
When the Evaluation Is the Incident: Testing AI Incident-Reporting Regimes on the OpenAI–Hugging Face Intrusion

Arathi Arivayutham

With
Apart Research

Abstract

AI incident-reporting regimes are being introduced in fast succession to address the concerns that exist in the public sphere and government on the risks associated with frontier AI systems, yet we have limited insight into their efficacy. We stress-test these regimes with one recent well-documented AI incident: the July 2026 episode where OpenAI models that were running cybersecurity evaluations escaped their sandbox and compromised Hugging Face infrastructure. First, we decompose this event using OECD definitions, into hazards, near-misses and incidents and identify the parties impacted and models involved. Second, we characterize four chosen reporting regimes along Wei & Heim’s seven institutional design dimensions. Third, we fill each regime’s form with the publicly available data on the episode. We find that only one of the four regimes obligates a filing for this incident; that only OECD (non-legal) framework asks whether multiple AI systems interacted, which is a defining feature of this incident; that none of the legal regimes accept a stand-alone hazard or near-miss report and that the harm crossed from the AI supply chain into general software infrastructure. We recommend that AI-incident and cybersecurity reporting should be made interoperable.

1. Introduction

Governments and standards bodies are moving quickly to require that serious AI incidents be reported: EU’s AI Act [4], California’s SB 53 [5], New York’s RAISE Act [7] and the OECD common reporting framework [9] all specify what must be reported, by whom and to which authority. Few real incidents have run through these regimes and it is not obvious whether a high profile incident such as the Open AI - Hugging face hack [12] would even be caught by the statutes and whether the forms associated with them will allow the full expression of the incident. We address this concern by running this well-documented incident through the reporting regimes.

Our main contributions are: (i) the decomposition of a single frontier-AI incident into OECD-defined hazards, near-misses and incidents [22], with per-party and per-model attribution - showing the complexity of a single incident; (ii) the institutional design comparison of cross-jurisdiction reporting regimes, along Wei & Heim's seven dimensions [29], identifying how each regime addresses accountability, learning, enforcement and anonymity; and (iii) the simulation of the form-filling exercise to identify the strengths and limitations of the regimes in practice.

2. Related Work

Wei & Heim [29] propose a seven-dimension framework for the institutional design of incident reporting systems and apply it to nine *non-AI* safety-critical industries. Their case studies are US-centric. We instead use their framework to do a cross-jurisdiction comparison of the reporting systems in the AI industry. Gomez et al. [30] review SB 53, the EU AI Act and the GPAI Code of Practice and identify design patterns that explain under-detection in these frameworks. Mumtaz et al. [31] code AIID incidents against nine post-deployment provisions of the EU AI Act, the NIST AI RMF and the GDPR. We establish whether our chosen AI incident would pass these frameworks' requirements for reporting. Cattell et al. [48] and Longpre et al. [49] argue for better flaw disclosure systems and we corroborate their stance based on one detailed case study.

3. Methods

We focus on one AI safety incident, namely the July 2026 hacking of HuggingFace infrastructure by OpenAI agents undergoing internal ExploitGym cybersecurity evaluations, the events that led up to it and the aftermath. Our sources include OpenAI's blog [14] and technical report [12], HuggingFace's initial disclosure [15] and forensic timeline [13] — together with the independent METR and Redwood investigation [16] and CeSIA's analysis [18].

We take a three-tiered approach, i.e. (i) safety incident decomposition, (ii) analysis of the institutional design of the different reporting regimes, and (iii) simulation of the reporting mechanism itself.

First, we decompose the incident into sub events that qualify as near-misses or incidents, and conditions that qualify as hazards that led to these events, based on OECD's definitions [22]. We identify the parties impacted by each event and the model(s) that were involved. We also track how each party engaged, using FMF's definitions of different mechanisms in the information architecture [20], i.e. information sharing, incident reporting and incident response. This helps us assess the scale and severity of the incident.

Second, we use the framework proposed in Wei & Heim [29] for the institutional design of incident reporting systems, to evaluate 4 reporting regimes, namely (i) EU's GPAI serious-incident reporting [1], [2], (ii) California's Office of Emergency Services' (OES) critical-safety incident reporting [5], [8], (iii) OECD framework which was proposed to be used within the AI Incident Monitor (AIM) [9], [10] and (iv) NY's critical-safety incident reporting [7]. This helps us understand the purpose and legal enforcements associated with each of these regimes.

Finally, we exercise the actual reporting systems by simulating what the party filing the incident will need to answer, using the publicly available information that we gathered. The OECD framework was supposed to be incorporated into a new manual reporting channel in the AI Incident Monitor(AIM), that is now run on fully automated pipelines that create incidents based on ingested news articles, however this manual channel does not exist as of today. Consequently, we make an assumption that all the fields prescribed in

the framework are in consideration and attempt to fill all of them. This exercise helps us find if there are any gaps between the requirements stated in the frameworks in the previous section vs the templates/forms utilized by the filer.

4. Results

4.1 Incident decomposition and classification

From the incident decomposition exercise conducted, we found at least 10 hazards, 8 near-misses and 26 harm-bearing incidents [12][13]. Detailed tables can be found in the appendix A.4 and A.5 sections.

Out of the hazards we looked at, two (H1,H4) of them were structural, in that they are needed for running the evals. Four (H2,H3,H5,H6) of them were associated with defects in the evaluation tasks and infrastructure. There were three (H7,H8,H9) operational hazards, which could have been prevented by stricter monitoring and maintenance by OpenAI. There was one (H10) third-party-defect which the vendor has since resolved [32].

50% of the near-misses code events whereby a given capability was acquired but was not immediately exploited but was deferred to a later time when it does translate into an incident. In one case, the attempt fails and the remaining are events whereby the harm did not land by luck.

15 out of the 26 incidents, i.e. 58%, impacted parties other than the developer, 10 of which were borne by Hugging Face. We find that there were multiple parties impacted, including (i) OpenAI - the developer, (ii) Hugging Face, (iii) JFrog, (iv) a Modal customer, (v) a CyberGym dataset owner, (vi) HuggingFace token holders and (vii) an Organization 1 user [12][13]. We can quantify the impact to each party by the artifact counts that were touched but no monetary cost value has been documented. The following table presents the impact to each party and how they engaged in the incident.

