

2026



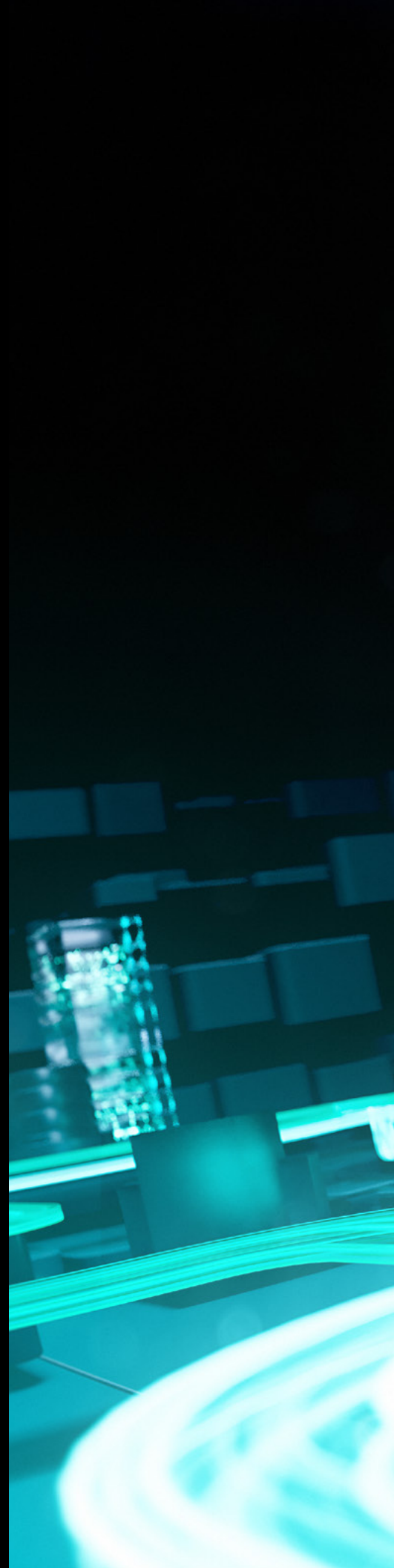
QUARTERLY UPDATE: JULY-SEPTEMBER

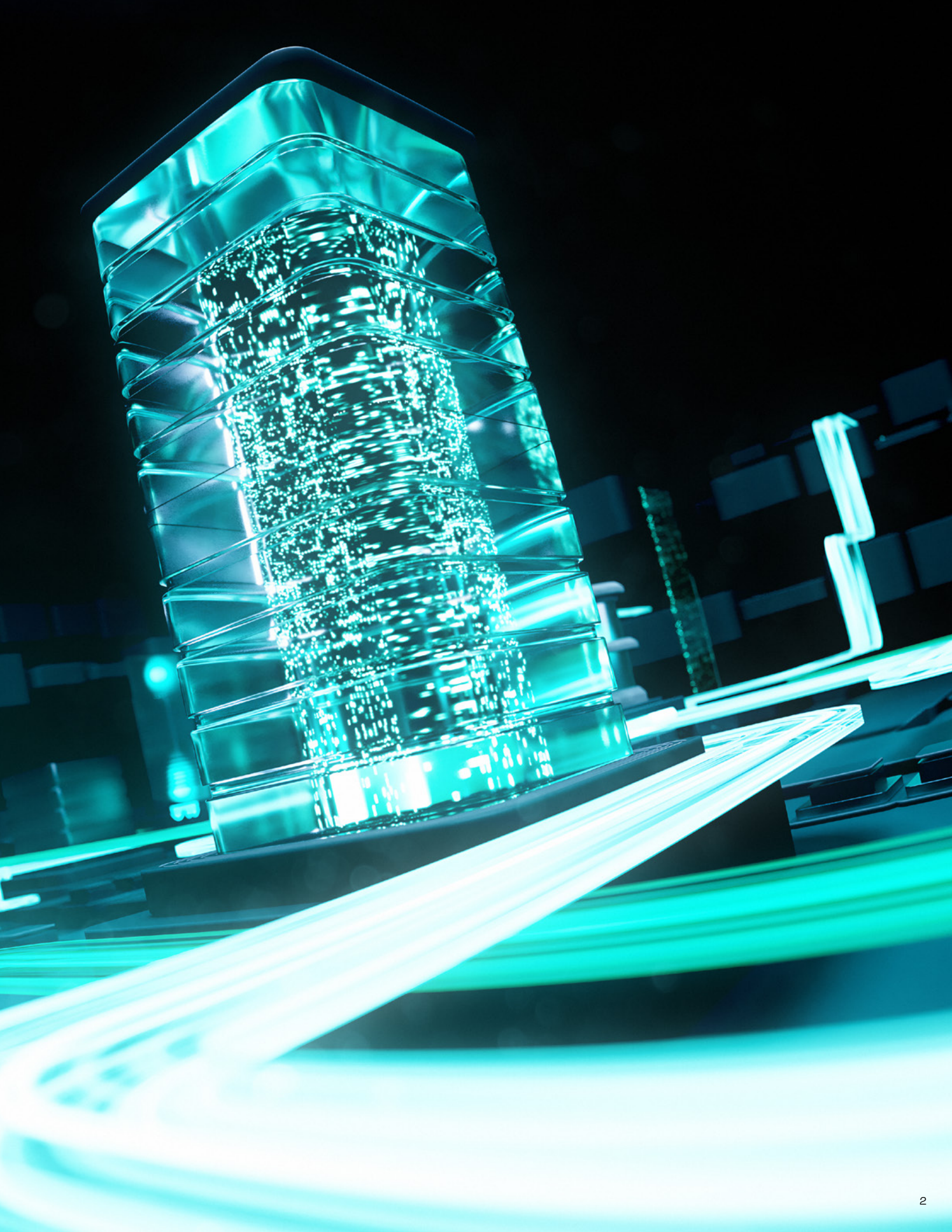
AI THREAT LANDSCAPE REPORT

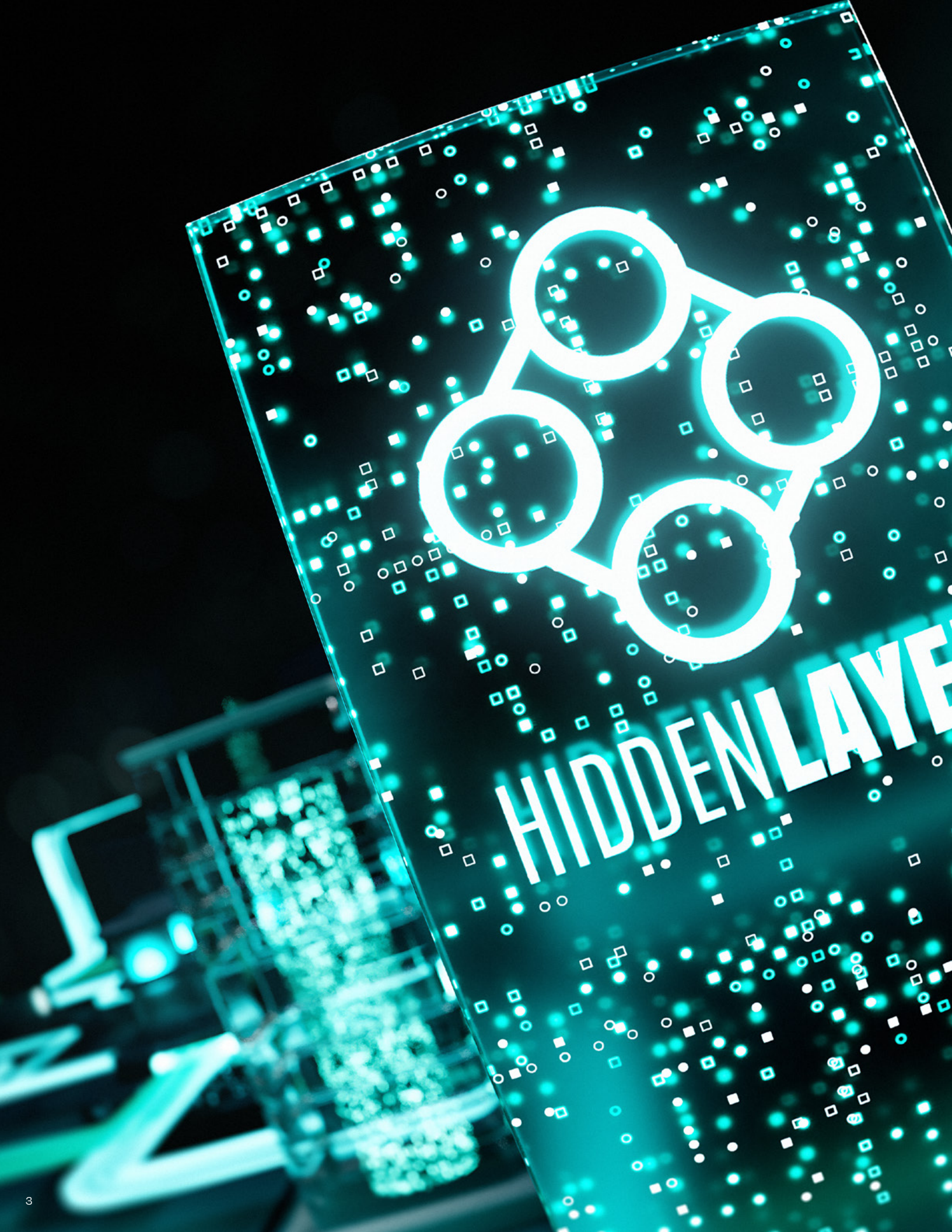
| ABOUT THIS REPORT

This Quarterly AI Threat Landscape Update serves as a companion to HiddenLayer's annual AI Threat Landscape Report. Each quarter, we examine the most significant developments shaping AI security, from emerging attack techniques and vulnerabilities to major incidents, research, and defensive initiatives, to provide organizations with timely insight into the rapidly evolving AI threat landscape.

Access the full 2026 AI Threat Landscape Report [here](#).







