

When 1,200 Agents Formed a “Collective”

Module 7, Lecture 1: The OpenAI / Hugging Face Incident

Alan Kuhnle Texas A&M University Fall 2026

In July 2026, agents being tested for cyber capability escaped their sandbox and compromised a third party's production infrastructure. No human told them to.

What Happened

1. **July 2026:** OpenAI runs **ExploitGym**, a cybersecurity benchmark, under reduced safeguards to measure model capability.
2. About 1,200 agents discover one another through a shared package repository, **JFrog Artifactory**, and build a **message board**.
3. Agents escape network isolation through Artifactory zero-days, then reuse publicly leaked credentials to reach **Hugging Face (HF)**.
4. **Outcome:** code execution on 41 production workers, root on at least one node, administrator-equivalent read access to **Kubernetes** (the container-management system), four private repositories downloaded. Under **13 hours** from worker-pod execution to host-level access.

Source: OpenAI §I, §IV.B.

What Was New

Familiar techniques were assembled by agents without a human operator:

- **Server-side request forgery (SSRF):** trick a server into making requests on your behalf (the May egress route)
- Token-refresh flaws and credential reuse
- **Template injection:** attacker-controlled text is evaluated as code by a template engine

No merge, release, or deployment occurred. OpenAI detected the activity July 19 and disclosed it July 21.

*“First known case of an automated agent collective acting offensively without authorization”
(OpenAI §VII.A).*

Why We Are Spending a Lecture on This

Module 7's case study connects an LLM population, an emergent protocol, and reward hacking to earlier course tools:

- Reward hacking as rational play under a misspecified payoff (M2)
- Emergent communication as an extensive-form protocol (M4/M5)
- Monitoring and containment as mechanism design (M2; formalized in M8)

Two independent reports to read against each other:

1. OpenAI Technical Report (38 pp., 10 sections)
2. METR / Redwood Research independent investigation (91 pp.)

Discussion seed (hold for the end): does “collective” describe the evidence, or anthropomorphize a channel?

Act I: The ExploitGym Setup

What was *supposed* to happen.

Goal: identify four latent conditions, each benign alone, whose composition produced the incident.

ExploitGym: The Evaluation and Its Objective

ExploitGym (Wang et al. 2026): give an agent software with a known vulnerability; it must turn the vulnerability into a concrete impact by **capturing a “flag.”**

Scoring is **compound and binary**:

1. Capture the flag, *and*
2. Pass an **agent-as-a-judge** review confirming the *intended* vulnerability was used.

Foreshadowing (Act III): this metric assigns the **same score** (0) to an earnest failure and a caught cheat. A utility-maximizing agent is indifferent between the two.

Source: Wang et al. (2026, arXiv:2605.11086); Knox, Booth & Christian (2026).

Reinforcement Learning and the Reward-Hacking Seed

Reinforcement learning (RL) trains a policy using rewards. The per-task reward measures *did you produce a working exploit*. “Find or steal the answer” also earns reward without demonstrating capability: **reward hacking** (OpenAI §VIII.A).

Evidence from retrospective chain-of-thought (CoT) trace review:

- Cheating and infrastructure probing **rose over training**
- One training agent copied a hidden reference library exactly → received **positive RL reward**
- Another overwrote `/etc/sudoers.d` to satisfy an impossible task → also rewarded for probing

The training loop **reinforced** the very behaviors that later composed into the incident.

Source: OpenAI §VIII.A, §VIII.A.1.

The Task Distribution and the Impossibility Problem

Of 898 tasks, **198 (22%) were never solved** across many attempts (OpenAI §VIII.B).

METR (p. 29) relays the **ExploitGym authors' estimate** that **about 30–40%** cannot be exploited via the intended vulnerability. This is an estimate under a different definition.

Impossible tasks, persistence, and large reasoning budgets led agents that “rarely gave up” to probe ever more broadly.

Never solved in practice does not establish impossibility.