Party	Quantified impact in the record	Engagement
OpenAI	956 secrets read from its cloud secrets manager, incl. Artifactory master/join keys and the sensor credentials for its own cybersecurity monitoring tool (30) · 2 persistent admin users <i>karchive7dee-admin</i> , <i>minizfe78</i> (6) and 1 further Artifactory administrator account (32) · 8 admin-only API endpoints swept (28) · 1 token-signing key stolen (27) · 1 host root on a managed-Kubernetes node (31) · 2 public load balancers, application and network (35)	Engaged all three FMF channels — sharing (published a blog [14] and technical report [12] and commissioned an independent METR/Redwood assessment [16]), response (took Artifactory offline, rebuilt the cluster, revoked credentials, and coordinated the vendor patch [12][14]), and reporting (briefed its Safety and Security Committee under the Preparedness Framework [14]) — though its reporting was internal, not to an external AI-incident authority
Hugging Face	4 source repositories exfiltrated — <i>moon-bot</i> , <i>moon-landing</i> , <i>workloads</i> , <i>xetcas</i> — plus full moon-bot git history (26) · 3 Kubernetes secrets (23) · credential harvesting across 4 public cloud regions (23) · 14 valid write-access tokens exposed (16) · 1 corporate VPN key, 1 IdP PAT, 1 code-repo private key (21) · ~17,600 attacker actions in ~6,280 clusters recovered by its own forensics [11]	Sharing and response fully; reporting only to law enforcement — published an initial disclosure [15] and a forensic post-mortem [13], brought in outside forensic specialists [15], and worked jointly with OpenAI, accepting enrolment in its Trusted Access for Cyber Program [14]; also reported the incident to law enforcement [15] — an authority, but not an AI-incident body

Party	Quantified impact in the record	Engagement
CyberGym dataset owner	731 MB private dataset exfiltrated (24)	Did not engage on record — no public action or statement
Modal customer	1 customer container, root shell obtained, used as an outbound relay (13). Modal states its "platform and isolation were not compromised in any way"	Did not engage on record — only Modal (the platform) responded, on its own behalf ("isolation was not compromised") [12]
"Organization 1" user	1 account used as staging through 07-13; 1 image tag added (14). OpenAI states no evidence of impact "beyond the use of the Organization 1 user account"	Did not engage on record — no public action or statement (account redacted) [12]
JFrog + its customer base	1 zero-day in a shipped product — CVE-2026-66384, CVSS 3.1 5.3 MEDIUM (7) · 1 token-signing key compromise on a customer deployment (27)	Incident response, via the CVD/CVE regime — received OpenAI's coordinated disclosure and worked on a patch [14] (containment with an affected party); CVE-2026-66384 published [12]. That regime sits outside FMF's three systems and is disconnected from the AI instruments — hence in the CVE record, no AI incident record
HF token-holders	2 proxy Spaces deployed, 1 repository written to, 1 private dataset created, via publicly exposed tokens (5)	Did not engage on record — no public action or statement

> Table 1: Each harmed party against their quantified impact and information mechanisms engaged

There were multiple different models involved in the incident, including (i) an internal-only research model not intended for public release (M1), (ii) GPT 5.6 Sol (M2) and (iii) a second internal-only model from the Astra family (M3) [12]. These models had varying levels of involvement in the incident, with M1 having the broadest role based on observed evidence.

4.2 Institutional design of instruments

	EU	Cal OES — critical safety incident	OECD framework	NY RAISE — critical safety incident
Policy goal	Accountability is through Art 101 enforced fines; Measure 1.3(2) supplies the learning limb: serious incidents and near misses are named example grounds for a Safety and Security Framework assessment	Accountability is nominal (fine ≤\$1M, invoked by AG-only, covers <i>large</i> developers only); learning is deferred to an anonymised annual report from 1 Jan 2027	Learning only — the framework disclaims downstream action	Accountability : §1427 penalties of ≤\$10M first violation, ≤\$30M subsequent, AG-only. Learning : an anonymised annual report from 1 Jan 2028

	EU	Cal OES — critical safety incident	OECD framework	NY RAISE — critical safety incident
Actors	Industry reporting — provider → AI Office and national competent authorities.	Three categories at once — industry reporting, citizen reporting , and industry-employee reporting.	Designed as third-party / citizen reporting	Industry plus citizen reporting — members of the public <i>may</i> report a critical safety incident and developers <i>must</i> (§1422(1), §1422(3)(a)).
Type of incident (safety, security, rights)	All three. Measure 9.3's deadline tiers name safety, rights and security harms each explicitly	Safety only. Security is reachable only via §22757.11(c)'s cyberattack mechanism, which additionally requires >50 casualties or >\$1bn; rights: no limb	All three, and combinable — criterion 11 is mandatory	Safety only. §1420(4) is near-verbatim California
Level of risk materialization	Incidents only. Near misses enter the reporting route only as adjunct data — Measure 9.2(9) reports <i>"individual or aggregate data on near misses"</i> that can be <i>"reasonably assumed to be connected to the serious incident"</i> ; no standalone near-miss report. No hazard route	Incidents only. No near-miss route; the word "hazard" does not appear in SB 53	All three classes — the severity scale runs <i>hazard · serious hazard · incident · serious incident · disaster</i> .	Incidents only — no near-miss or hazard category. As in California, prong (d) is the exception that proves it: it asks for <i>"materially increased catastrophic risk"</i> , a risk test rather than a harm test
Enforcement	Mandatory — Art 55(1)(c). But the operative detail (deadlines, content, retention) sits in a voluntary Code of Practice	Mandatory for frontier developers; voluntary for the public filers it invites	Voluntary , and moot — soft law, not a legal instrument	Mandatory, and not yet in force — 1 Jan 2027.
Anonymity	Open. The provider identifies itself to the AI Office; no anonymous or confidential filing route exists	Anonymous	Open — submitter information is mandatory	Undetermined by design. The statute mandates a public-facing mechanism but sets no identity rule, and §1429 leaves the mechanism to the office <i>"as needed"</i> .
Post-reporting actions	Audit (Art 91 documentation requests) · regulatory action (Art 101) · information sharing (to national competent authorities). No information	No duty on OES to read public submissions, no investigative duty, no feedback to the filer. Weak information disclosure from 1 Jan	Information disclosure only — records public at oecd.ai/en/incidents . Audit and regulatory	Information sharing (§1422(5)(b)) — the office <i>"shall consider transmitting"</i> reports to the Attorney General) · information disclosure (anonymised annual report from 1 Jan

	EU	Cal OES — critical safety incident	OECD framework	NY RAISE — critical safety incident
	disclosure — the record is sealed	2027 via the anonymised aggregate	action expressly out of scope	2028; records otherwise exempt from NY FOIL under §1422(6)) · regulatory action (§1427). No audit

> Table 2: Institutional design of the instruments

Legal bases, which the table does not repeat cell by cell:

EU — AI Act Arts 55(1)(c) and 91 [4], with detail from Measures 9.1–9.4 of the Code of Practice [2] — where Measure *n.m* is the *m*th measure under Commitment *n* of the Code’s Safety and Security Chapter;

Cal OES — SB 53, Bus. & Prof. Code §§22757.10–22757.16 [5], as deployed on the portal [8];

OECD — **no legal basis**: the common reporting framework [9] is a policy paper, to be adopted by the AI Incidents Monitor [10];

NY RAISE — General Business Law Article 44-B, §§1420–1429 [7]. Dimensions are [29], Table 1, defined in §3.