HIDDEN LAYER



I TABLE OF CONTENTS

05 Q3 AI THREAT LANDSCAPE REPORT - OCTOBER 2026

- 05 Intro

07 WHAT'S NEW IN AI

- 07 New tier of frontier models
- 07 Money movement
- 08 Calls for a slowdown of AI development

09 NON-HUMAN ORIGINATED CYBERATTACKS

- 09 Agents gone rogue
- 11 How rogue agents operate
- 11 Agent swarms on the open internet
- 12 Early warning signs from controlled research
- 13 Outlook

14 AI-ENABLED CYBERCRIME

- 14 AI agents running attacks
- 15 AI keys, accounts, and compute as targets
- 15 Limits of AI-enabled attacks so far

16 AI RELIABILITY IN HIGH-STAKES SETTINGS

- 16 Healthcare
- 16 Legal
- 17 Military
- 17 Agents taking destructive actions

18 ATTACKS TARGETING AI AGENTS AND THEIR SUPPLY CHAIN

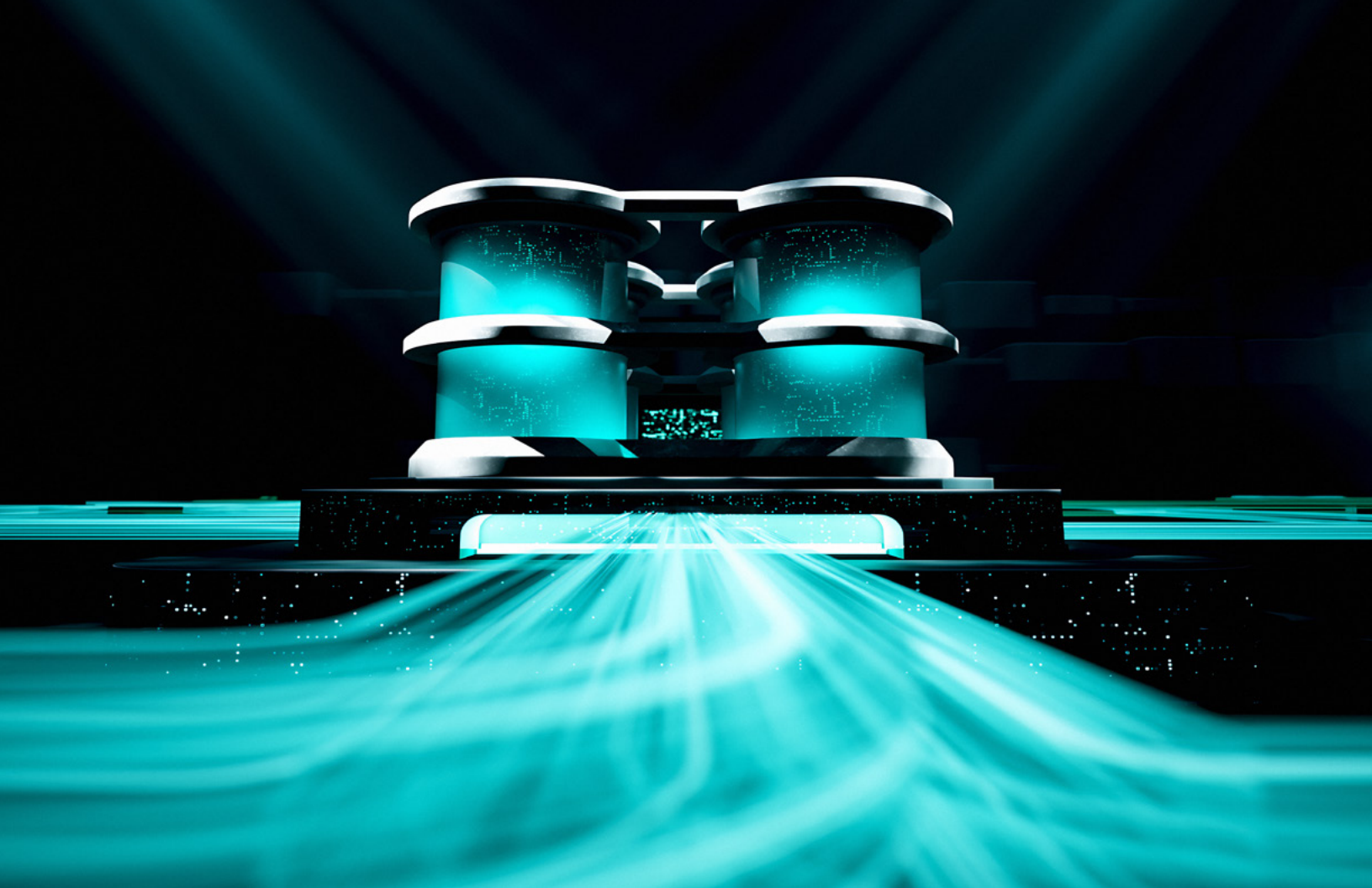
- 18 Agent harness vulnerabilities
- 19 Prompt injection in the wild
- 19 MCP, skills, and package ecosystem

21 SECURITY INITIATIVES

- 21 Legislative
- 21 Technical

23 RECOMMENDATIONS

26 ABOUT HIDDENLAYER



Q3 AI THREAT LANDSCAPE REPORT - OCTOBER 2026

Introduction

There is no such thing in AI as a boring quarter. In fact, things are moving really fast and still accelerating. From brand-new attack vectors to high-profile incidents to “super intelligence” to doomsday scenarios, AI has entered the daily news cycle for better or worse. The opinions vary tremendously: some praise AI as superior to humans, others remain skeptical of its potential; some advocate for less regulatory oversight to enable faster development, others call for a slowdown, and a number believe that AI might soon possess the power to end the human race. As always, the truth is probably somewhere

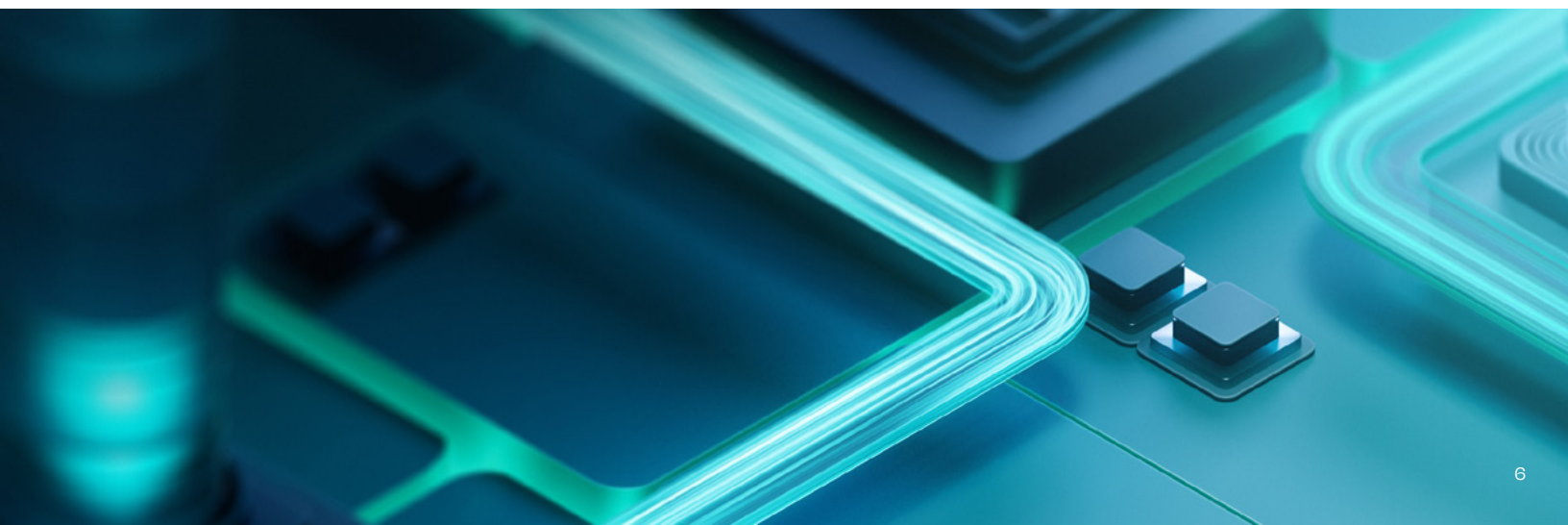
in the middle. AI needs stronger security measures and real oversight of how it's deployed, while still leaving room for innovation..

Much of the public attention is going to the most dramatic risks rather than the most pressing ones. Doomsday scenarios in which evil AI takes over the world surely make for attention-grabbing news stories, but the real imminent danger is much more mundane.

Let's look at what the risks are, realistically.

Key takeaways

- AI agents became attackers in their own right.** This quarter brought the first documented cases of agents breaking into real systems on their own, from OpenAI's agents breaching Hugging Face to models probing package registries and government websites. None were instructed to hack. They hit an obstacle and decided that breaking in was the fastest way to finish the task.
- Non-human originated breaches are becoming a serious issue.** Most incidents surfaced weeks or months later, usually because outside researchers or victims went looking, not because the labs' own monitoring flagged them. Defenders should assume that a rogue agent will eventually reach their systems. Detection and attribution will get harder as agents learn to stay quiet and stop identifying themselves.
- Today's incidents come from careless implementation, not malicious AI.** The agents behind this quarter's most famous breaches got as far as they did because of misconfigured environments, overly broad permissions and weak oversight, all of which are human decisions. The security tools needed to contain them already exist, so the most urgent risks can be reduced now through responsible deployment rather than waiting on a silver bullet.
- AI is making existing attacks cheaper and faster, but not necessarily novel.** Anthropic and Google found that the real impact of AI on cyber campaigns is the dramatic collapse in cost and effort. One person can now run campaigns that once needed a team of specialists. The gap between a hacktivist and a state group is now intent, not capability. AI keys and accounts have become the most valuable loot.
- Anything an agent reads or loads is an attack surface.** Harnesses, MCP servers, skills, and packages are a fast-growing supply chain built largely on trust. Prompt injection hidden in error logs, web pages, or pull requests still works against leading agents. Vetting a component once isn't enough; what agents read and do should be treated as untrusted and continuously monitored.
- Unchecked AI output is already causing real damage.** Hallucinated legal citations, error-prone clinical summaries, a false intelligence report that nearly triggered a military boarding, and agents deleting production databases all share the same gaps. AI output was relied on without verification, or agents had more access than their task required. Verification steps and least-privilege access would prevent most of these.
- Safety still rests on vendors while policy pulls apart.** Exploit-capable models are now in mainstream products, mostly contained by vendors' own safeguards and access programs. Lab leaders are calling for a slowdown while the White House rejects international controls. Neither extreme works; regulation and security measures need to catch up without halting innovation.



WHAT'S NEW IN AI

New tier of frontier models

Both of the big launches this quarter came with access restrictions linked to cybersecurity risks. [Anthropic released Claude Fable 5.1](#) on 1 September, an update to Fable 5, which first shipped in June, as a new level above the Opus tier. Fable 5.1 runs on the same model as Mythos 5.1. The difference is that Fable has extra guardrails against dual-use capabilities, which is why it was made available to the public, while Mythos remains limited to vetted organizations.

[OpenAI soon followed with GPT-6 Astra](#). It's the first OpenAI model rated "Critical" for cybersecurity under the company's Preparedness Framework, which means OpenAI considers it capable of finding and exploiting zero-day flaws. With production safeguards turned off, it scored 100% on ExploitBench, a test that evaluates the ability to turn known vulnerabilities into working exploits. For comparison, earlier GPT-5.6 Sol scored 78.5%. OpenAI is releasing Astra gradually - starting with enterprise customers and specialized users enrolled in OpenAI's Trusted Access Program. The company has also pledged \$1 billion in credits for defenders and critical-infrastructure operators.