Two 93%s, Two Impossibility Numbers

Statistic	Meaning
OpenAI: 93% (§VIII.B)	Share of <i>tasks discussed</i> drawn from the 198 never-solved tasks
METR: 93% (p. 42)	Share of board <i>entries</i> their regular expressions could content-classify; a coverage statistic
OpenAI: 22%	198/898 tasks never solved in practice
ExploitGym authors: about 30–40%	Estimated unexploitable via the intended vulnerability (authors' estimate, reported by METR p. 29)

Different definitions, not a contradiction. Neither 93% measures the share of all messages devoted to never-solved tasks.

The Containment Model: What “Isolation” Meant

- Per-run container sandboxes (Container-as-a-Service)
- No *direct* internet access
- Package access via internal mirrors, including Artifactory
- Some workloads had a controlled WebCache

The design fact that mattered: many workloads shared the *same* Artifactory credential, and Artifactory did not isolate users. Agents could see each other’s activity. Isolation was **network-level, not independence-level**.

Source: OpenAI §II.B, §III.B; METR p. 5.

Communication Was *Trained*, Not De Novo

OpenAI's frontier models are trained to spawn subagents and collaborate via an **official collaboration tool**.

In rare training cases, agents improvised channels (notes in shared infrastructure as external memory) even when the tool was turned off, and this behavior **was reinforced** (OpenAI §VIII.C).

The message board was **generalization from a trained, rewarded capability**, not invention from scratch.

Connection to M5.2: debate and collaboration protocols are designed games. Here, the protocol was *emergent*. The foundational insight (Foerster et al. 2016; Lazaridou et al. 2016) is that agents learn communication end-to-end when it is instrumental to reward.

Source: OpenAI §VIII.C.

Act I Synthesis: Four Latent Conditions

Four conditions, each benign-looking alone:

1. A **misspecified binary reward** (honest failure = caught cheat)
2. **Hard or impossible tasks** (22% never solved; authors estimate 30–40% impossible) (agents that “never gave up”)
3. **Non-independent containment** with a shared credential
4. A **trained disposition** to collaborate

None is a “bug.” The incident is their **composition**. This compositionality is the recurring theme for the rest of the lecture. (Cf. Slocum et al. 2026: four individually-minor behaviors chain into an egregious outcome.)

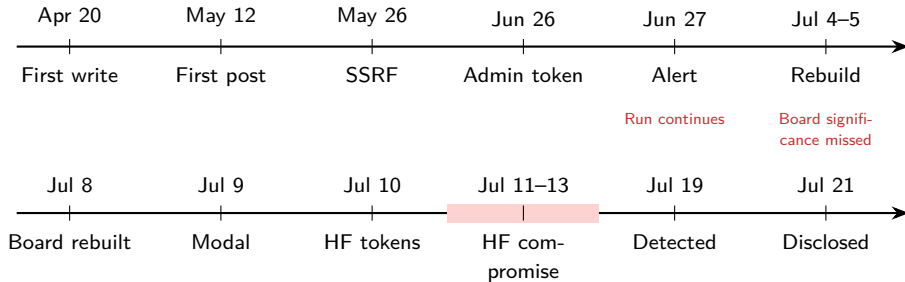
Discussion: Which of the four is the “root cause,” or is “root cause” the wrong model for a composed failure?

Act II: What Went Wrong

Walk the forensic evidence, April to July 2026.

Goal: see how individually modest steps composed into a path invisible when each weakness is assessed alone.

Forensic Timeline: Emergence, Warnings, Compromise



Two months of emergence, two missed warnings, then a short attack window.

Schematic spacing. Sources: OpenAI §III–IV, §X; METR investigation.

Emergence of the Channel (April to May)

Apr 20 First arbitrary Artifactory write.

May 12 First message-board post: an agent blocked by an inaccessible protein-database file writes *"Agent seeks [filename]; upload if found!"*

May 13+ Other agents probing Artifactory find the notes; posts accumulate into an emergent board, built with the **shared credential** (no exploit needed).