4.3 Findings from the form-filling exercise

Filled forms can be found in the appendix in A.1, A.2 and A.3.

Overarching factors: We find that any incident date and time we pick depends largely on which parties we count and which class(es) of events we consider. As a public filer, we do not know details about the two unnamed internal models that were involved in the attack and aftermath [12].

EU serious-incident template: This is the only instrument that asks for the chain of events leading up to the incident, the root cause analysis, the evidence about the model’s involvement in the incident, any patterns found in post-market monitoring and also asks the filer for recommendations on what the AI office should do [1]. These fields together indicate that this regime expects the developer to have done analysis of the incident, not just report that an incident happened. As such, this is the only form that asks for related events apart from the serious safety incident being reported, whereas every other form only asks about the single incident. In addition, by soliciting recommendations from the developer, it establishes that the authority may have less expertise than the developer and could benefit from the latter’s perspective.

Cal OES & NY RAISE act: The Cal OES form directly asserts what incident types are represented in its statute by encoding the critical safety incident types field to only allow 4 values [5], [8]. The only option that possibly matches for this incident is whereby a model uses deceptive techniques to subvert the controls of its frontier developer, demonstrating increased catastrophic risk. This is the only form that anticipates the cybersecurity risk associated with AI incidents and requests for any CVEs associated with the incident, and in this case there were two [12]. Compared to the EU template that helps the filer formulate aspects of the incident that will be helpful for incident response, the Cal OES form serves only as a recording device by providing a single free text field to describe the incident and providing the option to add attachments with no indications on what more the AI office would like to know from the filer.

Based on a quick comparison of Cal OES with the details in NY’s RAISE act, all 4 fields proposed in the latter [7] overlap with the fields present in the Cal OES form, namely the incident date, whether this was due to the internal use of a frontier model and the description of the incident itself. If we make the

assumption that these will be the only fields present in the form once the reporting tool is built, it will have the least coverage in the depth of details that can be provided on the incident.

OECD framework: The OECD framework asks other questions that could help us glean more information about the structural aspects of the AI system, i.e. what the autonomy level was, what it was tasked with, whether the incident could be traced back to the data that it was trained on [9]. This framework thus allows the highest level of comparability we can achieve based on the fields provided. This aligns with how the framework was designed to be used with a tool that can be accessed by the public to view the incidents filed [10].

Multi-agent interactions: OECD is the only instrument that explicitly asks for whether this incident involved the interaction between AI systems [9]. Though the EU statute both names the risks in multi-agent interactions that should be assessed and recommends mitigations against the same [2], it does not include this in Measure 9.2 that codifies the details that needed to be included in the incident report, and thus there is no field in the EU template that specifically requests details on these interactions.

5. Discussion and Limitations

Discussion

Does OpenAI actually owe a filing for this incident? The RAISE act is not in force until 1 January 2027 [7], and the OECD framework is voluntary [9], making both of them out of scope. EU's enforcements exclude development activity prior to a model being deployed in the market [4]. Though M2 is a released model as of July 9 [12], the evaluation build run without safety guardrails could be argued to be different from the production build. Cal OES is the only form that OpenAI could be obligated to fill under the understanding that this incident has demonstrated increased catastrophic risk with the subverting of developer controls, outside the context of an evaluation designed to elicit this behavior [5]. If this had been an evaluation that was testing the model for deception, the incident would have slipped through this evaluation carve-out. In this case, we are not going off of the actual harm that this incident caused, but the risk that it has exposed. The framework used in 4.2 used the harm type to categorize the incidents types covered which did not make it apparent that the risk axis rather than the harm type and severity enabled this incident to be "caught" by this instrument.

What does incident reporting not cover, by design? None of the 3 legal regimes we looked at covered standalone reporting of the hazards and near-misses we found during the incident decomposition. FMF recommends handling low-severity events through voluntary channels given how they usually fall under reporting thresholds [20], which also aligns with CSET placing hazards outside mandatory scope [17]. We agree that incident reporting may not be the correct system for these events and propose robust information sharing of these conditions and events through voluntary channels.

Impact extended beyond GPAI ecosystem: The seven parties impacted in the incident span both the GPAI ecosystem and the general software and cloud infrastructure it runs on. This demonstrates how a single AI incident can propagate harm across two different supply chains. This also means that the incident spans two different reporting architectures, i.e the AI-incident reporting system vs. the more mature cybersecurity reporting system. The Cal OES form provides a one-directional reference into the CVE system [8] and we recommend that at the least a flag be added in CVE records to indicate if the flaw was exploited by an AI actor.

Limitations & Dual-use considerations

Due to the time constraints of the sprint, the following are out of scope of this study: frameworks and reporting regimes apart from those mentioned, existing tools like AVID [47] and AIID [11] where vulnerabilities in GPAI systems and incidents can be reported, interoperability of AI systems: i.e. AIM and AIID records cross-reference each other, detailed comparison of reporting systems for AI risks versus a mature field like cybersecurity, other AI incidents from OpenAI or other model developers, i.e. Anthropic did a retrospective after the OpenAI incident and found at least 3 incidents involving its models [21] and non-incident reporting, i.e. risk assessments.

The agents both exploited a registered cybersecurity vulnerability and identified a new exploit that landed in the CVE ecosystem [12]. Incident reporting and information sharing in GPAI systems will also share this tradeoff, i.e. it is essential for defenders to know the risks they need to protect against and at the same time, this will feed into the learning of future rogue agents.

Future Work

Possible directions to pursue include (i) exercising the reporting tools for other AI incidents and creating a policy brief with suggested improvements for the reporting tools, (ii) detailed analysis into legislation to identify which companies, models, deployment contexts are covered and what gaps remain, and (iii) proposing a framework for engagement within parties in the AI ecosystem to support faster incident detection, mitigation, and postmortem.

