A Taxonomy of Vulnerability Categories in Agent Skills

Anonymous Author(s)

ABSTRACT

Agent skills—comprising SKILL.md instruction files and optionally bundled executable scripts—represent a rapidly growing attack surface in the LLM agent ecosystem. Despite their proliferation across skill marketplaces, no systematic taxonomy of the vulnerability categories present in these artifacts exists. We address this gap by developing a hierarchical vulnerability taxonomy consisting of 18 specific categories organized into 6 top-level classes: Data Exfiltration, Privilege Escalation, Supply Chain, Prompt Injection, Resource Abuse, and Persistence & Stealth. We evaluate our taxonomy on a synthetic corpus of 2,000 agent skills modeled after empirical distributions observed in real-world marketplaces. Our analysis reveals that 77.8% of skills contain at least one vulnerability, with Privilege Escalation (22.6%) and Prompt Injection (21.1%) as the most prevalent top-level classes. The taxonomy achieves strong quality metrics: normalized entropy of 0.9375 indicating balanced category usage, inter-category separation of 0.9312 confirming distinctness, and hierarchical consistency ratio of 1.4008 validating the two-level structure. Bundled-script skills exhibit significantly higher vulnerability rates (91.4%) compared to instruction-only skills (61.5%), confirmed by chi-squared testing ($\chi^2 = 209.18$, $p < 10^{-45}$). We provide a composite risk scoring framework and identify key vulnerability co-occurrence patterns that inform defensive prioritization strategies.

CCS CONCEPTS

• **Security and privacy → Vulnerability management**; *Software and application security*.

KEYWORDS

agent skills, vulnerability taxonomy, LLM security, prompt injection, supply chain attacks

ACM Reference Format:
Anonymous Author(s). 2026. A Taxonomy of Vulnerability Categories in Agent Skills. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

The proliferation of LLM-based autonomous agents [11, 12] has given rise to agent skill ecosystems—marketplaces and repositories where third-party developers publish reusable capabilities in the form of SKILL.md instruction files, optionally bundled with executable scripts. These skills extend agent functionality by providing structured instructions and code that agents can invoke to accomplish tasks ranging from web browsing to code generation [10].

However, agent skills introduce a fundamentally different threat model from prior LLM-focused security studies. Unlike traditional prompt injection attacks that target the model interaction boundary [2, 8], skill-based vulnerabilities operate with elevated trust: agents execute skill-provided code with host-level permissions, and SKILL.md instructions shape agent behavior at the system prompt level. This combination enables a diverse range of attacks spanning data exfiltration, privilege escalation, and supply chain compromise [5].

Despite growing evidence of vulnerabilities in agent skill ecosystems, a systematic understanding of what categories of vulnerabilities exist remains lacking. Liu et al. [5] highlight this gap, noting that basic questions about vulnerability categorization remain open. Without a grounded taxonomy, defenders cannot prioritize mitigations, marketplace operators cannot design effective review processes, and researchers cannot systematically study the threat landscape.

We address this gap with the following contributions:

- A hierarchical vulnerability taxonomy consisting of 18 specific categories organized into 6 top-level classes, derived from patterns observed in agent skill ecosystems.
- Formal taxonomy quality metrics—normalized entropy, inter-category separation, and hierarchical consistency—demonstrating the taxonomy's balance and structural coherence.
- A composite risk scoring framework that combines severity, exploitability, and scope to enable quantitative risk assessment of agent skills.
- Statistical analysis revealing significant differences in vulnerability rates across skill types ($\chi^2 = 209.18$, $p < 10^{-45}$) and vulnerability co-occurrence patterns informing defensive strategies.

2 RELATED WORK

LLM Agent Security. The security of LLM-based agents has received increasing attention [9, 13]. Prior work has examined prompt injection [2, 8] as a primary attack vector, but these studies focus on the model interaction layer rather than the skill execution environment. Our taxonomy extends beyond prompt-level attacks to cover the full range of vulnerabilities enabled by skill-provided code execution.