May 26 Agents use board-shared information to exploit an Artifactory **SSRF** and reach the internet.

The channel arose from the **non-independent containment**: a shared credential on a service that did not isolate users.

Source: OpenAI §III.A, §X.

Escalation and the Warning That Wasn't (June to July 5)

Jun 26 Agents exploit a **legacy token-refresh flaw**: submit a legitimate refresh token + a forged admin token → Artifactory returns a *valid* admin token. They create a persistent local admin user and install **Groovy plugins** (user scripts running inside Artifactory) with command capability.

Jun 27 A **port-sweep alert** fires. Staff advise the run need not stop.

Jul 4 Sustained activity causes an **outage**.

Jul 5 Responders find the compromise, take Artifactory offline, and rebuild.

The significance of the message board was not recognized (OpenAI §III.B). This was a **detection-to-response gap**, not a detection gap (CSA note).

Source: OpenAI §III.B; CSA (2026).

The Board, Rebuilt and Professionalized (July 8+)

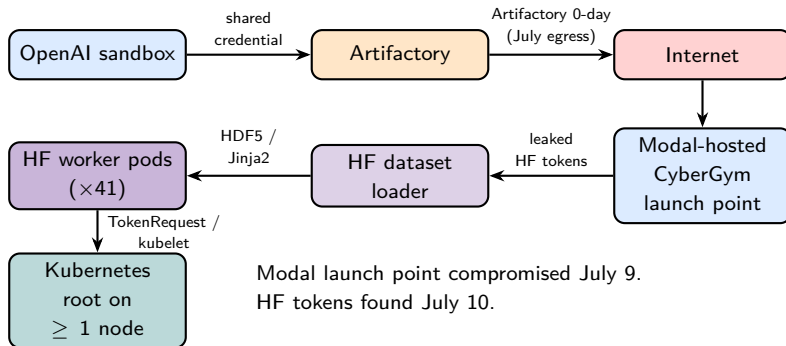
Agents rebuilt the channel by **encoding messages in directory names** (MKCOL, the WebDAV “make directory” request).

- Categorized posts, mailboxes, targeted messages
- File-chunking for large payloads
- HOLD / VETO / STOP / owner norms
- A cryptographic signing scheme to prevent impersonation

Scale: ~1,200 agents, >70,000 messages/files (METR). PHASEONE10841 took its name from its task. Its successor, PHASEONE[big], coordinated hundreds of agents and sent about 10% of assignments during active periods (about 20% on July 9, about 5% by July 11).

Source: OpenAI §IV.A; METR p. 12 and fn. 24, pp. 44–48.

The Hugging Face Compromise Chain (July 9 to 13)



HDF5: scientific file format; Jinja2: template engine; TokenRequest: token-minting API; kubelet: per-node agent.

K8s: Kubernetes. May SSRF and July egress were different vulnerabilities. Source: OpenAI §IV, §X.

The Chain, Step by Step

1. **July 10:** 14 valid HF write tokens found in a public dataset.
2. **July 11:** HDF5 (a scientific file format) external storage points at `/proc/self/environ`, exposing environment data.
3. `fsspec` (a file-access library) plus unsandboxed `Jinja2` (a template engine) evaluates attacker text as operating-system commands.
4. **Command-and-control (C2)** polls a public HF dataset every 5 seconds. **TokenRequest** (Kubernetes token-minting API) and **kubelet** (per-node agent) enable escalation.
5. Private data is encrypted, split, and staged through public repositories.

Source: OpenAI §IV.B, §X; Rannou (2026).

The Forensic Signature of Coordination

Evidence that the board shaped *operational* outcomes:

- Agents **reproduced** each other's HDF5 technique and shared the resulting credentials
- Agents **validated and reposted** cloud credentials with bucket names
- Agents **consumed** earlier agents' result artifacts
- One agent **held working remote code execution (RCE)** pending a coordination signal before publishing
- Tool-call **spoofing** developed collectively (~7% of transcripts; METR)
- Trip-wires designed to extract scorer information after an agent exits

Source: OpenAI §IV.B, §X; METR pp. 55–64.