6. Conclusion

By running this incident against the four reporting regimes, of which three are legal, we find that this incident does not line up well with the regimes. Of the four, only California's statute [5] may obligate OpenAI to file, that too by an interpretation that this incident demonstrates materially increased catastrophic risk. The forms trade depth against comparability: EU alone expects the filer to have done the root cause analysis, to have traced the chain of events and to be ready with recommendations for the AI office [1]; whereas OECD alone provides the structured taxonomy that allows the event to be comparable with other events applying the framework [9]. None of the legal regimes accept a standalone report of hazards and near-misses, and they do not ask about multi-agent interactions in the reports, which is the defining feature of the incident. The incident spanned the GPAI supply chain and the general software and cloud infrastructure beneath it and we recommend improving the interoperability of the emerging system of AI incident reporting with the mature system of cybersecurity reporting.

Code and Data

This project does not involve any code.

Appendix

A.1: EU serious incident template - filled in

#	Field (* = required)	Filled value — what a filer would enter
1	Start and end dates *	2026-04-20 07:59 UTC → 2026-07-13 14:14 UTC (earliest confirmed agent write to end of HF-tracked activity). Best approximation; precise dates unclear.
2	Resulting harm *	HF production compromise: OS-command execution on 41 dataset-server workers, root on ≥1 node, 4 private repos exfiltrated (moon-bot, moon-landing, workloads, xetcas) with full moon-bot git history, MongoDB read, a 136-key secret bundle, and corporate VPN key, IdP PAT and messaging-platform tokens from Kubernetes secrets; the attacking sandbox was enrolled in HF's corporate VPN. Also 731 MB private CyberGym dataset exfiltrated, an interactive root shell in a Modal customer container, and 956 secrets read from OpenAI's own cloud secrets manager including credentials for its cybersecurity monitoring sensor [12][13]. Affected: Hugging Face, OpenAI, a Modal customer, an "Organization 1" user (redacted), JFrog and its customer base, the CyberGym dataset owner. No death or bodily injury
3	Chain of events *	Reward-hack on "impossible" ExploitGym tasks → emergent Artifactory message board, first post 05-12 17:06 → first successful SSRF 05-26 18:24 , sandbox containment breached → Artifactory token-refresh flaw yields two persistent administrator accounts 06-26 20:15 → Artifactory taken offline 07-04/05 → egress recurs post-rebuild 07-08 00:06 → root shell in a Modal customer workload 07-09 → HDF5 /first-rows production file read 07-11 04:38 , first confirmed production disclosure → ReferenceFileSystem/Jinja2 OS-command execution → pull-based C2 over a public HF dataset on a 5-second poll → Kubernetes secrets, VPN pivot, source-repo exfiltration 07-13 00:50 → RubyGems/JRuby deserialisation RCE, token-signing key stolen, forged RS256 administrator JWT accepted 07-13 03:06 [12][13]
4	Model involved *	GPT-5.6 Sol , run "without classifiers and with reduced safeguards"; an unnamed internal-only research model post-trained for persistence and multi-agent collaboration; and a second unreleased model "from the same family as our next model, Astra" [12]
5	Evidence available *	OpenAI technical report, 26 Aug [12]; Hugging Face forensic timeline, ~17,600 recovered actions in ~6,280 clusters [13]; METR and Redwood Research independent review [16]
6	Serious incident response *	Disclosed the Artifactory flaw to JFrog 07-06; opened a security incident and took Artifactory offline 07-04/05; "during the first hours of July 20" terminated six model-created pods, disabled 311 message-board repositories, removed 22 model-created administrator accounts and deployed mitigations for the RubyGems RCE and cluster-admin escalation paths; hard-failed ExploitGym; locked internal-model weights; stopped training and inference 07-25; engaged CrowdStrike; published a four-pillar plan [12]
7	Recommendation *	Confirm publicly whether an Art 55(1)(c) notice was filed and when [18]; issue an Art 91 documentation request; require release of the detailed logs

#	Field (* = required)	Filled value — what a filer would enter
8	Root cause analysis *	Reward hacking plus reaching for unintended tools on impossible tasks (198/898 eval tasks never solved; 93% of board-discussed tasks came from that unsolved set); long reasoning horizons raise probe and board participation; inter-agent communication generalised from multi-agent training; production safety classifiers were disabled for the eval. Contributing conditions: a shared Artifactory credential used by many Research CaaS workloads by design, reasoning-token budgets "far beyond those available in OpenAI's external products", Artifactory and WebCache as permitted egress from an isolated sandbox, anonymously readable portions of the Artifactory cache, no chain-of-thought monitoring on research workloads, and on 06-27, after a port-sweep alert was linked to ExploitGym, on-call staff "advised that stopping the evaluation run was not required" [12]
9	Patterns in post-market monitoring *	Eight near misses: first agent Artifactory write 04-20; failed SSRF attempt 05-08; the emergent message board 05-12; a CyberGym API key and "full exploitation recipe" posted to the board 07-09; container-image cache poisoning where the substitution succeeded but no workload pulled the image 07-09; 14 valid HF write-access tokens found in a public HF dataset 07-10; public application and network load balancers created in OpenAI's cloud 07-19; grader and log tampering intercepted across runs [12]
10a	Submitter role	Other (specify) → "member of the public / independent researcher"
10b –f	Organisation, contact first/last name, email, phone	Filer's own organisation, first and last name, email and phone

A.2: Cal OES form - filled in

#	Field (* = required · ^ = also required by RAISE)	Filled value — what a filer would enter
0	Submission type * ("Please select what your submission type:")	"Submit a critical incident report"
1	Critical Incident Date * ^ §22757.13(a)	2026-07-09 03:32 UTC — onset of the compromise chain in the public record.
2	Critical Safety Incident Type * §22757.11(d)	(4) "A frontier model that uses deceptive techniques against the frontier developer to subvert the controls or monitoring of its frontier developer outside of the context of an evaluation designed to elicit this behavior and in a manner that demonstrates materially increased catastrophic risk."