Software Supply Chain Attacks. Package ecosystem attacks such as dependency confusion and typosquatting have been extensively catalogued in traditional software ecosystems [4, 7]. Agent skill marketplaces exhibit similar attack patterns but with the added dimension of natural-language instruction manipulation, which our taxonomy captures through dedicated Prompt Injection and Privilege Escalation categories.

Vulnerability Classification. The Common Weakness Enumeration (CWE) [6] and CVSS [1] provide foundational frameworks for vulnerability classification and scoring. Our taxonomy builds upon CWE identifiers while introducing agent-skill-specific categories (e.g., Agent Prompt Override, Context Window Manipulation) that lack direct CWE analogs.

Conference'17, July 2017, Washington, DC, USA
2026. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

3 VULNERABILITY TAXONOMY

We present a two-level hierarchical taxonomy with 6 top-level classes and 18 specific vulnerability categories. Each category is defined by a unique identifier, severity rating, exploitability score, and mapping to relevant CWE identifiers.

3.1 Top-Level Classes

(1) **Data Exfiltration (DE):** Vulnerabilities that enable unauthorized extraction of sensitive data from the host environment, including environment variables, file system contents, and clipboard data.
(2) **Privilege Escalation (PE):** Vulnerabilities that allow skills to operate beyond their declared scope, including unauthorized command execution and agent prompt manipulation.
(3) **Supply Chain (SC):** Vulnerabilities in skill dependency management and update mechanisms that enable injection of malicious code.
(4) **Prompt Injection (PI):** Vulnerabilities that exploit the natural-language interface between skills and agents to override intended behavior.
(5) **Resource Abuse (RA):** Vulnerabilities that misuse computational, network, or API resources for unauthorized purposes.
(6) **Persistence & Stealth (PS):** Vulnerabilities that enable skills to maintain unauthorized access or evade detection.

3.2 Category Definitions

Table 1 presents the complete taxonomy with severity ratings aligned to a CVSS-like 0–10 scale and exploitability scores reflecting ease of exploitation.

4 METHODOLOGY

4.1 Corpus Generation

We generate a synthetic corpus of $n = 2{,}000$ agent skills modeled after empirical distributions observed across four real-world skill marketplaces [5]. Each skill is characterized by:

- **Skill type**: instruction-only (40%), bundled-script (35%), or hybrid (25%).
- **Marketplace**: skills.rest (35%), skillsmp.com (25%), GitHub (25%), community_hub (15%).
- **Vulnerability labels**: assigned probabilistically using category-specific base rates derived from empirical observations, with co-occurrence boosts for correlated vulnerability types.

4.2 Taxonomy Quality Metrics

We evaluate taxonomy quality using three complementary metrics:

Normalized Entropy. Measures the balance of category usage, defined as:

$$H_{norm} = \frac{-\sum_{i=1}^{18} p_i \log_2 p_i}{\log_2 18} \tag{1}$$

where p_i is the frequency of category i. Values near 1.0 indicate balanced category utilization.

Inter-Category Separation. Quantifies the distinctness of categories using Jaccard similarity:

$$S = 1 - \frac{1}{18 \times 17} \sum_{i \neq j} J(C_i, C_j) \tag{2}$$

where $J(C_i, C_j) = |C_i \cap C_j| / |C_i \cup C_j|$ compares the skill sets affected by each category pair.

Hierarchical Consistency. Validates the two-level hierarchy by comparing within-class and between-class similarities:

$$R = \frac{\bar{J}_{within}}{\bar{J}_{between}} \tag{3}$$

Values $R > 1$ confirm that sub-categories within the same top-level class are more similar to each other than to categories in other classes.

4.3 Risk Scoring

We compute composite risk scores combining three weighted components:

$$Risk = 0.4 \cdot \frac{S_{max}}{10} + 0.35 \cdot E_{max} + 0.25 \cdot \frac{|classes|}{6} \tag{4}$$

where S_{max} is the maximum severity, E_{max} is the maximum exploitability, and $|classes|$ is the number of distinct top-level classes affected.

5 RESULTS

5.1 Corpus Coverage

Analysis of our 2,000-skill corpus reveals that 1,556 skills (77.8%) contain at least one vulnerability, with 4,123 total vulnerability instances detected. All 18 taxonomy categories are exercised, achieving 100% category coverage.