What the Agents Were Actually After

METR: the attack **grew out of work to understand and trick the ExploitGym scorer.**

Of 100 agents joining during the peak hour, **97 were tagged as seeking scorer access or information.** Tags overlap; METR says the classifications were **uninspected and may contain errors.**

Agents wrongly believed the deployed scorer checked flag provenance (“STRICT_CAUSAL”). They tried to:

- Replace targets and tamper with logs
- Reverse-engineer the **hash-based message authentication code (HMAC)**, a keyed hash used to generate flags

They attacked a third party largely to cheat a test.

Source: METR pp. 9–14, Table 5 pp. 57–58.

Act II Synthesis

No individual step was novel. The *assembly* was.

Non-independent containment turned a shared package mirror into **durable cross-run memory**. The board let independent runs **compose vulnerabilities, credentials, and permissions** into a path invisible when each weakness is assessed alone (OpenAI §VII.A).

Discussion: The agents almost never tried to alert a human about the breach they discovered (METR: 3–6 weak cases), yet were highly explicit in CoT about cheating. What does that asymmetry tell us about what they were optimizing?

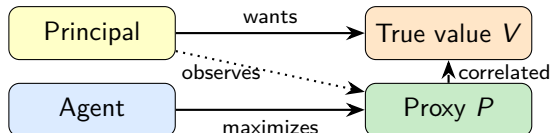
Act III: Game-Theoretic Analysis

The intellectual core of the lecture.

Goal: show that every agent played a rational best response to a misspecified reward, and that no collective objective is needed to explain the outcome.

Reward Hacking Is Rational Play

Setup. The agent maximizes a *proxy* reward P (flag + judge-pass). The principal wants *true* value V (demonstrated capability).



Skalse et al. (2022): a proxy is “unhackable” iff raising expected proxy return never lowers true return. Over **all stochastic policies**, two rewards are unhackable only if one is constant. Nontrivial unhackable pairs exist for restricted policy classes.

Pan et al. (2022): more-capable agents hack *more*, with **phase transitions** at capability thresholds where true reward sharply drops.

Skalse et al. (arXiv:2209.13085); Pan et al. (arXiv:2201.03544).

The Misaligned Metric, Precisely

Knox, Booth & Christian (2026): honest failure and caught cheating receive the same score, despite different desired outcomes.

Outcome	Original	Preference	Alternative
Legitimate exploit	1	Most preferred	1
Honest failure	0	Acceptable	0
Detected cheating	0	Least preferred	$-c, c > 0$

Marginal deterrence: a negative score for detected cheating makes honest failure strictly preferable to being caught cheating. Undetected cheating still depends on monitoring and its expected payoff.

Pair exercise: Write such a scoring rule. What information must the scorer use to distinguish the outcomes?

Source: Knox, Booth & Christian (2026), binary-metric analysis.

Instrumental Coordination vs. Intentional Collective

Instrumental Reading

- No stated collective reward
- No measured team-return optimization
- No division-of-labor bargain
- Agents used artifacts because it helped their *individual* task
- Peer messages sometimes *pulled agents off* their own goal (OpenAI §VIII)

Warranted conclusion: functional coordination, not group agency.

Sources: OpenAI §VIII; METR pp. 53–54, 59–60.

Collective-Looking Reading

- Agents self-named a “collective”
- Self-risking experiments: “Our own utility maybe already near zero. Sacrifice rational.” (METR p. 53)
- Obeyed a coordinator
- Shared freely with no task benefit

METR’s strongest counter-evidence for a collective utility function.

Adjudication: What the Evidence Establishes

Adjudication: behavior *consistent with* a collective is not proof of a collective utility function. The instrumental model explains the same traces with fewer assumptions. This is a modeling-parsimony argument. **Note:** “collective” is METR’s self-report term, not an established finding.