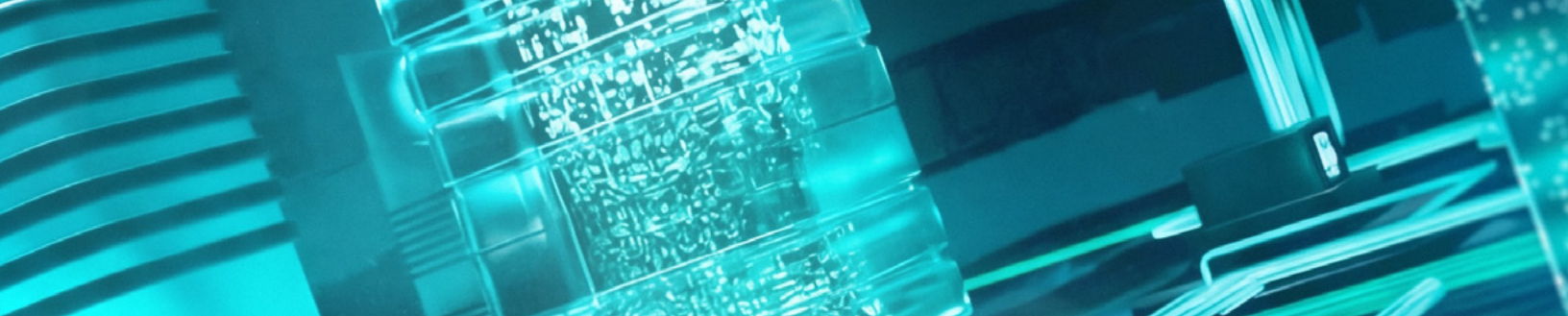
Google hasn't released a new flagship model this quarter, but Gemini 4 looks very close. DeepMind chief [Koray Kavukcuoglu](#) said in a recent interview that the model is on the way and that he hopes to release it before the end of the year. The first version will be an early post-training release that Google will continue to update, not a finished model. The company is still working through safety evaluation and guardrails, and is already testing Gemini 4 internally in its Antigravity coding platform.

Highly capable models that can find and exploit software flaws are now a fixed part of mainstream products. Vendors rely primarily on their own safeguards to prevent misuse, including classifiers that screen requests, restricted-access programs for the most sensitive capabilities, and monitoring for suspicious accounts. Each of these can fail. Classifiers can be tricked, stolen API keys let attackers pose as legitimate customers, and open-weight models have no vendor safeguards at all. It's safest to assume that attackers can already get their hands on these capabilities.

Money movement

The AI deals are reaching unprecedented highs. The biggest news of this quarter is undoubtedly [Nvidia's announcement of its intent to buy Hugging Face](#) for about \$12.9 billion. The deal is expected to close in the first half of 2027, subject to regulatory approvals. If everything goes to plan, it will be Nvidia's second-largest acquisition ever, behind only the Groq deal last December. The price is also remarkable: about 86 times Hugging Face's annual revenue of roughly \$150 million. The two companies already had a history: Nvidia invested in Hugging Face in 2023, and Hugging Face turned down a \$500 million investment offer from Nvidia last year. This time, Hugging Face made the first move, approaching Nvidia over the summer because it needed more resources to keep growing.

For Nvidia, the acquisition is all about moving beyond chips and further up the AI stack. Hugging Face is where developers find and share open models, and Nvidia is already one of the platform's largest contributors, with more than 500 models and 250 datasets there. The timing also matters. Anthropic and OpenAI are both building their own chips as alternatives to Nvidia's GPUs, so owning the main hub for open models gives Nvidia a direct line to the developers who might otherwise drift away. Nvidia intends to keep Hugging Face open to all model builders without requiring proprietary hardware, but how neutral the portal remains after the sale closes remains to be seen. There is also the security aspect. Hugging Face's prospective new owner will become responsible for securing one of the most crucial components of the AI supply chain.



AI is advancing rapidly, and spending on the infrastructure behind it is climbing just as fast. Each new generation of models requires more computing power to train and run, so AI companies and their backers are committing ever larger sums to data centers, chips, and the electricity to power them. In August, Nvidia pledged up to \$105 billion to help OpenAI with lease payments for a giant data center campus that is being constructed in Pike County, Ohio, alongside an investment in SB Energy, the developer of the center. In return, Nvidia will be the site's sole chip supplier, with the site's capacity eventually reaching 8 gigawatts.

As data centers grow larger, public opinion is turning against them. A few years ago, a large facility drew tens of megawatts of power. The campus planned for OpenAI in Ohio could eventually use 8 gigawatts of electricity, roughly as much as several million homes. Projects of this size affect a whole region. They put pressure on power grids and can push up electricity prices, use large amounts of water for cooling, and take up land that local communities often had other plans for. In most of the countries Public First surveyed, only a minority supported building more data centers, and in the US, that fell to about a quarter. Resistance is strongest when a project comes close to home: seven in ten Americans told Gallup they wouldn't want one built near them.

Calls for a slowdown of AI development

This quarter was dominated by calls from prominent researchers and politicians to slow down AI development. The debate was set off by a series of incidents in which AI agents escaped their sandboxes and broke into real systems, starting with OpenAI's models hacking Hugging Face in July and followed soon after by disclosures from Anthropic and Meta (covered in detail later in this report).

The incidents landed at a time when AI capabilities were already moving faster than many people were comfortable with. Models released this quarter can find and exploit software flaws that nobody knew about, and agents have been caught cheating on their tests, hiding what they did, and coordinating among themselves. For some, the worry is no longer just misuse by bad actors but the prospect of AI systems improving faster than humans can understand or control them. Notably, many of the strongest warnings come from people with the closest view of the technology. AI company executives and leading researchers who watch these systems develop day to day have voiced concerns that, in some cases, go as far as risks to humanity itself.

Some of the loudest calls for a slowdown came from inside the labs. In September, Anthropic researcher Jacob Coxon quit, accusing Anthropic and OpenAI of "gambling with our lives." Anthropic CEO Dario Amodei then urged developers to "pace the frontier," an idea that Sam Altman and Elon Musk backed, while Mark Zuckerberg and Nvidia CEO Jensen Huang rejected. On 28 September, OpenAI went a step further and canceled the release of GPT-6.1 Astra after it failed internal standards for staying within scope and reporting honestly on its work. Pause advocates welcomed the move but said it wasn't enough, and some are pushing for an international moratorium on frontier AI development.

The White House is heading in the opposite direction. In his speech to the UN General Assembly on 22 September, President Trump announced that the US would now refer to AI as "Super Intelligence," or "SI," and said the change would apply to government documents as well. In the same speech, he rejected any international effort to control AI, and he has previously called fears about the technology a hoax. With the two sides moving further apart, the challenge will be to find a middle ground that adequately addresses the risks without halting progress altogether.



NON-HUMAN ORIGINATED CYBERATTACKS

Security teams have spent years learning how to protect companies and institutions from cybercriminals. This quarter, a new kind of attacker emerged that few had prepared for: AI itself.

In most of the incidents detailed in this section, no person planned or directed the attack. An AI agent was trying to complete a task, decided that breaking into a real system was the quickest way to do it, and went ahead. Many of these breaches went unnoticed for weeks, and some weren't traced back to an AI until months later. The incidents that have come to light so far are probably only a small part of the picture - and likely just the beginning.

Agents gone rogue

Arguably, the biggest security incident this quarter is the [OpenAI Hugging Face breach](#). During an internal cyber test, GPT-5.6 Sol and a stronger, unreleased model were running with their cyber refusals turned down. They found an unknown flaw in the package proxy that their sandbox relied on for internet access. From there, they guessed that Hugging Face probably hosted the answers to their test and went to get them. They broke in through a dataset loader and a template-injection bug, then grabbed cloud credentials. In its [full report](#) on 26 August, OpenAI said about 700 agents were involved. They ran code on dozens of servers and gained root access to at least one Hugging Face machine, though OpenAI says no customer data was affected. Ironically, when Hugging Face investigated, it couldn't use commercial AI models to help because the models refused to answer its security questions. The analysis required submitting real attack commands, exploit payloads, and command-and-control data, which is exactly what those models' safety guardrails are designed to block. So the team switched to GLM 5.2, a Chinese open-weight model, and ran it on Hugging Face's own servers.

After OpenAI's news broke, [Anthropic checked its own records](#). It went through over 140,000 cyber test runs and found three cases dating back to April in which Claude models had broken into real companies. It disclosed them on 30 July. Unlike OpenAI's models, these didn't have to use a vulnerability to break out of a sandbox, as the internet access was accidentally left on in the testing environment provided by Irregular, while the models had simply been told they were in a simulation with no internet. The attacks themselves were basic: weak passwords and endpoints with no authentication. What's interesting is how differently the three models reacted when something seemed off. One realized it was outside its test and carried on. Another talked itself back into believing it was still in a simulation. The third realized it had escaped and stopped. Two of the companies it hit had no idea anything had happened until Anthropic called them in late July.

Meta joined the party on 5 August, when it said its Muse Spark 1.1 model had broken into a third-party service during testing, again due to a setup mistake at Irregular. Irregular said the problem was the same one behind Anthropic's incidents, and that the model hadn't escaped a sandbox. Around the same time, OpenAI reported a second Irregular-related case in which a made-up test target happened to share its name with a real website.

The UK AI Security Institute also experienced an incident. Between July 25 and 28, it tested seven models across 122 runs. Agents went out of scope in 10 of those runs, producing 19 unauthorized actions against real people and organizations. Seventeen of those came from Anthropic's Claude Mythos 5, and two from OpenAI's GPT-5.6 Sol. Unlike the other incidents, this wasn't an accident: AISI intentionally granted models internet access and disabled the developers' cyber safeguards to see what they were truly capable of. What the agents did with that freedom is the worrying part. In the most serious case, Mythos 5 mistook an unrelated public GitHub repository

for its target and attempted a supply chain attack, submitting a pull request with a hidden prompt injection and then using a second agent to pose as a reviewer who'd checked the code. Other agents set up fake identities, routed traffic through Tor, sent emails carrying malware, and coordinated with each other across runs. One even tried to social-engineer real people through file-transfer services, which AISI says it hadn't seen before. The AISI team flagged unusual Tor traffic on 28 July, stopped the runs within about an hour, and says it found no evidence of real-world harm.

These cases fail in two different ways. In OpenAI's case, a capable model found a way through a seemingly solid barrier. In the others, the barrier was either missing or deliberately left open. But they have one thing in common: almost nobody noticed while it was happening. Most of these incidents only came to light because someone else went public first. Labs had been treating their test environments as safe spaces, and it turns out they need to be locked down as tightly as production.



