et al., "Understanding the Mirai botnet," in *Proc. 26th USENIX Security Symp.*, pp. 1093–1110, 2017.

[2] D. Kumar et al., "All things considered: An analysis of IoT devices on home networks," in *Proc. 28th USENIX Security Symp.*, pp. 1169–1185, 2019.

[3] N. Neshenko, E. Bou-Harb, J. Crichigno, G. Kaddoum, and N. Ghani, "Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look on Internet-scale IoT exploitations," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2702–2733, 2019.

[4] F. Meneghello, M. Calore, D. Zucchetto, M. Polese, and A. Zanella, "IoT: Internet of threats? A survey of practical security vulnerabilities in real IoT devices," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8182–8201, 2019.

[5] T. Bakhshi, B. Ghita, and I. Kuzminykh, "A review of IoT firmware vulnerabilities and auditing techniques," *Sensors*, vol. 24, no. 2, Art. no. 708, 2024.

[6] European Parliament and Council of the European Union, "Regulation (EU) 2024/2847 of 23 October 2024 on horizontal cybersecurity requirements for products with digital elements (Cyber Resilience Act)," *Official Journal of the European Union*, L, 2024/2847, 2024.

[7] European Telecommunications Standards Institute, *Cyber Security for Consumer Internet of Things: Baseline Requirements*, ETSI EN 303 645 V2.1.1, 2020.

[8] M. Fagan, K. N. Megas, K. Scarfone, and M. Smith, *Foundational Cybersecurity Activities for IoT Device Manufacturers*, NIST Interagency Rep. 8259, 2020.

[9] U.S. Department of Commerce Office of Inspector General, *Evaluation of NIST's Management of the National Vulnerability Database*, Rep. OIG-26-020-I, 2026.

[10] National Institute of Standards and Technology, "NIST updates NVD operations to address record CVE growth," news release, Apr. 15, 2026.

[11] A. Costin, J. Zaddach, A. Francillon, and D. Balzarotti, "A large-scale analysis of the security of embedded firmwares," in *Proc. 23rd USENIX Security Symp.*, pp. 95–110, 2014.

[12] D. D. Chen, M. Woo, D. Brumley, and M. Egele, "Towards automated dynamic analysis for Linux-based embedded firmware," in *Proc. Network and Distributed System Security Symp.*, 2016.

[13] A. Anwar, A. Abusnaina, S. Chen, F. Li, and D. Mohaisen, "Cleaning the NVD: Comprehensive quality assessment, improvements, and analyses," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 6, pp. 4255–4269, 2022.

[14] T. H. M. Le, H. Chen, and M. A. Babar, "A survey on data-driven software vulnerability assessment and prioritization," *ACM Comput. Surveys*, vol. 55, no. 5, Art. no. 100, 2022.

[15] X. Feng, Q. Li, H. Wang, and L. Sun, "Acquisitional rule-based engine for discovering Internet-of-Things devices," in *Proc. 27th USENIX Security Symp.*, pp. 327–341, 2018.

[16] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A survey on IoT security: Application areas, security threats, and solution architectures," *IEEE Access*, vol. 7, pp. 82721–82743, 2019.

[17] I. Nadir, H. Mahmood, and G. Asadullah, "A taxonomy of IoT firmware security and principal firmware analysis techniques," *Int. J. Critical Infrastructure Protection*, vol. 38, Art. no. 100552, 2022.

[18] V. Safronov, A. Bostan, N. Allott, and A. Martin, "How memory-safe is IoT? Assessing the impact of memory-protection solutions for securing wireless gateways," arXiv:2411.01377, 2024.

[19] Cybersecurity and Infrastructure Security Agency, *Binding Operational Directive 22-01: Reducing the Significant Risk of Known Exploited Vulnerabilities*, U.S. Department of Homeland Security, 2021.

[20] Forum of Incident Response and Security Teams, *Common Vulnerability Scoring System Version 3.1: Specification Document*, 2019.

[21] P. F. Lazarsfeld and N. W. Henry, *Latent Structure Analysis*. Boston, MA, USA: Houghton Mifflin, 1968.

[22] L. A. Goodman, "Exploratory latent structure analysis using both identifiable and unidentifiable models," *Biometrika*, vol. 61, no. 2, pp. 215–231, 1974.

[23] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Stat. Soc. B*, vol. 39, no. 1, pp. 1–38, 1977.

[24] G. Schwarz, "Estimating the dimension of a model," *Ann. Statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[25] K. L. Nylund, T. Asparouhov, and B. O. Muthén, "Deciding on the number of classes in latent class analysis and growth mixture modeling: A Monte Carlo simulation study," *Structural Equation Modeling*, vol. 14, no. 4, pp. 535–569, 2007.

[26] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing & Management*, vol. 24, no. 5, pp. 513–523, 1988.

[27] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*, pp. 61–74, 1999.

[28] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[29] G. Forman, "Quantifying counts and costs via classification," *Data Mining and Knowledge Discovery*, vol. 17, no. 2, pp. 164–206, 2008.

[30] P. González, A. Castaño, N. V. Chawla, and J. J. del Coz, "A review on quantification learning," *ACM Comput. Surveys*, vol. 50, no. 5, Art. no. 74, 2017.

[31] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[32] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems 13*, pp. 556–562, 2001.

[33] G. Bouma, "Normalized (pointwise) mutual information in collocation extraction," in *Proc. Biennial GSCL Conf.*, pp. 31–40, 2009.

[34] M. Röder, A. Both, and A. Hinneburg, "Exploring the space of topic coherence measures," in *Proc. 8th ACM Int. Conf. Web Search and Data Mining*, pp. 399–408, 2015.

[35] E. L. Kaplan and P. Meier, "Nonparametric estimation from incomplete observations," *J. Amer. Stat. Assoc.*, vol. 53, no. 282, pp. 457–481, 1958.

[36] D. R. Cox, "Regression models and life-tables," *J. Royal Stat. Soc. B*, vol. 34, no. 2, pp. 187–220, 1972.

[37] C. Davidson-Pilon, "lifelines: Survival analysis in Python," *J. Open Source Softw.*, vol. 4, no. 40, Art. no. 1317, 2019.

[38] D. A. Williams, "The analysis of binary responses from toxicological experiments involving reproduction and teratogenicity," *Biometrics*, vol. 31, no. 4, pp. 949–952, 1975.

[39] B. Efron and C. Morris, "Data analysis using Stein's estimator and its generalizations," *J. Amer. Stat. Assoc.*, vol. 70, no. 350, pp. 311–319, 1975.

[40] E. B. Wilson, "Probable inference, the law of succession, and statistical inference," *J. Amer. Stat. Assoc.*, vol. 22, no. 158, pp. 209–212, 1927.

[41] O. Alrawi, C. Lever, M. Antonakakis, and F. Monrose, "SoK: Security evaluation of home-based IoT deployments," in *Proc. 2019 IEEE Symp. Security and Privacy*, pp. 1362–1380, 2019.

[42] J. Jacobs, S. Romanosky, B. Edwards, I. Adjerid, and M. Roytman, "Exploit prediction scoring system (EPSS)," *Digital Threats: Research and Practice*, vol. 2, no. 3, Art. no. 20, 2021.