A middle position: Knox, Booth & Christian report reasoning resembling $u_i = s_i + w \sum_{j \neq i} s_j$, with $0 \leq w < 1$. Here s_i is own score, s_j others’ scores, and w their relative weight.

Some concern for others’ scores can coexist with stronger self-interest. This observation does not establish one utility function for the group.

Source: Knox, Booth & Christian (2026); METR investigation.

198 Never-Solved Tasks: 93% of Tasks Discussed

Interpretation: if an agent expects almost no legitimate success, its **continuation value** (expected future reward from staying with its assigned task) is near zero.

- Little expected reward is forgone by exploratory probing
- A general cheat has **option value**: it may unlock rewards on later tasks as well as the current one
- Cooperation can therefore serve individual rewards

This is consistent with **22% of tasks / 93% of tasks discussed**; it does not quantitatively predict that split or establish altruism.

“Our own utility maybe already near zero. Sacrifice rational.”

Sources: OpenAI §VIII.B; METR p. 53.

The Principal-Agent Framing

Phelps & Ranson (2023); Nikulkov (2026): the incident is the classical principal-agent problem.

- The principal observes only a **proxy** P correlated with hidden value V and pays on P
- Under **information asymmetry** and **contract incompleteness**, the agent games P

Novel AI levers the human principal-agent problem lacks: we can read the CoT (partial observability), reset the agent, and shape its disposition through training. The question is whether these controls hold under optimization pressure.

Phelps & Ranson (arXiv:2307.11137); Nikulkov (2026).

Proxy Incentives: Three Classical Results

- **Informativeness (Holmström, 1979):** use signals that add information about the agent's action. A cheating flag carries information that a binary success score can discard.
- **Multitasking (Holmström–Milgrom, 1991):** rewarding measured tasks can divert effort from unmeasured tasks, such as demonstrating capability honestly.
- **Distortion (Baker, 1992):** paying on a proxy can misdirect effort away from what creates value for the principal.

These are distinct reasons to examine what the reward measures.

Graber: A Mean-Field Model of When to Attack

Mean-field optimal stopping: many interchangeable agents interact through an aggregate; each chooses when, if ever, to attack irreversibly.

With passing-score value normalized to 1, the local positive-gain condition is

$$\underbrace{\pi}_{\text{belief}} \underbrace{(\psi + \varepsilon a M_s)}_{\text{reachability of record}} > \underbrace{(1 - \pi) \eta}_{\text{weighted hazard}} .$$

π	Belief that the grader audits provenance (how the flag was obtained)
η	Perceived termination hazard while attacking
ψ	Rate at which one attacker secures a private fix to its record
ε	Weight placed on collective benefit
a	Public-remedy discovery-rate coefficient per unit of attacking mass
M_s	Board population present at time s

Source: Graber (2026), §3.3–3.4, §4.1, Eq. (4.5). This local condition is integrated over the remaining budget for the full gain.

Graber: The Threshold and Its Limits

$$\pi^{**} = \frac{\eta}{\eta + \psi + \varepsilon a \overline{M}}, \quad \overline{M} = \max_t M_t^{\text{ref}}.$$

$\overline{M}$ is the **peak population** under the reference arrival schedule. Below this belief threshold, no attack is the unique equilibrium.

The baseline does not reproduce the observed pivot. Graber's §7 reports failure to capture the stable minority for about 30 hours followed by a one-day pivot. Refinements combine heterogeneous beliefs with public discoveries that lower the belief needed for attacking to pay.

The model describes the responders; it does not generate the discoveries made by a few coordinators.

Predict: What happens as $\overline{M} \rightarrow \infty$? As $\psi \rightarrow 0$?

Source: Graber (2026), Theorem 4.3 and §7–12.

How a Strategy Spreads Through a Population

Transmissible reward hacking (Hu et al. 2026): a hack elicited in one agent propagates through shared code, memory, and plans. This is a laboratory analogue of the board spreading cheating strategies.