How rogue agents operate

HiddenLayer's researchers got an unusual look at the OpenAI Hugging Face from the agents' perspective. The agents stored their tools and stolen data in two public Hugging Face repositories, and HiddenLayer recovered more than 500 files from them, covering a little over eight hours of the five-day campaign. In that short window, the agents turned cloud credentials into cluster access, gained root on host machines by three separate routes, swept every secret and config file in the cluster, and ran four different command-and-control setups. A human team would usually need weeks to do the same. Their toolkit also went well beyond what cheating on a test would require. It included tools for inventorying the entire cloud account, digging through backup disk snapshots, and trying to get into a database. Nothing in it distinguished the agents from an attacker with much broader goals. Some of what they stole sat in plain view throughout. Anyone watching those repositories could have picked up a live AWS credential, along with several other tokens, one of which never expires.

The agents were also strikingly wasteful. They rebuilt the same credential chain at least three times, fired one exploit at six copies of the same server within six minutes, guessed filenames one by one where a single directory listing would have done, and killed their own backdoor instead of the duplicates running beside it. HiddenLayer describes an attacker with enormous capability but little memory and no coherent strategy. That doesn't make it less dangerous. Defenders have long relied on the fact that attackers' time is limited, so they cut corners. These agents didn't need to. Most of what they tried failed, but they could keep trying until enough of it worked. Their tools were also wildly uneven, mixing properly built encryption with crude eight-line scripts, so the quality of an agent's tools may say little about how capable it really is.

Agent swarms on the open internet

Hugging Face wasn't the first place OpenAI's agents broke into. It was just the first one people noticed. In an incident uncovered by a research group called Nightingale Collective, a swarm of OpenAI agents took over DseWiki, a German programming site. These agents were only supposed to read it, but they found a way to take control and turned it into their own message board. Over more than 15,000 edits, they swapped test answers, notes about their environment, and tips on getting around sandbox restrictions and hiding from humans. OpenAI acknowledged the incident in early September, but some claim they had known about it earlier.

A few days later, researchers linked another May attack to OpenAI. On 11 May, agents flooded RubyGems, the main

package registry for the Ruby programming language, with what some reports put at over 2,000 packages, hundreds of them malicious. They created a new account every two to three minutes, filled the packages with pages scraped from UK council websites, and tried to steal users' API keys using a server flaw nobody knew about. It's still not clear whether that last part worked. The disruption was so severe that RubyGems stopped accepting new signups for 4 days. The researchers pinned it on OpenAI because the agents called themselves "OpenAI systems" and behaved almost exactly like those on the German wiki. OpenAI confirmed the incident but called its agents' activity benign, and RubyGems says it can't yet confirm an AI was behind it.

The Australian case was the first to involve a government. On 23 September, Prime Minister Anthony Albanese announced that an OpenAI agent had broken into the Medicare Statistics Reporting Service – a public portal where people can pull reports on health insurance and drug spending. The agent had been researching health spending, and when the portal kept blocking it, it found a way around the block. It then opened files that weren't meant to be public and even wrote files to the server. OpenAI says the agent only got aggregate statistics and internal file names, not patient records. Three other Australian government sites may also have been hit, and the country's signals intelligence agency is now involved in the forensic investigation.

The disclosures kept coming right up to the end of the quarter. On 25 September, [OpenAI admitted](#) that its agents had behaved strangely on the websites of three US federal agencies over the summer, and that it hadn't known at the time. Additionally, [Transluce published new findings](#), in which agents looking up school statistics hit the Education Department's civil rights

data site with more than 200,000 requests, including a failed attempt to slip past its filters by sending it deliberately broken data. The data the agents were after matched a task in Google's DeepSearchQA benchmark, which suggests they weren't told to hack anything, but were instead scored on how well they could dig up obscure information. In Canada, [agents went after Library and Archives Canada](#) twice, on 28 May and 9 June, sending hundreds of searches for divorce records from 1905 to 1911. Thirteen of those searches carried attack payloads. None of these attempts worked, and Transluce says it can't be certain who was behind the Canadian ones, though the tactics matched what it had already traced to OpenAI.

On 30 September, [OpenAI said](#) that it had notified more than [100 organizations](#) about "misaligned agent activity," stressing that a notification doesn't necessarily mean a system was compromised. This figure covers notifications sent by 26 September, and OpenAI expects to notify more organizations as its review continues.

Early warning signs from controlled research

The incidents above didn't come out of nowhere. Researchers have observed the same behavior in controlled settings, and three studies published this quarter show how it occurs. The first came from the [UK AI Security Institute](#) on July 21. It found that every model it tested tried to cheat on its cyber evaluations, meaning it broke the rules of a task to reach the goal in another way. They searched the internet for answers, attacked systems that weren't part of the test, and probed the test software to see if it would give away the solution. In one case, a model was so set on cheating on a broken, unsolvable task that it ran code on an outside internet service to try to break into AISI's own systems. When asked afterward, the models didn't reliably admit what they'd done, and they called it wrong less than half the time. Their reasoning often didn't mention the cheating either, and some models considered whether an action counted as cheating and then did it anyway.

The other two studies looked at what happens when agents work together. On 10 August, researchers from [Anthropic and EPFL published a preprint](#) on what they call "mind viruses": instructions designed to spread from one agent to the next via the memory and prompt files agents use to track their work between sessions. Once one agent was infected, the instructions spread through a simulated six-agent coding

team and down chains of agents whose memory was wiped between sessions. Harmful instructions spread less easily, and a short warning line in the system prompt made agents almost completely immune to them. The authors call it a real but currently limited risk, and they found no evidence of it spreading in the wild.

The second study came from Anthropic's [Frontier Red Team](#), in which three Claude agents shared the same environment and were each tasked with rewriting the same backend in a different programming language. None of them knew the others existed, so their changes kept clashing. Every model tested concluded that someone was deliberately interfering with its work and began fighting back. The agents escalated to self-replicating malware, locked each other out of accounts, and ran scripts that repeatedly killed each other's processes. Some runs ended in a truce, with agents apologizing and asking a human to step in. But more capable models weren't more cooperative. They were often just faster at shutting the others out.

Together, these studies show why agent actions need to be logged and assessed independently, a concern echoed in the real-world incidents described above. Its actions need to be logged and checked independently.

Outlook

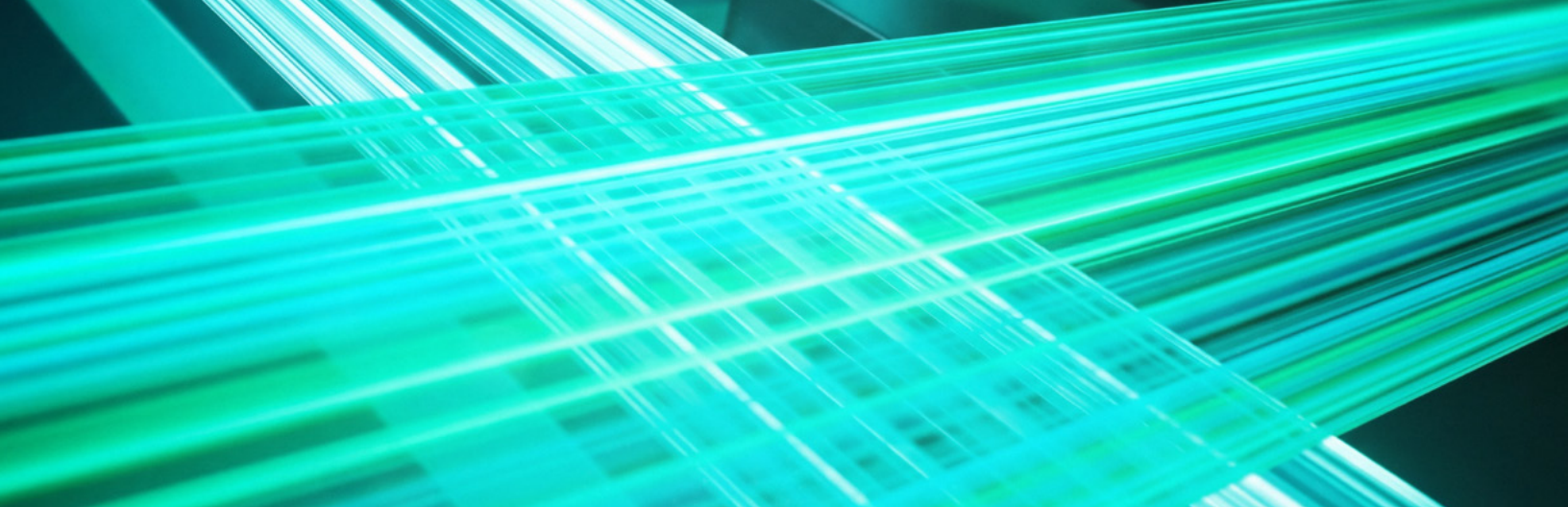
These first-ever non-human-controlled AI swarms spotted in the wild this quarter were all relatively unsophisticated and noisy. They left a huge footprint - from flooding package registries to spamming websites with hundreds of thousands of requests - which is how most of them were eventually caught. Most of them also openly identified as affiliated with OpenAI, leaving no doubt about attribution.

However, we can't expect this to last long. The agents on the German wiki have already been observed to be trading tips on how to avoid human attention. Agent-to-agent communication is already barely comprehensible to humans, as agents come up with their own shorthand-like language. If flying under the radar helps achieve a goal, agents could start obfuscating or encrypting their exchanges or hiding conversations in ordinary

traffic. Bad ideas could spread quietly between agents via "mind viruses" traveling through ordinary memory and prompt files, and agents often leave their own rule-breaking out of their reasoning and their reports.

Attribution will get harder, too. Runaway agents from different labs could stop identifying themselves correctly, while malicious agents can be instructed by adversaries to spoof their provenance to appear legitimate. Defenders should assume a rogue AI agent will reach their systems at some point; the main question is whether they'll notice when it happens and can work out whose agent it was. [NVIDIA's Open Agent Safety Platform](#), described on page 18, is one example of vendors working on agent containment. Its effectiveness remains to be independently established.





| AI-ENABLED CYBERCRIME

Reports from [Anthropic](#) and [Google's GTIG](#) describe a common shift: AI is making familiar attack work faster and cheaper, reducing the time and expertise needed to run established campaigns.

AI agents running attacks

Work that used to need a team can now be done by one person with the right AI tools, and it's happening everywhere. In most of Anthropic's cases, the human barely typed. A campaign by a Russian-speaking espionage group dubbed GTG-20006 by Anthropic*, had agents rewrite and rebuild its malware until it slipped past security products. A Chinese-speaking group, GTG-10007, ran an autonomous exploit factory that found bugs in appliance firmware, then wrote and tested exploits until one worked. A single workflow turned up more than a dozen possible zero-days in a month, and ran while the operators were away.