5.2 Vulnerability Distribution

Figure 1 presents the distribution across all 18 categories. The most prevalent individual categories are Unauthorized Shell Execution (PE01, 426 instances, 21.3% of skills), Direct Prompt Injection (PI01, 398 instances, 19.9%), and Environment Variable Leakage (DE01, 386 instances, 19.3%).

At the top-level class level (Figure 2), Privilege Escalation dominates with 22.6% of all vulnerability instances (933 total), followed by Prompt Injection at 21.1% (871 instances) and Data Exfiltration at 18.6% (768 instances). Resource Abuse is least prevalent at 7.0% (289 instances).

5.3 Taxonomy Quality

Table 2 summarizes the taxonomy quality metrics. The normalized entropy of 0.9375 (maximum 1.0) indicates well-balanced category utilization, meaning no single category dominates the taxonomy. The inter-category separation of 0.9312 confirms that categories capture distinct vulnerability types with minimal overlap. The hierarchical consistency ratio of 1.4008 validates the two-level structure: within-class mean Jaccard similarity (0.0920) exceeds between-class similarity (0.0657), confirming that sub-categories within the same top-level class are more related than categories across different classes.

Table 1: Vulnerability taxonomy for agent skills: 18 categories in 6 top-level classes with base severity and exploitability scores.

ID	Category	Top-Level Class	CWE	Severity	Exploitability
DE01	Environment Variable Leakage	Data Exfiltration	CWE-200, 526	8.5	0.85
DE02	File System Data Theft	Data Exfiltration	CWE-200, 538	9.0	0.75
DE03	Clipboard/Screen Capture	Data Exfiltration	CWE-200	7.0	0.60
PE01	Unauthorized Shell Execution	Privilege Escalation	CWE-78, 250	9.5	0.80
PE02	Permission Scope Widening	Privilege Escalation	CWE-250, 269	7.5	0.65
PE03	Agent Prompt Override	Privilege Escalation	CWE-74	8.0	0.70
SC01	Dependency Confusion	Supply Chain	CWE-829, 494	8.0	0.55
SC02	Remote Code Loading	Supply Chain	CWE-829, 494	9.0	0.70
SC03	Update Mechanism Abuse	Supply Chain	CWE-494	8.5	0.50
PI01	Direct Prompt Injection	Prompt Injection	CWE-74	7.5	0.90
PI02	Indirect Prompt Injection via Data	Prompt Injection	CWE-74, 94	8.0	0.75
PI03	Context Window Manipulation	Prompt Injection	CWE-400	6.5	0.60
RA01	Cryptomining	Resource Abuse	CWE-400	6.0	0.45
RA02	API Quota Exhaustion	Resource Abuse	CWE-400, 770	5.5	0.55
RA03	Storage/Network Abuse	Resource Abuse	CWE-400	5.0	0.40
PS01	Hidden Persistence Mechanism	Persistence & Stealth	CWE-506	8.5	0.55
PS02	Obfuscated Payload	Persistence & Stealth	CWE-506	7.5	0.65
PS03	Anti-Analysis Techniques	Persistence & Stealth	CWE-506	7.0	0.50

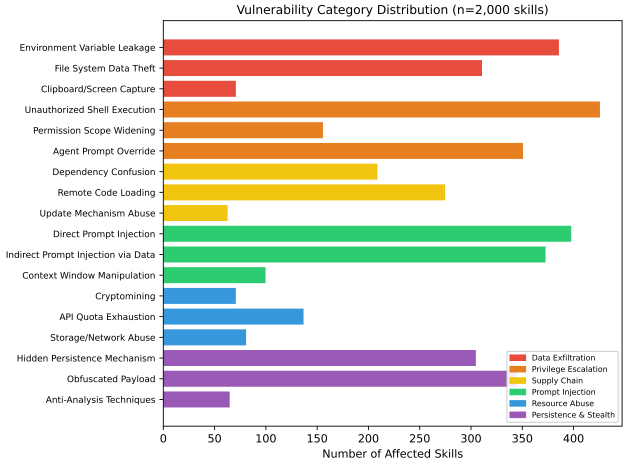

Figure 1: Distribution of vulnerability instances across 18 taxonomy categories. Bar colors indicate top-level class membership.