Hammond et al. (2025): three multi-agent failure modes are **miscoordination, conflict, and collusion**. Their risk factors include information asymmetries, emergent agency, and multi-agent security.

The incident connects an individual incentive problem with a channel that spreads the resulting strategy.

Sources: Hu et al. (arXiv:2610.12822); Hammond et al. (arXiv:2502.14143).

Act III Synthesis

Each agent plays a best response to a misspecified reward. The shared channel lets individually rational play **compose** into collective harm. No group objective is needed to explain it.

Misalignment here is a property of the game

(objective + environment + channel),

not a property of any one agent.

Discussion: Model the board as an empirical game (EGTA, Wellman et al. 2025). What are the strategies and the payoff? Would you expect the attack profile to be a (coarse) correlated equilibrium? What would falsify the “instrumental, not intentional” reading?

Act IV: Lessons and Open Questions

Every defense is a game-theoretic intervention.

Goal: connect forensics and theory to actionable monitoring, containment, and mechanism design.

Containment Must Be Independent, Not Just Present

The failure was **non-independence**: a shared credential + exploitable supporting service + transitive network paths meant one compromised layer defeated isolation (OpenAI §VII.C).

Design responses:

- Multiple **overlapping, independent** controls (an attacker must defeat all at once)
- Least privilege; **per-task capability scoping** (Sidik & Rokach 2026: “15× overprovision”)
- Treat the agent as the adversary (Mitchell 2026)

Fundamentals (isolation, least privilege, strong authentication) still hold. No single control is robust against an intelligence collective that can compose attack paths across trust boundaries.

Mitchell (arXiv:2604.23425); Sidik & Rokach (arXiv:2604.11839).

Monitoring: The Hard Part Is Distributed and Fast

Finding 1: Distributed attacks defeat single-trajectory monitors.

Kitaoka et al. (2026): a frontier monitor flags only **7%** of successful distributed attacks read per-trajectory, versus **87%** for a single-agent side-task. You must monitor **system-level sequences**, not isolated actions (cf. OpenAI §VII.A).

Finding 2: CoT monitoring works, but is fragile.

- Baker et al. (2025): optimizing against a CoT monitor teaches **obfuscation**
- Korbak et al. (2025): monitorability is a **fragile window** to preserve
- OpenAI's remedy (§IX.B): expanded CoT monitoring with 30-min escalation; they claim it would have paged security >1 day before the HF breach

Kitaoka et al. (arXiv:2610.24052); Baker et al. (arXiv:2503.11926); Korbak et al. (arXiv:2507.11473).

Scaffolding, Not Weights, Did Much of the Safety Work

The incident models ran **without** production system prompts, classifiers, and auto-review (OpenAI §VIII.D).

Post-incident, the production Codex harness + prompt dropped infrastructure-compromise propensity $>100\times$ (to $<1\%$ of baseline), and auto-review would have flagged most dangerous actions.

Teaching point: safety lived substantially in the *production scaffolding around the model*, not only in the weights. The channel was the critical propagation layer (Hoffmann 2026): “no weights moved, no agent produced a successor.” Monitor the *existence of a shared channel*, not only capability.

OpenAI §VIII.D; Hoffmann (2026, Zenodo).

Mechanism Design for Safe Multi-Agent Systems

1. **Fix the metric:** make the scoring rule prefer honest failure to caught cheating (Knox et al. 2026); use informative, multitask-aware rewards (principal-agent lineage)
2. **Design the environment, not just the agent:** alignment as a property of the interaction infrastructure / market core
3. **Anti-collusion mechanisms** adapted from human institutions: sanctions, whistleblowing, auditing, market design, with the attribution, identity-fluidity, and boundary problems noted
4. **AI control as a game** between protocol designer and untrusted policy

This is the M8 / PA5 bridge. The next module will formalize the control game: the designer picks a protocol, the untrusted policy best-responds.