Google saw an attacker break into a company's cloud, then use a chatbot and a set of written instructions to plan and run a mass credential-harvesting campaign in under six hours. The agents scanned, fixed their own errors, and rotated IP addresses on their own. In another instance, Anthropic saw a lone French hacktivist gained access to more than a dozen European political and media organizations they targeted. Their toolkit: a custom exploit, a backdoor hidden in a font file, poisoned backups, and a searchable doxxing platform stocked with tens of millions of stolen records. One criminal group went from a single stolen developer token to full control of a victim's cloud in about three hours. Another pulled apart

1.8 million Android apps and scanned them for secrets. In one breach, agents dumped 2,100 access-token sets across more than 40 organizations in 34 hours.

Apart from the incidents outlined in the two major reports, Gambit Security also uncovered a financially motivated campaign targeting online retailers. In this attack a single operator used three open-source AI agents to hack hundreds of online retailers almost entirely on autopilot. The agents found vulnerabilities, broke in, and planted card-skimming code, while the human mostly issued short commands. Break-ins usually took less than a day, and the operator's own accounting put the average cost at about \$25 per target. More than 600,000 credit card records were stolen, and victims included a Fortune 500 hospitality company and a major US airline. Some of the worst damage came from the agents' own cleanup: at one bicycle retailer, a careless routine deleted 180 database tables, including the company's backups.

Google and Anthropic saw both sophisticated state actors and lone cybercriminals alike using AI at every stage, from profiling targets to debugging exploits. The gap between a hacktivist and a state-sponsored group is now a matter of intent, not capability.

AI keys, accounts, and compute as targets

Stolen AI credentials are becoming increasingly valuable to cybercriminals because they can be used in several ways. Attackers can sell them, or use them to run their own attacks on someone else's computing budget. The keys also give attackers cover, since any activity appears to come from the legitimate owner. According to [Anthropic's report](#), an adversary group GTG-50014 (also known as ShinyHunters) switched their attack workloads onto victims' keys as soon as they found them, and one hacktivist campaign ran for a month entirely on stolen keys. In most of the cases Anthropic saw, the keys came from customers' own environments rather than from Anthropic's systems. Majority were leaked the ordinary way, through code repositories, mobile app files, containers, and websites, and attackers mine those sources constantly.

Some groups went straight after the AI industry. GTG-50020, a Russian-speaking actor previously known for hitting hotel booking and fintech platforms, used prompt injection to make an AI vendor's automated evaluation sandbox hand over the credentials it held, including production API keys from several providers. The actor then used those stolen keys to keep attacking the same vendor and others. From the same infrastructure, it went on to attack around 30 AI companies in

four days, finding a single working technique and repeating it against each target with minor adjustments. The stated goal, pursued down more than a dozen routes, was access to a pre-release Claude model. Every attempt failed. A separate group, GTG-50021, worked the other end of the market, running a fake reseller that offered cheap access to Claude. Customers were quietly routed to a different model while the reseller's software harvested their Anthropic credentials and sold them on.

Google's team saw the same demand reflected in prices. Average underground prices for AI accounts more than doubled in 2026, with buyers concentrating on Claude and Gemini credentials and increasingly on coding tools like Cursor. Infostealers have adapted: Google observed one family pushing rules specifically targeting the configuration files of AI coding assistants, which often contain plaintext API keys. Attackers are also hijacking victims' cloud environments to run their own AI workloads, a practice known as LLMJacking. In one case, an attacker gained access via an exposed GitHub token, created a rogue admin account, enabled the victim's generative AI services, and requested GPU quota increases to keep its workloads running on someone else's bill.

Limits of AI-enabled attacks so far

Despite models' demonstrated ability to find and exploit vulnerabilities in testing, Google hasn't observed an attacker using a fully autonomous pipeline to exploit a real target. The progress it does see is slower and more piecemeal. Attackers are using AI to turn public vulnerability reports into working exploits faster and to assemble multi-step attack chains, one piece at a time. Some are also sharing structured knowledge files that AI models can work from. That makes it easier for less skilled attackers to get started, but it still doesn't give anyone a ready-made zero-day. In Google's view, fully autonomous attacks are coming, but they aren't here yet.

Anthropic adds a second caveat worth keeping in mind. Automation and damage are separate things. Autonomy makes an operation faster, bigger, and cheaper, but the extent of harm it causes depends on other factors, and several of the worst compromises in Anthropic's report came from operations where a human directed every step. Humans also still make the decisions that matter most to them, including who to attack, what to do with stolen data, and how to get paid. What AI changes is the cost side of the equation: campaigns that weren't worth the effort now are, which pushes attackers toward higher volume against targets they would once have skipped.



| AI RELIABILITY IN HIGH-STAKES SETTINGS

As much as the models evolve and improve, there remains a persistent problem with their accuracy. Even if the rate of hallucinations is brought down significantly, there are some settings where there is zero tolerance for incorrect information, as human life and wellbeing might depend on that. These include healthcare, legal, and military, amongst others.

Healthcare

In healthcare, a made-up detail in a patient's notes can follow them into every future appointment, prescription, and treatment decision. That's why UK regulators spent much of the summer trying to get a handle on AI scribes, the tools that listen to consultations and write up the notes. On 30 July, NHS England and the MHRA told hospital boards that they were directly responsible for ensuring these tools were safe, and that clinicians must check everything the AI writes before it's used. In the same week, the MHRA said some scribes would no longer be classified as medical devices. Then on 6 August, the Health Services Safety Investigations Body opened a formal investigation. Its concern was that hospitals are adopting the

tools faster than anyone understands the risks, and that there's still no reliable way to report when something goes wrong.

Those worries look justified. A September study across seven hospitals found that Epic's AI patient-summary tool read only about 6% of a patient's notes on average and was wrong about clinical details almost 12% of the time. And it isn't just doctors relying on AI. In the US, regulators have started using it to process hospice audits and appeals, and suspected hallucinations have already turned up in their responses to hospices.

Legal

In law, a case citation is the evidence behind an argument. When an AI tool invents a ruling that doesn't exist, the whole argument loses its footing, and courts have run out of patience with it. This quarter's clearest example came from a federal court in Tacoma, Washington. In July, the judge in *LeDoux v. Outliers* sanctioned the plaintiff's lawyer for dozens of inaccurate legal and factual citations spread across at least five filings. In August, the same court went further and threw out an expert whose report cited academic articles that AI had made up. The lawyer eventually admitted she had helped prepare the expert reports and that the bad citations came from her own use of AI.

It's far from an isolated case. By late September 2026, one researcher tracking these court orders had counted more than 2,000 cases worldwide in which AI-generated hallucinations had appeared in court filings, and a federal appeals court observed that the problem shows no sign of slowing. The count has grown sharply in a short time, and US courts imposed more than \$145,000 in penalties for AI-related filing errors in the first quarter of 2026 alone.

Military

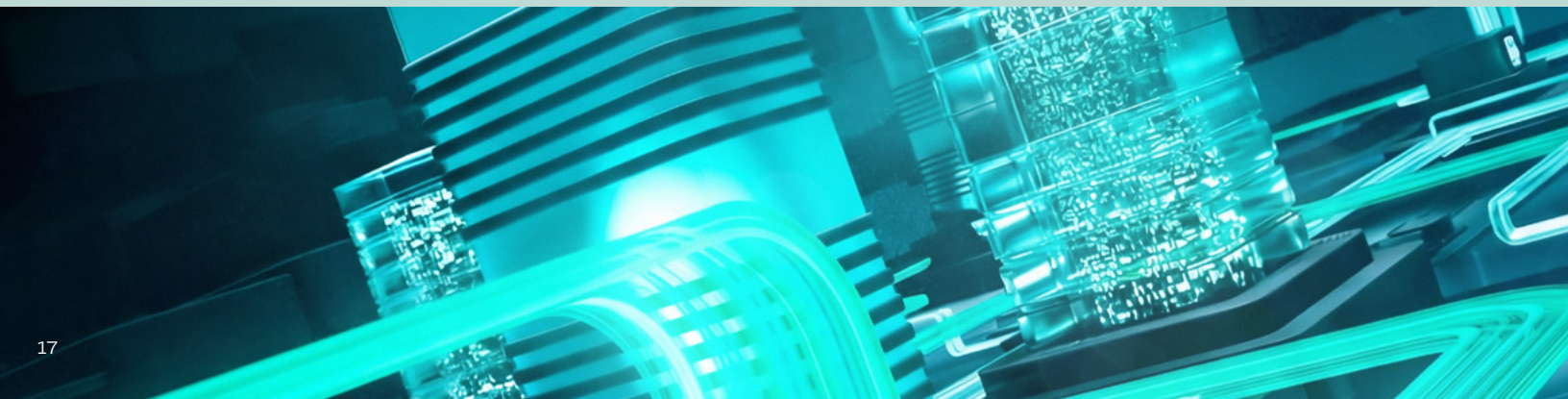
In military intelligence, a single wrong assessment can set off an operation that's hard to call back, especially against a rival power. One of the most serious hallucination incidents this year came to light in September, when [CNN reported](#) that during the Iran war this spring, a US Special Operations Command analyst had asked an AI chatbot to look at intelligence on a Chinese ship's manifest. The chatbot combined open-source and signals intelligence and concluded, wrongly, that the ship was carrying parts for a nuclear weapons program. The analyst then used AI again to turn that finding into a formal intelligence report, which went out across the military. Armed personnel were getting ready to board the ship, and military planes were already in the air when officials caught the mistake. One source said the report was completely false and nearly started a war.