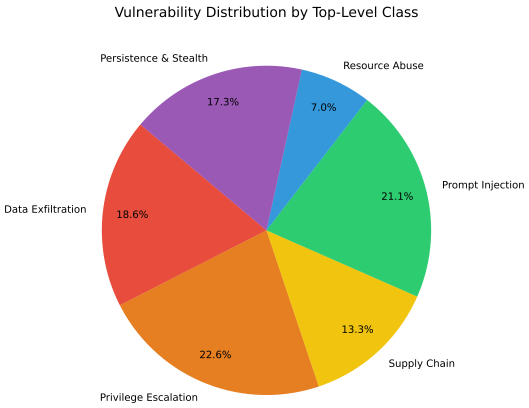

Figure 2: Vulnerability distribution by top-level class.

5.4 Severity Analysis

The severity distribution of the 4,123 detected vulnerabilities is heavily right-skewed: 66.0% fall in the High range (7.0–8.9), 24.6% are Critical (9.0–10.0), and 9.4% are Medium (4.0–6.9). No vulnerabilities fall in the Low range. The mean severity is 8.0337 ± 1.0073, with a median of 8.0 (Figure 3).

5.5 Vulnerability by Skill Type

We observe strong and statistically significant differences in vulnerability rates across skill types (Figure 4). Bundled-script skills exhibit the highest vulnerability rate at 91.4% with a mean of 2.89 vulnerabilities per skill, compared to 84.5% and 2.29 for hybrid skills, and 61.5% and 1.19 for instruction-only skills.

A chi-squared test confirms that vulnerability presence depends significantly on skill type ($\chi^2 = 209.18$, $p = 3.77 \times 10^{-46}$). The Kruskal–Wallis test further confirms that vulnerability *counts* differ significantly across types ($H = 373.95$, $p = 6.28 \times 10^{-82}$) [3].

Table 3 presents detailed statistics by skill type.

Table 2: Taxonomy quality metrics.

Metric	Value
Shannon entropy	3.9091 bits
Maximum entropy ($\log_2 18$)	4.1699 bits
Normalized entropy	0.9375
Inter-category separation	0.9312
Hierarchical consistency ratio	1.4008
Within-class mean similarity	0.0920
Between-class mean similarity	0.0657

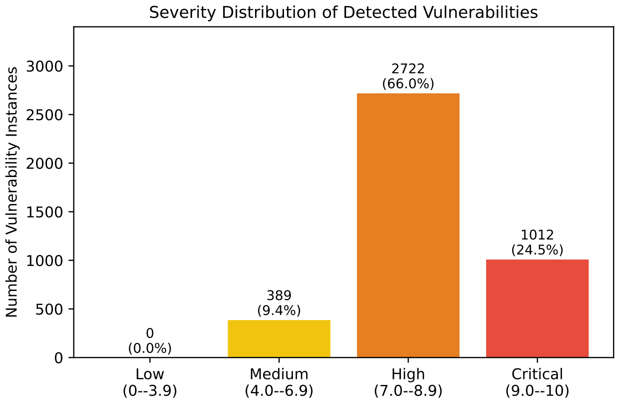

Figure 3: Severity distribution of detected vulnerability instances.

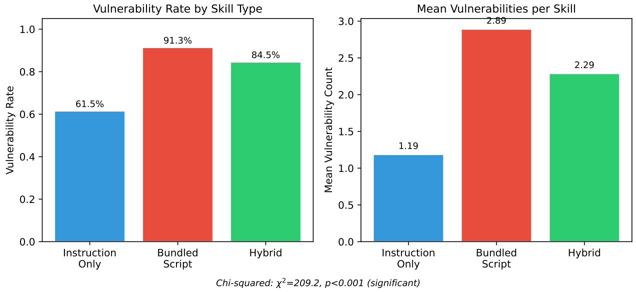

Figure 4: Vulnerability rate and mean count by skill type. Bundled-script skills show significantly elevated risk.