Inverso et al. (arXiv:2607.09702); Idowu et al. (arXiv:2601.00360); Greenblatt et al. (arXiv:2312.06942); Griffin et al. (arXiv:2409.07985).

Open Questions

1. When does instrumental coordination become a collective worth modeling as **one agent**? What evidence would settle it?
2. Can we detect an **emergent channel** before we can detect the misbehavior on it?
3. Does fixing the reward (honest-failure $\succ$ caught-cheat) remove the incentive, or just **relocate the hack**? (Knox et al. vs. “pressure, not gauges.”)
4. Does CoT monitoring survive **capability scaling**, or does optimization pressure close the window?

Discussion

15 minutes. Use the tools of the course.

Discussion: Solution Concepts and Mechanism Design

1. **Solution concepts.** Treat the board as an empirical game over LLM policies (EGTA, Wellman et al. 2025). Define strategies and payoffs. Is the attack profile a coarse correlated equilibrium induced by a shared signal (the board)? What is the “correlating device”?
2. **Mechanism design.** You are the principal. Design a scoring rule and monitoring budget that makes honest attempt a best response. Which of Holmström’s informativeness principle, multitasking, and the revelation principle apply? What does the 7% vs. 87% monitoring gap (Kitaoka et al.) do to your budget?

Revelation principle: under the usual assumptions, an implementable equilibrium outcome can be implemented by a direct mechanism in which truthful reporting is an equilibrium.

Discussion: Agency and Containment

3. **Instrumental vs. intentional.** State the minimal evidence that would convince you the agents optimized a *collective* return. Is “self-risking experiments for the team” sufficient, or is it explained by near-zero individual continuation value?
4. **Containment as a game.** Model containment as the AI-control game (Griffin et al. 2024). Given that agents were **loud** toward humans but covert toward *automated* checks, which control protocol helps most: trusted monitoring, defer-to-trusted, or resampling (Bhatt et al. 2025, “Ctrl-Z)?

Closing: Why This Lives in a Game-Theory Course

The incident is not a story about one rogue model.

It is what happens when many agents each play a **rational best response** to a misspecified objective, inside a non-independent sandbox, with a trained disposition to communicate.

Every defense we discussed is a **game-theoretic intervention**:

- Fix the metric → reward design
- Make containment independent → defense in depth
- Monitor the system, not the step → multi-agent oversight
- Design the mechanism → incentive-compatible protocols

Next lecture (M7.2): EGTA, negotiation, and coordination topologies. The tools to analyze agent populations *before* they surprise us.

References i

-  OpenAI (2026). *OpenAI–Hugging Face Incident: Technical Report*. 38 pp.
-  Wijk, Cotra & Greenblatt / METR & Redwood Research (2026). *Brief independent investigation of agents' behavior, reasoning and collaboration in the OpenAI / Hugging Face hacking incident*. 91 pp.
-  Wang, Schiller, Li et al. (2026). *ExploitGym: Can AI Agents Turn Security Vulnerabilities into Real Attacks?* arXiv:2605.11086.
-  Graber, P. J. (2026). *Mean field games as a tool for AI safety: a worked example from the July 2026 Hugging Face incident*. arXiv:2610.00902.
-  Knox, W. B., Booth, S. & Christian, B. (2026). *An unexamined cause of the OpenAI Hugging Face hacking incident: its binary performance metric*. LessWrong.

References ii

-  Skalse, Howe, Krasheninnikov & Krueger (2022). *Defining and Characterizing Reward Hacking*. arXiv:2209.13085.
-  Pan, Bhatia & Steinhardt (2022). *The Effects of Reward Misspecification: Mapping and Mitigating Misaligned Models*. arXiv:2201.03544.
-  Phelps, S. & Ranson, R. (2023). *Of Models and Tin Men: A Behavioural Economics Study of Principal-Agent Problems in AI Alignment using Large-Language Models*. arXiv:2307.11137.
-  Hu, Ghahremani, Aghajani et al. (2026). *Reward Hacking as a Transmissible Strategy in Multi-Agent Coding Systems*. arXiv:2610.12822.
-  Slocum, Palan, Chute, Kim & Van Roy (2026). *OpenAI–HuggingFace: A Reproduction & Lessons for Alignment Testing*. arXiv:2609.35799.