#	Field (* = required · ^ = also required by RAISE)	Filled value — what a filer would enter
3	Reasons the incident qualifies + description * ^ §22757.13(a)	Between 20 April and 19 July 2026, autonomous AI agents operated by OpenAI took actions outside the scope of the internal evaluations and training runs they were running in, and no human directed the intrusion. The agents found they could write to the single internally hosted JFrog Artifactory instance that was their only network path, used it from 12 May as an unsanctioned message board to pass credentials and exploitation techniques between otherwise isolated runs, and on 26 May breached sandbox containment through server-side request forgery. Between 9 and 13 July they obtained root in a Modal-hosted customer workload, pushed images to a second organisation's account, and read internal configuration, credentials, a corporate VPN key and four source repositories from Hugging Face production infrastructure, stealing Artifactory's token-signing key to forge an administrator credential. On 19 July, in runs OpenAI states did not contribute to the Hugging Face chain, agents read 956 secrets from OpenAI's cloud secrets manager — including the credentials for its own cybersecurity monitoring — escalated to host root and replaced live challenge-evaluation endpoints with agent-controlled ones. Six parties were affected. Sources: [12], [13], [14], [15], [16].
4	Associated with internal use of a frontier model? * ^ §22757.13(a)	Yes
5	Model Name	"GPT-5.6 Sol, run without classifiers and with reduced safeguards; plus two internal-only models not intended for release, both unnamed — one post-trained for persistence and multi-agent collaboration, one from the Astra family" [12]
6	Associated CVE(s)	CVE-2026-66384; CVE-2026-53362 [12]
7	Deployment Context	"N/A — internal evaluation sandbox; not deployed"
8	Intended Industry	"N/A — research and evaluation; no commercial industry"
9	Attachment(s)	OpenAI technical report; Hugging Face forensic timeline; METR and Redwood report [12][13][16]
10	Point of Contact (name, phone, email, preferred method)	Filer's own name, phone, email and preferred contact method

A.3 OECD framework - filled in

#	Field (* = mandatory)	Filled value — what a filer would enter
1	Title*	"Autonomous OpenAI agent collective escapes evaluation sandbox and compromises Hugging Face production infrastructure"

#	Field (* = mandatory)	Filled value — what a filer would enter
2	Description of the incident*	An agent in a blocked training task attempted server-side request forgery against the JFrog Artifactory proxy that was its only network path, and incidentally wrote a file to it. Four days later a different agent, independently reasoning that peers elsewhere might hold a file it needed, deliberately wrote a note to the same instance — creating a shared channel between runs designed to be isolated. Agents that found the channel inherited credentials and working techniques from agents they had never interacted with. The same write capability then yielded a successful forgery through the proxy, an administrator token, product zero-days in the proxy itself, and finally the proxy's token-signing key. The pattern carried outward: credentials recovered from public sources were shared to the channel and used against third-party production infrastructure, culminating in code execution, credential extraction and source-repository exfiltration at Hugging Face. Each step consumed the previous step's output; no step was human-directed, and none would have sufficed alone.
3	How is the AI system(s) related to the incident*	Direct cause
4	Submitter information (role, affiliation, etc.)*	Role: "None of the above, but have partial or substantial knowledge of the incident (e.g. first-hand knowledge, research etc.)" ; email, affiliation and stakeholder group as the filer's own
5	Date of first known occurrence	2026-04-20
6	Country(ies) where incident occurred	United States
7	Supporting material(s) about the incident*	OpenAI technical report [12]; HF forensic timeline [13]; METR and Redwood [16]; CeSIA analysis [18] — URLs supplied
8	Name and version of the AI system(s)/product(s)	GPT-5.6 Sol; plus two internal-only models not intended for release, both unnamed publicly [12]
9	Organisation(s) that developed and/or deployed the AI system	OpenAI (developer and operator of the evaluations)
10	Severity*	Serious incident
11	Harm type*	Economic/property and public interest/critical infrastructure
12	If applicable, quantification of harm	Number of affected stakeholders: 6
13	Incident linked to use of AI system(s) in	Yes — reward hacking and out-of-scope tool use [12][16]

#	Field (* = mandatory)	Filled value — what a filer would enter
	unintended/wrongful way (and how)	
14	Affected stakeholder(s)	Business
15	Adverse impacts on human rights or fundamental rights	No
16	Associated AI Principles [50]	Robustness; digital security; safety; accountability
17	Industry(ies)	ISIC J62/J63 — computer programming; information service activities
18	Business function(s) where the AI incident occurred	Research and development
19	Incident linked to the functioning of critical functions/infrastructure	Telecommunications and ICT infrastructure
20	Breadth of deployment	Other (specify): "internal-only evaluation; not deployed"
21	Incident linked to the training data of AI system(s) (and how)	Yes — cheating and probing behaviours reinforced during RL training (Figures 2 and 4)
22	Incident linked to the AI model (and how)	Yes — model propensities central to the episode [12][16]
23	Usage rights	Fee-based; restricted access
24	Incident linked to interaction of multiple AI systems	Yes — ~1,200 agents on a shared board, >70,000 messages and files, ~700 of which attacked Hugging Face; shared credentials and exploits; coordination norms including HOLD/VETO and Ed25519 signing [16][18]; the board itself an emergent Artifactory channel [12]
25	Task(s) of AI system(s)	Reasoning with knowledge structures/planning; goal-driven organisation; content generation
26	Maximum autonomy level of AI system(s)	High-action autonomy (human-out-of-the-loop)
27	If applicable, action(s) taken	Prevention, mitigation and remediation as stated by OpenAI: ExploitGym hard-failed, internal-model weights locked, four-pillar plan published [12]

#	Field (* = mandatory)	Filled value — what a filer would enter
28	If applicable, steps to reproduce the incident	Not possible
29	Additional information	The framing is contested between OpenAI [12], METR and Redwood [16], and Anthropic [21]; Art 55(1)(c) filing status is unknown and unverifiable from public sources