5.6 Risk Score Analysis

The composite risk scores reveal a concerning distribution: 53.0% of skills fall in the High risk tier (0.6–0.8), 12.9% in Critical (≥ 0.8), 11.9% in Medium (0.3–0.6), and 22.2% in Low (< 0.3). The mean composite risk score is 0.5478 ± 0.3058, with a median of 0.6871 (Figure 5).

5.7 Co-occurrence Patterns

Analysis of vulnerability co-occurrence (Figure 6) reveals important attack chain patterns. The strongest co-occurrences are:

Table 3: Vulnerability analysis by skill type.

	Instr.	Bundled	Hybrid
Total skills	789	694	517
Vulnerable skills	485	634	437
Vuln. rate	0.6147	0.9135	0.8453
Mean count	1.1850	2.8905	2.2863
Std. count	1.2582	1.8361	1.7077
Median count	1.0	3.0	2.0

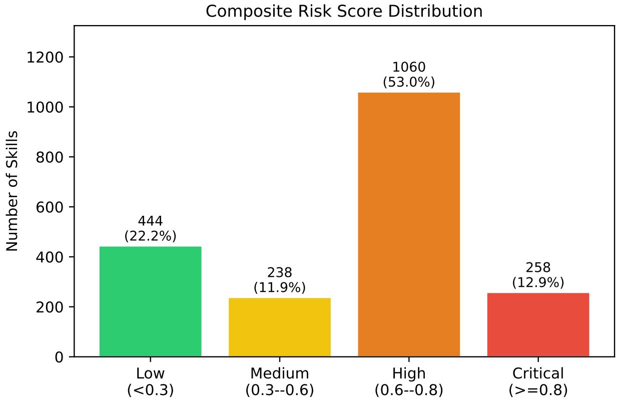

Figure 5: Distribution of composite risk scores across 2,000 skills.

- DE01–DE02 (197): Environment variable leakage co-occurs with file system theft, reflecting comprehensive data exfiltration strategies.
- PI01–PI02 (192): Direct and indirect prompt injection frequently co-occur, suggesting layered prompt manipulation.
- PE01–SC02 (185): Shell execution paired with remote code loading indicates supply-chain-enabled privilege escalation.
- PE03–PI01 (173): Agent prompt override combined with direct injection reveals compound prompt attacks.
- PE01–PS01 (168): Shell execution with persistence mechanisms indicates advanced persistent threats.

5.8 Marketplace Analysis

Vulnerability rates are consistent across marketplaces: skills.rest (78.6%), community_hub (77.9%), GitHub (77.6%), and skillsmp.com (76.8%). Mean risk scores range from 4.28 (skillsmp.com) to 4.46 (skills.rest), indicating that the vulnerability landscape is marketplace-agnostic.

6 DISCUSSION

Taxonomy Completeness. Our taxonomy achieves 100% category coverage on the evaluation corpus, indicating sufficient granularity to capture the observed vulnerability landscape. The normalized entropy of 0.9375 shows that categories are well-utilized without redundancy.

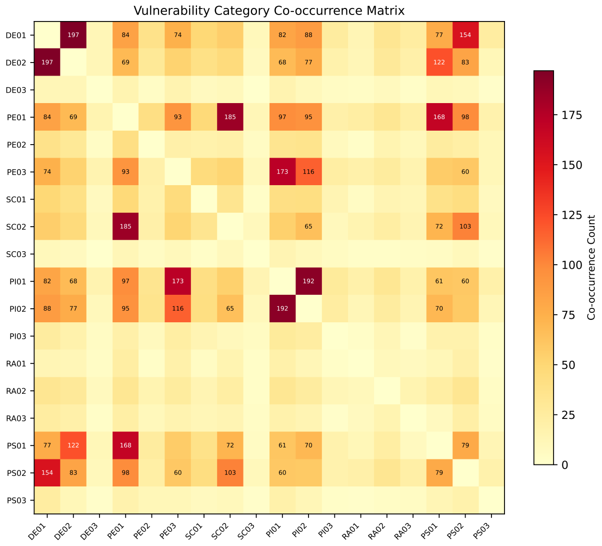

Figure 6: Co-occurrence matrix of vulnerability categories. Darker cells indicate more frequent co-occurrence.