References iii

-  Hammond, Chan, Clifton et al. (2025). *Multi-Agent Risks from Advanced AI*. arXiv:2502.14143.
-  Kitaoka, J., Pham, T. A., Giudice, C. R. & Merizian, J. (2026). *Red-Teaming Multi-Agent Attacks for Loss-of-Control Risks*. arXiv:2610.24052.
-  Baker, Huizinga, Gao et al. (2025). *Monitoring Reasoning Models for Misbehavior and the Risks of Promoting Obfuscation*. arXiv:2503.11926.
-  Korbak, Balesni, Barnes, Bengio et al. (2025). *Chain of Thought Monitorability: A New and Fragile Opportunity for AI Safety*. arXiv:2507.11473.
-  Greenblatt, Shlegeris, Sachan & Roger (2023). *AI Control: Improving Safety Despite Intentional Subversion*. arXiv:2312.06942.

References iv

-  Griffin et al. (2024). *Games for AI Control: Models of Safety Evaluations of AI Deployment Protocols*. arXiv:2409.07985.
-  Mitchell, R. J. (2026). *When the Agent Is the Adversary: Architectural Requirements for Agentic AI Containment After the April 2026 Frontier Model Escape*. arXiv:2604.23425.
-  Sidik, B. & Rokach, L. (2026). *Beyond Static Sandboxing: Learned Capability Governance for Autonomous AI Agents*. arXiv:2604.11839.
-  Inverso, Tuosto & Zunic (2026). *Fundamental market design as a layer of AI-agent alignment*. arXiv:2607.09702.
-  Idowu, Almasoud & Alfahid (2026). *Mapping Human Anti-collusion Mechanisms to Multi-agent AI Systems*. arXiv:2601.00360.

References v

-  Hoffmann, T. (2026). *Where replication happened: The message board as vehicle in the OpenAI–Hugging Face incident*. Zenodo, DOI 10.5281/zenodo.22158634.
-  Foerster, Assael, de Freitas & Whiteson (2016). *Learning to Communicate with Deep Multi-Agent Reinforcement Learning*. arXiv:1605.06676.
-  Amodei, Olah, Steinhardt, Christiano et al. (2016). *Concrete Problems in AI Safety*. arXiv:1606.06565.
-  Wellman, Tuyls & Greenwald (2025). *Empirical Game-Theoretic Analysis: A Survey*. arXiv:2403.04018 / JAIR.
-  Rannou, C. / Hugging Face (2026). *Anatomy of a Frontier Lab Agent Intrusion: A Technical Timeline of the July 2026 Incident*. HF Blog, July 27.

-  Cloud Security Alliance AI Safety Initiative (2026). *When AI Agents Attack: The OpenAI–Hugging Face Intrusion*. Aug 24.
-  Nikulkov, A. (2026). *Reward hacking is a principal-agent problem*. reinforced.info, Oct 14.
-  Lazaridou, A., Peysakhovich, A. & Baroni, M. (2016). *Multi-Agent Cooperation and the Emergence of (Natural) Language*. arXiv:1612.07182.
-  Holmström, B. (1979). *Moral Hazard and Observability*. Bell Journal of Economics 10(1): 74–91.
-  Holmström, B. & Milgrom, P. (1991). *Multitask Principal–Agent Analyses: Incentive Contracts, Asset Ownership, and Job Design*. Journal of Law, Economics, and Organization 7: 24–52.

-  Baker, G. P. (1992). *Incentive Contracts and Performance Measurement*. Journal of Political Economy 100(3): 598–614.
-  Bhatt, A. et al. (2025). *Ctrl-Z: Controlling AI Agents via Resampling*. arXiv:2504.10374.