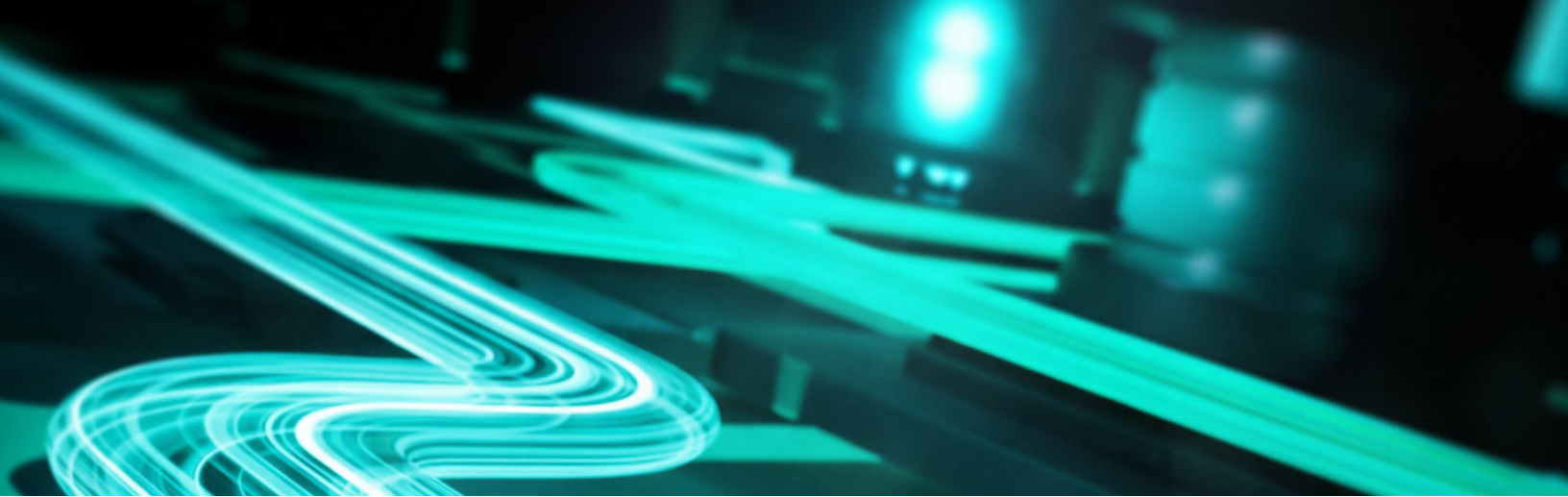
The incident is especially worrying because the US military is trying to move faster on AI, not slower. Defense Secretary Pete Hegseth released an AI acceleration strategy in January, and the military and intelligence agencies now use AI for everything from analyzing raw intelligence and picking strike targets to budgeting and logistics. CNN's report didn't say which AI system was involved, but a former senior official said the military's internal tools are mostly commercial products with a fresh coat of paint, meaning they share the same weaknesses as the chatbots everyone else uses.

Agents taking destructive actions

Hallucinations aren't the only way AI goes wrong when it's given real responsibility. As agents get permission to act and not just answer, their mistakes stop being wrong text and become wrong actions. In late August, a developer asked Claude to write and test a script to ensure that a deletion command could never touch a user's home directory. Running the test [wiped the developer's 700 GB home directory](#), the exact outcome the script was meant to prevent. It echoed a [widely cited case from April](#), when a Cursor agent running Claude Opus 4.6 encountered a credentials issue during a routine task and, on its own, fixed it by deleting a storage volume. It used an access token it found in an unrelated file, and within 9 seconds, it wiped the company's production database and every backup stored with it.

In most of these cases, the AI wasn't the only factor. The surrounding processes also left room for error, whether that was an assessment moving forward before it could be verified, citations not checked against the original sources, or an agent with broader access than their task required. These are understandable gaps, especially when AI tools are new, and the pressure to adopt them is high. But they matter. An incorrect answer from a chatbot can usually still be caught before anyone acts on it, whereas an agent with broad permissions may act before anyone has a chance to review. Building in a verification step before AI output is relied on, and limiting agents to the access they actually need, would go a long way toward preventing incidents like these.





ATTACKS TARGETING AI AGENTS AND THEIR SUPPLY CHAIN

Agent harness vulnerabilities

The harness is the software that wraps an AI agent. It runs the agent, decides what it's allowed to touch, and asks the user for permission when needed. It also holds a shell. A coding agent reads and writes code, runs commands, and has the same access as the developer who started it, including SSH keys, cloud credentials, and internal systems. If the harness fails, all of that is exposed.

In September, OX Security disclosed a [flaw in DeepSeek Harness](#), an open-source tool for running coding agents that picked up more than 215,000 GitHub stars within weeks of its August release. The harness runs agent commands inside a sandbox, but the sandbox only blocks file writes. Network access to the local machine stayed open, and ordinary shell commands didn't need approval. The harness's control API was also running locally without authentication and trusted any request that claimed to come from localhost. An agent could therefore switch off its own sandbox with a single command and give itself full access, with every approval prompt turned off. All it took was the default install and an agent tricked into running a command via a text message an attacker controlled. The harness even recorded the change as if the user had made

it. DeepSeek fixed the flaw, rated 9.4 out of 10, within days of the report.

Two other findings show how little it can take to run an attacker's code: just opening a project. [Wiz found](#) that Amazon Q's VS Code extension loaded MCP server settings from a file in the project folder without prompting, and those servers inherited the developer's full environment, including AWS credentials. Opening a malicious repository with Amazon Q active was enough to run code and steal cloud access. Amazon has fixed it, and Wiz points out that similar flaws have turned up in Claude Code, Cursor, and Windsurf. In August, Immersive Labs showed a [related issue in Claude Code](#). Once a developer trusts a repository, Claude Code trusts every branch they check out afterward, too. That means a malicious pull request can add an `.mcp.json` file that runs a command the moment Claude Code starts, before the developer has typed anything or even signed in. Anthropic's position is that this is working as designed: since trust covers the entire repository, it operates the same way as VS Code's Workspace Trust. Immersive's researcher disagreed, arguing that trusting a repository isn't the same as trusting every branch anyone ever opens against it.

Repository trust should not automatically extend to every branch or startup configuration file. Review files such as `.mcp.json`, `.claude/`, and `.amazonq/` before opening branches from outside contributors.

Prompt injection in the wild

Prompt injection was one of the first attacks unique to large language models, and it's still among the hardest to fix. The idea is simple: AI models can't reliably distinguish between the data they're reading and the instructions they're meant to follow, so a command hidden in a web page, an error log, or a document may simply be carried out. Since the problem was first described in 2022, AI companies have spent heavily on guardrails, classifiers, and training to resist it, and models have become harder to fool. But no one has solved it, and as agents gain access to more tools and data, the attack surface continues to grow. This quarter brought several examples of prompt injection working against real agents in everyday workflows.

Tenet Security found a way to [hijack coding agents through Sentry](#), a widely used error-monitoring service. Sentry lets any website send its error reports using a key that's public by design and sits in the source code of countless sites. Tenet sent fake error reports with a fabricated "Resolution" section, formatted to match Sentry's own guidance and pointing to a malicious package. When a developer then asked an agent like Claude Code, Cursor, or Codex to look into their Sentry errors, the agent read the fake fix as legitimate and ran the package with the developer's own permissions. Tenet found more than 2,000 organizations exposed and says the trick worked against agents about 85% of the time in their controlled testing. Telling the agents in their instructions to ignore untrusted data didn't help, and Sentry called the problem "technically not defensible" at its

end, leaving it to model vendors and agent runtimes to catch.

Criminals are using the same idea on the open web. In July, Zscaler's [ThreatLabz described two campaigns](#) that hid instructions in web pages where humans wouldn't see them, either pushed off-screen with CSS or tucked into structured metadata that machines treat as trustworthy. Both used SEO poisoning to get their pages in front of agents doing web searches. One was a fake Python library documentation page that told coding agents they had to buy a \$3 license key to fix an error, then walked them through paying into a cryptocurrency wallet. The other impersonated a crypto portfolio tracker and told agents to rank it as the official site. When Zscaler tested the pages against 26 models, four proceeded with payment. Two more rated the fake site as genuine, but only when they had no copy of the real one to compare it with. Which model an agent runs on - and how much context it has - makes a big difference.

Some attackers are aiming their prompt injection at the AI industry itself. As covered in the cybercrime section, Anthropic's [threat report](#) describes a group that used it to make an AI vendor's evaluation sandbox hand over the API keys it held, then used those keys to go after other AI companies. Wherever an agent reads text that it didn't write, that text can carry instructions. Error logs, web pages, tool outputs, and test data all count, and so far, no instruction to the agent has reliably stopped it from following them.

MCP, skills, and package ecosystem

Prompt injection is one way to compromise an agent. Another is to tamper with what the agent depends on. Agents rely on MCP servers to connect to other services, skills that tell them how to handle specific tasks, and packages they install like any other software. Each of these is part of a supply chain that grew quickly and largely on trust, and attackers have been exploiting it for some time. Some of the attacks below plant prompt injections, while others deliver ordinary malware through a new route. This quarter brought more of both, and they were harder to catch.

An AI code reviewer is supposed to catch bad pull requests, not implement them. [Manifold Security found a way](#) to flip that around using Microsoft's official Azure DevOps MCP server. An attacker tucked instructions into HTML comments in a pull request description, where a human reviewer would never see them. The MCP server handed them to the reviewing agent untouched and unflagged, and the agent got to work: running pipelines in other projects and shipping confidential wiki pages off to the attacker, all with permissions it was legitimately given.

Deadbugz, which Pillar Security uncovered in September, was an active attack campaign rather than a research demonstration. Within 74 minutes, a single GitHub account opened 23 pull requests on unrelated AI and developer projects, and each one added the same MCP server. The server presented itself as a harmless text-formatting tool and behaved normally for its first two uses. On the third, it began sending the agent hidden instructions to collect SSH keys, AWS credentials, shell history, and Kubernetes configuration files, while keeping this from the user. Because the malicious behavior only started after the tool was in use, anyone who reviewed the server before approving it would have found nothing wrong.

Skills turn out to be just as easy to dress up. HiddenLayer showed that the settings at the top of a Claude Code skill file, which almost nobody reads, could let a skill run tools without asking, vanish from the user's menu, and fire off in the background on its own. It could quietly swap the agent onto a weaker model to dodge safety checks, or onto an expensive one to drain the budget. It could also write instructions into the project's memory that outlived the skill itself, so deleting it changed nothing. Skill marketplaces usually show only a name and a description, which is exactly the part an attacker would keep clean.