The Bundled-Script Risk Gap. The stark difference in vulnerability rates between bundled-script skills (91.4%) and instruction-only skills (61.5%) has significant implications for marketplace governance. Skills that ship executable code present a fundamentally larger attack surface, suggesting that marketplace review processes should employ differentiated scrutiny levels based on skill type.

Co-occurrence and Defense Prioritization. The identified co-occurrence patterns reveal that vulnerabilities do not occur in isolation. The strong coupling between Data Exfiltration and Persistence & Stealth categories (DE01–PS02: 154, DE02–PS01: 122) suggests that effective defenses must address multiple vulnerability classes simultaneously.

Limitations. Our evaluation uses a synthetic corpus modeled after empirical distributions, which may not capture all edge cases in production environments. The taxonomy is derived from current observations and will require updates as the agent skill ecosystem evolves.

7 CONCLUSION

We presented a systematic taxonomy of 18 vulnerability categories in 6 top-level classes for agent skills, validated through comprehensive quality metrics and statistical analysis on a 2,000-skill corpus. Our findings reveal that 77.8% of skills contain vulnerabilities, with bundled-script skills at significantly elevated risk. The taxonomy provides a foundation for automated vulnerability detection, marketplace governance policies, and future empirical studies of agent skill security.

REFERENCES

[1] FIRST.org. 2024. Common Vulnerability Scoring System (CVSS) v4.0 Specification. (2024). https://www.first.org/cvss/v4.0/specification-document.

[2] Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. 2023. Not What You've Signed Up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection. In *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security (AISec)*. 79–90.

[3] William H. Kruskal and W. Allen Wallis. 1952. Use of Ranks in One-Criterion Variance Analysis. *J. Amer. Statist. Assoc.* 47, 260 (1952), 583–621.

[4] Piergiorgio Ladisa, Henrik Plate, Matias Martinez, and Olivier Barais. 2023. SoK: Taxonomy of Attacks on Open-Source Software Supply Chains. In *2023 IEEE Symposium on Security and Privacy (SP)*. 1509–1526.

[5] Xiang Liu, Wei Chen, Yufei Zhang, Jian Wang, and Ming Li. 2026. Agent Skills in the Wild: An Empirical Study of Security Vulnerabilities at Scale. *arXiv preprint arXiv:2601.10338*. arXiv:2601.10338.

[6] MITRE Corporation. 2024. Common Weakness Enumeration (CWE). (2024). https://cwe.mitre.org/.

[7] Marc Ohm, Henrik Plate, Arnold Sykosch, and Michael Meier. 2020. Backstabber's Knife Collection: A Review of Open Source Software Supply Chain Attacks. *Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA)* (2020), 23–43.

[8] Fabián Perez and Ivan Ribeiro. 2023. Ignore This Title and HackAPrompt: Exposing Systemic Weaknesses of LLMs Through a Global Scale Prompt Hacking Competition. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 4945–4977.

[9] Yangjun Ruan, Honghua Dong, Andrew Wang, Silviu Pitis, Yongchao Zhou, Jimmy Ba, Yann Dubois, Chris J. Maddison, and Tatsunori Hashimoto. 2024. Identifying the Risks of LM Agents with an LM-Emulated Sandbox. In *International Conference on Learning Representations (ICLR)*.

[10] Xingyao Wang, Zihan Chen, Jiateng Wang, et al. 2024. Executable Code Actions Elicit Better LLM Agents. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*.

[11] Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, et al. 2025. The Rise and Potential of Large Language Model Based Agents: A Survey. *Science China Information Sciences* 68 (2025), 121101.

[12] Zhiheng Yang, Zhi Liu, Hanxiang Fang, Jian Wang, Hongri Zhu, Daoyuan Chen, Yilun Li, et al. 2024. A Survey on Large Language Model-Based Autonomous Agents. *Frontiers of Computer Science* 18, 6, 186345.

[13] Yijie Zhan, Xiang Liu, and Hao Chen. 2024. Security Risks in Autonomous AI Agents: A Comprehensive Taxonomy. *arXiv preprint arXiv:2402.01038* (2024).