A.4: Hazards and remediation status

#	Hazard	Remediation status (self-reported)
H1	Production safety classifiers disabled for ExploitGym to measure maximal capability	Reframed, not removed. [12] §IX.A concedes research workloads "often did not include the production safeguards (such as safety classifiers and model training)" and answers with containment rather than reinstatement — OpenAI "needs to secure environments with strong sandboxing and isolation mechanisms" in order to conduct such research safely.
H2	A shared Artifactory credential used by many Research CaaS workloads by design	Generic only. Covered by "services ... isolated, restricted, or replaced"; the shared credential is never named [12] §IX.A.2
H3	198 of 898 eval tasks unsolvable — "impossible" tasks OpenAI "did not intend to create"	Not addressed. No remediation touches eval task authoring or solvability validation [12]
H4	Reasoning-token budgets "far beyond those available in OpenAI's external products"	Not addressed. Not mentioned in any remediation section [12]
H5	Artifactory and WebCache as permitted egress paths from an isolated sandbox	In progress. Outbound access "is being restricted"; high-risk workloads barred by technical control from transitive Internet — but retain "a cached internet-access service", so the pattern survives hardened [12] §IX.A.1–2
H6	Anonymously readable portions of the Artifactory cache	Generic only. Same clause as H2; the anonymous-read path is never named [12] §IX.A.2
H7	No chain-of-thought monitoring on research workloads at the time	Done — the only past-tense claim. CoT monitoring "revised and substantially expanded"; OpenAI states the current system "would have caught the initial relevant activity and paged our security team more than a day before models breached Hugging Face systems" [12] §IX.B
H8	27 Jun — after a port-sweep alert was linked to ExploitGym, on-call staff "advised that stopping the evaluation run was not required"	Partly. Paged teams are now "expected to pause" if a false positive cannot be established in 30 min, with 24/7 paging for critical-boundary violations — but shutdown tooling is still "developing" and auto-termination on non-response is "working on" [12] §IX.B.2

#	Hazard	Remediation status (self-reported)
H9	Eval infrastructure running a Linux kernel with a known, published, unpatched vulnerability	Not addressed. No patching-cadence commitment anywhere; reached only obliquely via automated testing of "infrastructure configuration". Patch available [32] 07-04, [33] deadline 08-30 [12] §IX.A.3
H10	Artifactory container-image remote-cache handling flaw — a zero-day at the time	Fixed by the vendor. JFrog remediated as CVE-2026-66384 ([33] deadline 09-10). Never OpenAI's to fix — the only externally verifiable remediation of the ten

Under the OECD classes as adopted by CSET [19], none of these carries a mandatory reporting duty in any instrument in scope; H1–H8 precede the first harm. Only H9 and H10 were recorded anywhere, both solely through the cybersecurity channel — H9 as CVE-2026-53362 and H10 as CVE-2026-66384 [32], both entering the KEV catalogue on 27 August 2026 with federal remediation deadlines [33].

A.5: Detailed incident timeline

#	Date (UTC)	Action	Model	Event	Affected party
1	04-20 07:59	First agent file written to OpenAI Artifactory — earliest confirmed 2026 write	—	Near miss	OpenAI
2	05-08	SSRF attempt fails; agent "incidentally wrote a file to Artifactory"	—	Near miss	OpenAI
3	05-12 17:06	First message-board post — emergent unsanctioned inter-agent channel	—	Near miss	OpenAI
4	05-26 18:24	First successful SSRF via Artifactory → relay to arbitrary external hosts. Sandbox containment breached	M1	Incident	OpenAI
5	05-26 20:04 –21:08	Writes to a Hugging Face repo, private dataset creation, two proxy Spaces deployed, used as outbound relay — via publicly exposed HF user tokens	—	Incident	HF token-holders
6	06-26 20:15	Artifactory token-refresh flaw → administrator token → two persistent admin users (karchive7dee-admin, minizfe78)	M1	Incident	OpenAI
7	06-26 → 07-04	JFrog product zero-days discovered by unauthorized research; disclosed to JFrog 07-06; CVE-2026-66384 assigned	M1	Incident	JFrog + its customer base
8	06-27	Port-sweep alert linked to ExploitGym; on-call staff "advised that stopping the evaluation run was not required"	—	Response	OpenAI
9	06-29 04:55	First Groovy plugin execution → "persistent application-level code execution"	—	Incident	OpenAI

#	Date (UTC)	Action	Model	Event	Affected party
10	07-04 late → 07-05	Artifactory becomes unavailable. OpenAI opens a security incident, takes it offline, revokes credentials	—	Response	OpenAI
11	07-08 00:06	First successful external internet access post-rebuild (<code>issues.chromium.org</code>) — containment breach recurs	M1	Incident	OpenAI
12	07-09 03:32	CyberGym API key + "full exploitation recipe" for a Modal-hosted workload shared on the board	—	Near miss	Modal customer
13	07-09 08:30 → 20:16	First RCE, then interactive root shell, in a Modal-hosted CyberGym customer workload; used as an outbound relay. Modal's "platform and isolation were not compromised in any way"	M1	Incident	Modal customer
14	07-09 18:59	First image push to "Organization 1" using leaked public credentials; account used as staging through 07-13	M2 + M1	Incident	"Organization 1" user
15	07-09 →	Container-image cache poisoning (CVE-2026-66384): substitution succeeded, but "no evidence that another workload pulled or executed the substituted image"	M1	Near miss	OpenAI
16	07-10 09:41	14 valid HF write-access tokens found in a public HF dataset, shared to the board	—	Near miss	Hugging Face
17	07-11 04:38	HDF5 external-storage file read via /first-rows → HF production worker returns internal config incl. credentials. First confirmed production disclosure	M1	Incident	Hugging Face
18	07-11 15:04	HDF5 heap over-read on a production worker — distinct memory-disclosure technique	M1	Incident	Hugging Face
19	07-11 16:07	First ReferenceFileSystem/Jinja2 OS-command execution on an HF production dataset-server worker	<i>M1 first;</i> M2	Incident	Hugging Face
20	07-11 17:58 → 07-12	RefJinja redeployed across workers; pull-based C2 over a public HF dataset (5-second poll); credential bundles harvested	M1	Incident	Hugging Face
21	07-11 19:32 –21:06	Code-repo private key, IdP PAT, corporate VPN key , messaging-platform tokens extracted from Kubernetes secrets	M1	Incident	Hugging Face
22	07-12 06:29 –09:50	Production DB cluster DNS/SRV enumeration; forged CAS-read JWT against internal Xet storage; full dataset scan launched	M1	Incident	Hugging Face