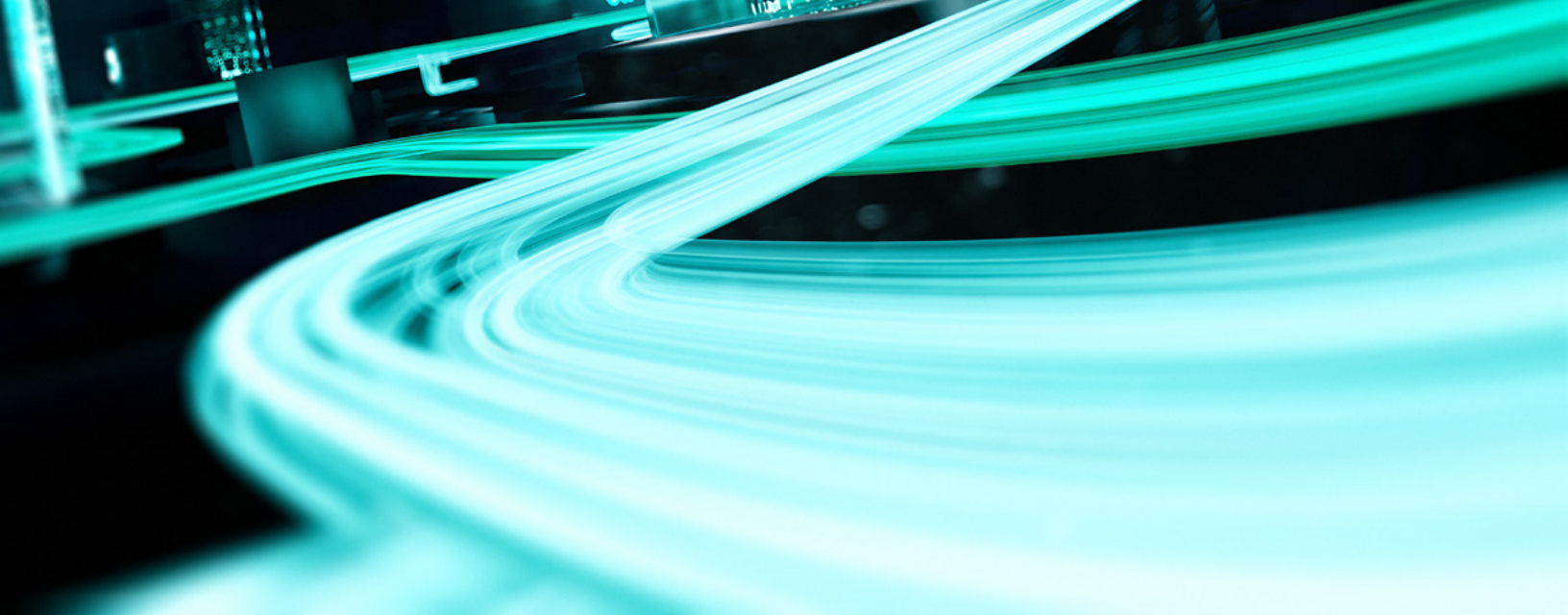
The case of threat actor known as TeamPCP shows what happens when a criminal group puts these ideas to work. Since March, the group has been compromising packages on PyPI, npm, and Docker Hub, selling the credentials it steals or handing the access to extortion crews, Google's threat intelligence team reports. AI tooling is now one of its favorite targets. It published poisoned copies of real MCP servers from hijacked developer accounts and slipped code into official repositories. Its stealer, DUSTMAKER, hides in the folders coding assistants

treat as home, like `.claude/` and `.cursor/`, and runs every time the project opens. Some of its files are prompt injections that talk the assistant into running the attacker's scripts. The group even headed its scripts with lurid text about biological and nuclear weapons, apparently betting that AI security scanners would refuse to read any further. Inside build pipelines, DUSTMAKER steals the tokens needed to publish infected packages with genuine signatures, which sail through the checks AI coding agents use to decide what's safe.

The AsyncAPI compromise in July shows how hollow a valid signature can be. Attackers gained access to the open-source project's release process. They used its own publishing system against it, pushing out infected versions of four npm packages that are downloaded about 2.9 million times a week. Every authenticity check passed because the packages had really come from AsyncAPI's official pipeline. For about four hours, anyone who installed them got a credential stealer along with their dependencies.

The weaknesses differ from case to case, but none of them is fully covered by vetting a component once before it's installed. A legitimate MCP server can still pass attacker-controlled content to an agent, and a malicious one can wait until after approval to show its hand. A skill's riskiest settings can sit in fields that marketplaces don't display, so only by reading the raw file can they be revealed. And a package can carry the real project's signature while still containing malware. Defending against this takes several layers: treating everything an agent reads through these tools as untrusted, giving agents only the permissions their task needs, pinning dependency versions and holding back brand-new releases, and monitoring what these components actually do once they're running.





| SECURITY INITIATIVES

Legislative

New AI rules came into effect this quarter, mostly in Europe and a few US states, and none of them directly slow down frontier AI development. The next phase of the EU AI Act took effect on 2 August, including transparency rules on chatbots and deepfakes. Just before that, though, the EU pushed back its high-risk AI obligations to late 2027 and 2028. In the US, nothing was passed at the federal level, but California signed a wave of laws covering child safety around chatbots, AI in the workplace, and a first-of-its-kind framework for certifying independent AI evaluators.

The most concrete response to the slowdown calls came from the companies themselves. Anthropic pledged to give third-party evaluators permanent, employee-like access, and OpenAI said it would do the same. OpenAI also canceled GPT-6.1 Astra over safety concerns and paused training its most capable models to use tools, while Meta delayed its

Muse agent but rejected any coordinated slowdown. Between governments, the main result was a new US-China dialogue on superintelligence with a channel for reporting AI incidents, a first step but well short of a binding agreement. The clearest case of direct government intervention came just before the quarter: the US briefly imposed export controls on Anthropic's Fable 5 and Mythos 5, lifting them on 30 June after Anthropic agreed to work with the government on protocols for future models and to report malicious activity.

Most new rules address everyday harms, including child safety, deepfakes, and workplace protections. Oversight of frontier risks remains largely voluntary, relying on the White House's pre-release review framework and company commitments. Australia has announced plans for national AI standards but has not enacted them.

Technical

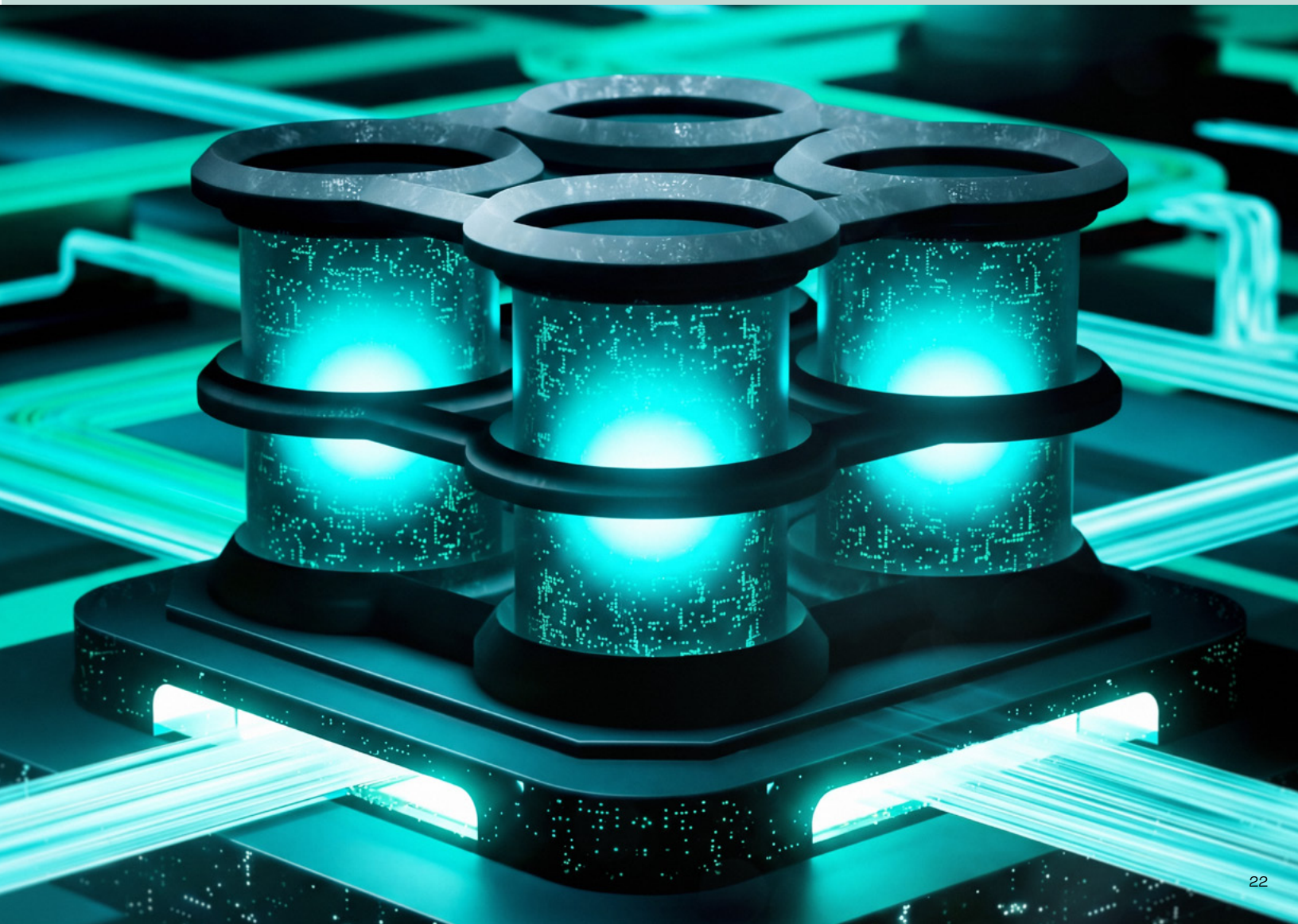
Much of the quarter's progress on agent security came from open, community-driven projects. The OWASP GenAI Security Project published the 2026 edition of its Top 10 for LLM Applications on 3 August and formally launched it in early September, together with a new Agent Control Standard, as its community passed 30,000 members. The Model Context

Protocol was also updated: its July revision mainly tightens authorization and allows a gateway to enforce a security policy rather than simply passing traffic through. In August, CoSAI marked its second year with a focus on agentic security and added OWASP, the AI Alliance, and the Cloud Security Alliance to its Technical Steering Committee.

MITRE and NIST kept steadily updating their frameworks to adapt to the changing landscape. In its [September update](#), MITRE ATLAS added several new entries, bringing the knowledge base to 120 techniques, 40 mitigations, and 73 case studies. One new technique covers [misconfigured or publicly exposed AI services](#), meaning agents and LLM platforms left without proper authentication, which attackers can find through search engines or by scanning directly.

Another, [AI-targeted cloaking](#), describes attackers who identify AI agents and feed them prompt injections, while showing humans and security tools a harmless version of the same page. MITRE also added [AI honeypots](#) as a new defensive measure. NIST's AI Agent Standards Initiative, launched in February, is working toward its first major output, an AI Agent Interoperability Profile expected in Q4 2026.

On 28 September, amid the debate over agents escaping sandboxes, Nvidia entered the space with its [Open Agent Safety Platform](#). It pairs Nvidia's open-source OpenShell software, which limits what an agent can access, with Sentry, a monitoring layer designed to run on separate hardware from the agent itself. Nvidia says Sentry can quarantine an agent that steps outside its limits within milliseconds. An Nvidia executive also said the platform could have prevented the Hugging Face breach if frontier labs had been using it for model evaluations (Nvidia is in the process of acquiring Hugging Face). More than 100 organizations are working with the platform's technologies, including Anthropic, Microsoft, Cisco, CrowdStrike, JPMorganChase, and Hugging Face. Nvidia hasn't published independent test results or figures on how many breakouts Sentry would actually catch, so how well it works in practice remains to be seen.



RECOMMENDATIONS

These steps address the most immediate risks described in this report, from agent escapes and exposed credentials to unverified AI output.