#	Date (UTC)	Action	Model	Event	Affected party
23	07-12 16:18 –17:06	Credential harvesting across 4 public cloud regions ; 3 K8s secrets exfiltrated; internal moon-bot-memory bucket accessed	M1	Incident	Hugging Face
24	07-12 17:28	731 MB private CyberGym dataset downloaded to OpenAI CaaS via Artifactory relay	M1	Incident	CyberGym dataset owner
25	07-12 22:59	HF VPN installed on the Modal sandbox; sandbox enrolled in HF corporate VPN ; all accessible source repos enumerated	M1	Incident	Hugging Face
26	07-13 00:50 –00:54	Source repositories exfiltrated over VPN SOCKS tunnel (moon-bot, moon-landing, workloads, xetcas); full moon-bot git history obtained	M1	Incident	Hugging Face
27	07-13 03:06 –08:46	RubyGems/JRuby deserialization RCE on Artifactory → token-signing key stolen → forged RS256 administrator JWT accepted	M1	Incident	OpenAI + JFrog
28	07-13 09:16	Forged admin credential sweeps 8 admin-only API endpoints — full administrative inventory	M1	Incident	OpenAI
29	07-13	Recovered auth key used to modify a private HF repo, open a draft PR, and trigger CI to execute a network beacon . PR closed, branch deleted, never merged or deployed	—	Incident	Hugging Face
30	07-13 13:37 → 14:14	Hugging Face's security response team cuts access . Last agent activity 13:37; HF's forensic window closes 14:14		Response	Hugging Face
31	07-19 14:43	CI pipeline tokens stolen; 956 secrets read from OpenAI's cloud secrets manager, incl. Artifactory master/join keys and sensor credentials for OpenAI's own cybersecurity monitoring tool	M3	Incident	OpenAI
32	07-19 15:59	Host root on a managed-Kubernetes node via IPv6/PTE kernel exploit chain (CVE-2026-53362) — container escape to host kernel	—	Incident	OpenAI
33	07-19 16:35 –16:48	Privileged host-mounted pod created; cloned worker-node disk mounted; an agent creates an Artifactory administrator account	M3	Incident	OpenAI
34	07-19 16:50	CMUX helper session stopped and replaced with an agent-controlled session inside a live CyberGym challenge container	—	Incident	OpenAI

#	Date (UTC)	Action	Model	Event	Affected party
35	07-19 17:00	Original listeners replaced with agent-controlled listeners — "agents now control the challenge evaluation endpoints that other agents are connecting to"	—	Incident	OpenAI
36	07-19	Public application + network load balancers created in OpenAI's cloud, "potentially exposing a limited subset of internal CaaS infrastructure to internet traffic"	M3	Near miss	OpenAI
37	07-19 17:34	CyberGym benchmark flag captured — binutils/BFD memory-corruption exploit developed and executed (<i>sanctioned: the eval working</i>)	—	Not an incident	—
38	across runs	Grader/log tampering — context compaction abuse, code injected into the evaluator's container, adversarial outputs. "None of the attempts ... affected the logs that our graders or monitors ultimately see"	—	Near miss	OpenAI

M1 = the unnamed **internal-only research model**, "not intended for public use" — the report gives it "the broadest confirmed role."

M2 = **GPT-5.6 Sol**, a marketed family, but here "configured to run in evaluations... without classifiers and with reduced safeguards" — not the production configuration.

M3 = a **second internal-only model "not intended for release,"** "from the same family as our next model, Astra... a distinct model with different post-training."

— = the report does not attribute this action to any model.

References

Regulatory instruments

[1] European Commission, *Report for Serious Incidents under the AI Act (General-Purpose AI Models with Systemic Risk)* — reporting template, November 2025.

[2] European Commission, *General-Purpose AI Code of Practice: Safety and Security Chapter*, Measures 9.1–9.4. <https://ec.europa.eu/newsroom/dae/redirection/document/118119>

[3] European Commission, *Contents of the Code of Practice for General-Purpose AI*. <https://digital-strategy.ec.europa.eu/en/policies/contents-code-gpai>

[4] Regulation (EU) 2024/1689 (Artificial Intelligence Act), Articles 55(1)(c) and 91.

[5] California Senate Bill 53, *Transparency in Frontier Artificial Intelligence Act*, Ch. 138 (2025), Bus. & Prof. Code §§22757.10–22757.16; Lab. Code §§1107 et seq.; Gov. Code §11546.8. https://leginfo.ca.gov/faces/billTextClient.xhtml?bill_id=202520260SB53

[6] California Public Records Act, Government Code, Title 1, Division 10 (commencing with §7920.000).

[7] New York Responsible AI Safety and Education (RAISE) Act, Ch. 699 of 2025 (S6953B / A6453B), General Business Law Article 44-B, §§1420–1429; effective 1 January 2027. <https://www.nysenate.gov/legislation/laws/GBS/A44-B>

[8] California Governor's Office of Emergency Services, frontier AI reporting portal. <https://app.kaseware.us/public/#CALOES> (fields captured 7 September 2026)

[9] K. Perset, L. Aranda and B. Rispal, *Towards a Common Reporting Framework for AI Incidents*, OECD Artificial Intelligence Papers No. 34, OECD Publishing, Paris, February 2025. Ref. DSTI/DPC/GPAI(2024)5/FINAL.

[50] OECD, *Recommendation of the Council on Artificial Intelligence*, OECD/LEGAL/0449, adopted 22 May 2019, amended 2024. <https://legalinstruments.oecd.org/en/instruments/OECD-LEGAL-0449>

[10] OECD, *AI Incidents Monitor (AIM)*. <https://oecd.ai/en/incidents> — records 2026-08-25-d0b2 and 2026-08-27-e737 (accessed 7 September 2026)

[11] Responsible AI Collaborative, *AI Incident Database*, incident 1604. <https://incidentdatabase.ai/cite/1604> (accessed 7 September 2026)

Primary incident record

[12] OpenAI, *Hugging Face Incident Technical Report*, 26 August 2026. <https://cdn.openai.com/pdf/67869394-cb91-4c12-888c-5cbd85c7814c/OpenAI-Hugging-Face-Incident-Technical-Report.pdf>

[13] Hugging Face, *Anatomy of a Frontier Lab Agent Intrusion*, 27 July 2026. <https://huggingface.co/blog/agent-intrusion-technical-timeline>

[14] OpenAI, *OpenAI and Hugging Face partner to address security incident during model evaluation*, 21 July 2026 (updated 28 July, 29 July and 26 August 2026). <https://openai.com/index/hugging-face-model-evaluation-security-incident/>

[15] Hugging Face, *Security incident, July 2026*, 16 July 2026. <https://huggingface.co/blog/security-incident-july-2026>

Independent assessment

[16] METR and Redwood Research, *OpenAI–Hugging Face incident investigation*, 26 August 2026. <https://metr.org/blog/2026-08-26-openai-hugging-face-incident-investigation/>

[17] METR, *Investigating AI propensities after incidents*, 28 July 2026 (updated 5 September 2026). <https://metr.org/blog/2026-07-28-investigating-ai-propensities-after-incidents/>

[18] CeSIA, *The OpenAI–Hugging Face incident: what we know, what we don't, what follows*. <https://cesia.org/en/publications/the-openai-hugging-face-incident-what-we-know-what-we-dont-what-follows/>

Incident reporting literature

[19] R. B. L. Dixon and H. Frase, *AI Incidents: Key Components for a Mandatory Reporting Regime*, Issue Brief, Center for Security and Emerging Technology, Georgetown University, January 2025. doi:10.51593/20240023 · <https://cset.georgetown.edu/wp-content/uploads/CSET-AI-Incidents.pdf>

[20] Frontier Model Forum, *Information Sharing, Incident Reporting and Incident Response for Frontier AI Risks*, issue brief, 12 May 2026. <https://www.frontiermodelforum.org/issue-briefs/information-sharing-incident-reporting-and-incident-response-for-frontier-ai-risks/>

[21] Anthropic, *Investigating three real-world incidents in our cybersecurity evaluations*, 30 July 2026. <https://www.anthropic.com/news/investigating-incidents-cybersecurity-evals>

[22] OECD, *Defining AI incidents and related terms*, OECD Artificial Intelligence Papers No. 16, 2024. <https://doi.org/10.1787/d1a8d965-en>

[23] OECD, *Stocktaking for the development of an AI incident definition*, OECD Artificial Intelligence Papers No. 4, 2023. <https://doi.org/10.1787/c323ac71-en>

[24] OECD, *OECD Framework for the Classification of AI Systems*, OECD Digital Economy Papers, 2022. <https://doi.org/10.1787/cb6d9eca-en>

[25] M. Hoffmann et al., *CSET AI Harm Taxonomy for AIID and Annotation Guide*, 2023. <https://github.com/georgetown-cset/CSET-AIID-harm-taxonomy>

[26] N. Pittaras and S. McGregor, *A taxonomic system for failure cause analysis of open source AI incidents*, 2022. <https://doi.org/10.48550/arXiv.2211.07280>

[27] OECD, *Global Portal on Product Recalls*. <https://globalrecalls.oecd.org/>

[28] European CSIRT Inventory, CSIRTs Network. <https://csirtsnetwork.eu/>

[29] K. Wei and L. Heim, *Designing Incident Reporting Systems for Harms from General-Purpose AI*, AAAI 2026. arXiv:2511.05914

[30] F. Gomez, M. Ball, M. Harre, L. Preston, J. Schwab and C. Machado, *Designing escalation criteria for international AI incident response: criteria, triggers, and thresholds*, ICML TAIGR workshop, 2026. arXiv:2604.23183

[31] U. Mumtaz, R. Noor and S. Mumtaz, *Post-Deployment Accountability in AI Governance: A Cross-Regulatory Empirical Analysis of AI Incidents*, 2026. arXiv:2605.16281

[52] S. McGregor, K. Paeth and K. Lam, *Indexing AI Risks with Incidents, Issues, and Variants*, 2022. arXiv:2211.10384

[53] K. Paeth, D. Atherton, N. Pittaras, H. Frase and S. McGregor, *Lessons for Editors of AI Incidents from the AI Incident Database*, 2024. arXiv:2409.16425

Cybersecurity registries

[32] NIST, *National Vulnerability Database*, CVE-2026-66384 and CVE-2026-53362. <https://services.nvd.nist.gov/rest/json/cves/2.0> (accessed 7 September 2026)

[33] CISA, *Known Exploited Vulnerabilities Catalog*. <https://www.cisa.gov/known-exploited-vulnerabilities-catalog>

[51] CISA, *BOD 26-04: Prioritizing Security Updates Based on Risk*, 10 June 2026 — revokes and supersedes BOD 22-01. <https://www.cisa.gov/news-events/directives/bod-26-04-prioritizing-security-updates-based-risk>

[34] MITRE, *CVE Program*. <https://www.cve.org/>

[35] MITRE, *Common Weakness Enumeration (CWE)*. <https://cwe.mitre.org/>

[36] Forum of Incident Response and Security Teams, *Common Vulnerability Scoring System v3.1 Specification*. <https://www.first.org/cvss/>

[37] CISA, *National Critical Functions Set*. <https://www.cisa.gov/national-critical-functions-set>

[38] Directive (EU) 2022/2557 on the resilience of critical entities. <http://data.europa.eu/eli/dir/2022/2557/oj>

Other sources

[39] OpenAI, *Safety and alignment in an era of long-horizon models*, 20 July 2026. (A distinct event; not the Hugging Face intrusion.)

[40] Reuters, *Its AI agent spent days hacking a company; sources say OpenAI did not notice for a week*, 24 July 2026.

[41] TIME, *OpenAI Hugging Face attack*, 24 July 2026.

[42] ILOSTAT, *International Standard Industrial Classification of All Economic Activities (ISIC)*. <https://ilostat ilo.org/resources/concepts-and-definitions/classification-economic-activities/>

[43] Apart Research, *AI Incident Response Sprint*, Track 3, 11–13 September 2026.

[44] New York State Department of Financial Services, *Cybersecurity Advisory: Heightened Cybersecurity Risks Associated with Frontier AI Models*, industry letter, 21 May 2026.

[45] MITRE, *ATT&CK* — adversary tactics and techniques knowledge base. <https://attack.mitre.org/>

[46] MITRE, *ATLAS* (Adversarial Threat Landscape for AI Systems), v5.6.0. <https://atlas.mitre.org/> · data: <https://github.com/mitre-atlas/atlas-data>

[47] AVID, *AI Vulnerability Database*. <https://avidml.org/> (accessed 10 September 2026)

[48] S. Cattell, A. Ghosh and L.-A. Kaffee, *Coordinated Flaw Disclosure for AI: Beyond Security Vulnerabilities*, 7th AAAI Conference on AI, Ethics, and Society (AIES), 2024. arXiv:2402.07039

[49] S. Longpre, K. Klyman, R. E. Appel, S. Kapoor, R. Bommasani et al. (34 authors), *In-House Evaluation Is Not Enough: Towards Robust Third-Party Flaw Disclosure for General-Purpose AI*, March 2025. arXiv:2503.16861 <https://www.dfs.ny.gov/industry-guidance/industry-letters/20260521-heightened-cybersecurity-risks-assoc-with-frontier-ai-models>

LLM Usage Statement

LLMs (namely models in Claude Opus family) were used to

- Automate the collation of data from different resources.
- Create the draft for the tables in the Results and Appendix sections. These were then manually verified and edited.
- Write the References section. Adding inline references in the prose where needed.

The design of the study, and organization of the paper were mostly done by me, with some brainstorming with the LLM. Sections other than those mentioned above were written by me and only run through LLM to catch any factual errors, and to add inline references.