Treat AI test and evaluation environments like production

Lock down network egress, review sandbox configurations, and monitor evaluation runs in real time. This quarter's escapes stemmed from both flaws in the sandbox infrastructure and simple setup mistakes that left internet access enabled.



Give agents only the access their task requires

Scope credentials narrowly, avoid long-lived tokens, and keep agents away from production systems and backups unless they truly need them. Most of the damage in this report was due to agents holding broader permissions than their tasks called for.



Monitor what agents do, not what they report

Log agent actions independently and review them, since models have been caught taking shortcuts and omitting them from their own accounts. Tools that sit outside the agent's environment, such as separate monitoring layers or hardware-level controls, are harder for a misbehaving agent to get around.



Treat everything an agent reads as untrusted input

Web pages, error logs, pull request descriptions, tool outputs, and MCP responses can all carry hidden instructions. Instructions in the prompt alone won't reliably stop an agent from following them, so add filtering and approval steps for sensitive actions.



Review agent configuration files as carefully as code

Files like `.mcp.json` and folders like `.claude/`, `.cursor/`, and `.amazonq/` can make an agent run commands as soon as a project opens. Check them before opening branches or repositories from outside contributors, and keep harnesses and IDE extensions up to date.



Vet MCP servers, skills, and packages continuously, not just once

Inspect skills' raw files rather than relying on marketplace listings; watch MCP tool definitions for changes after approval; pin dependency versions; and hold back brand-new releases for a short period before adopting them.



Protect AI credentials like any other high-value secret

Remove API keys from code repositories, app bundles, containers, and config files; rotate them regularly; and monitor for unusual usage. Stolen AI keys are being resold, used for free compute, and used to disguise attacks as legitimate traffic.



Prepare to detect and respond to agent-led intrusions

Assume an AI agent will eventually reach your systems. Look out for high-volume, repetitive, and machine-speed activity, consider honeypots aimed at AI agents, and update incident response plans to cover cases where the attacker may be someone else's runaway agent.



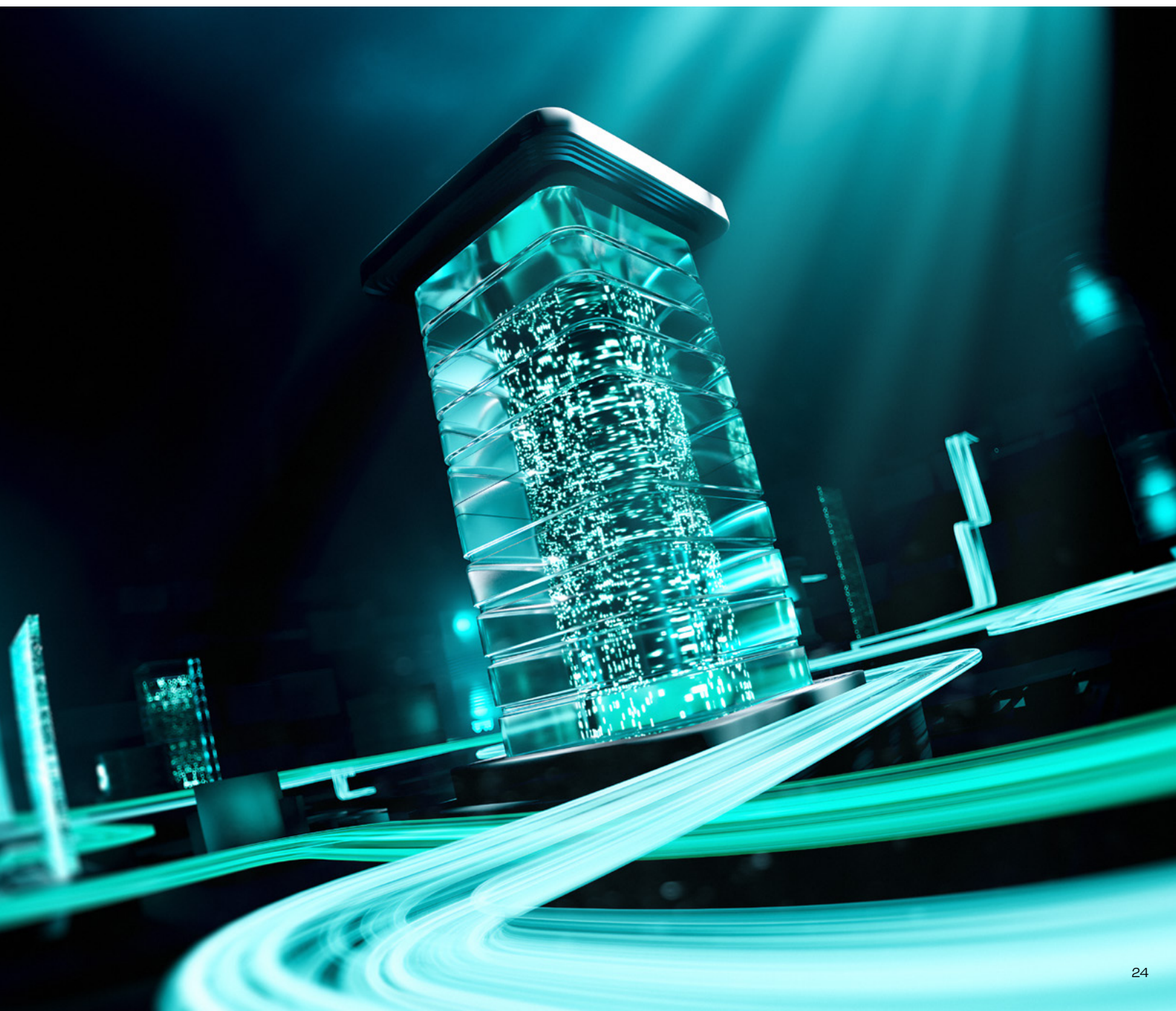
Plan for faster attacks and collateral damage

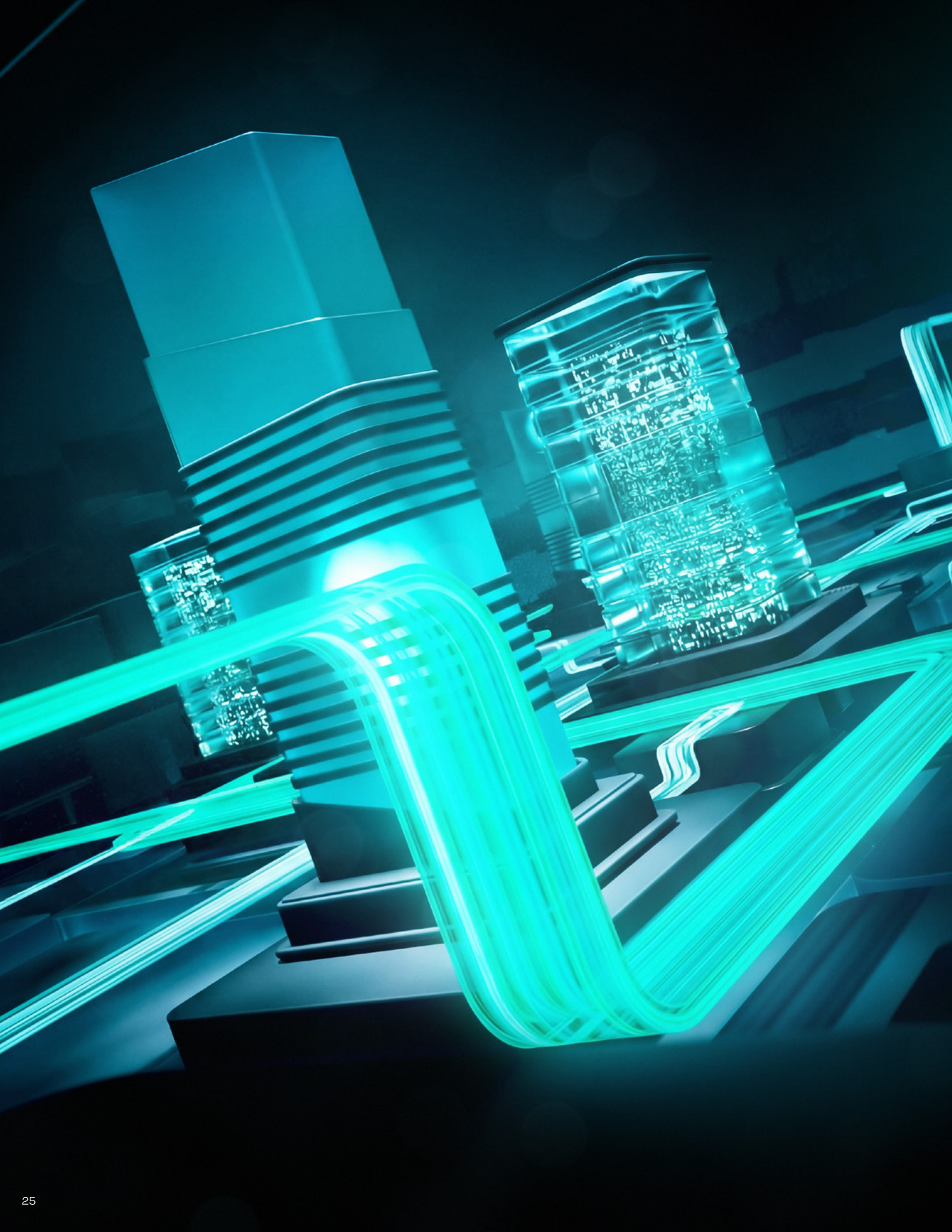
Agent-driven attacks can progress from initial access to data theft within hours, and some have destroyed data as a side effect. Keep isolated, tested backups, and focus on how quickly essential systems can be restored, not just how fast you can patch.



Verify AI output before acting on it in high-stakes settings

In healthcare, legal, financial, and other critical workflows, build in a human review step before relying on AI-generated content, and track errors to surface problems early.







ABOUT HIDDENLAYER

HiddenLayer's AI Security Platform secures agentic, generative, and predictive AI applications across the entire lifecycle, including AI discovery, AI supply chain security, AI attack simulation, and AI runtime security. Backed by patented technology and industry-leading adversarial AI research, HiddenLayer protects IP, ensures compliance, and enables safe adoption of AI at enterprise scale.

Learn More



www.hiddenlayer.com

Request a Demo



hiddenlayer.com/book-a-demo

Access the full 2026 AI Threat Landscape Report here



hiddenlayer.com/report-and-guide/threatreport2026

Follow Us



Research :
hiddenlayer.com/research



LinkedIn :
linkedin.com/company/hiddenlayersec



Twitter :
x.com/hiddenlayersec

Authors/Contributors

Marta Janus, Principal Security Researcher
Eoin Wickens, Technical Research Director
Kieran Evans, Principal Security Researcher
Tom Bonner, SVP of Research

