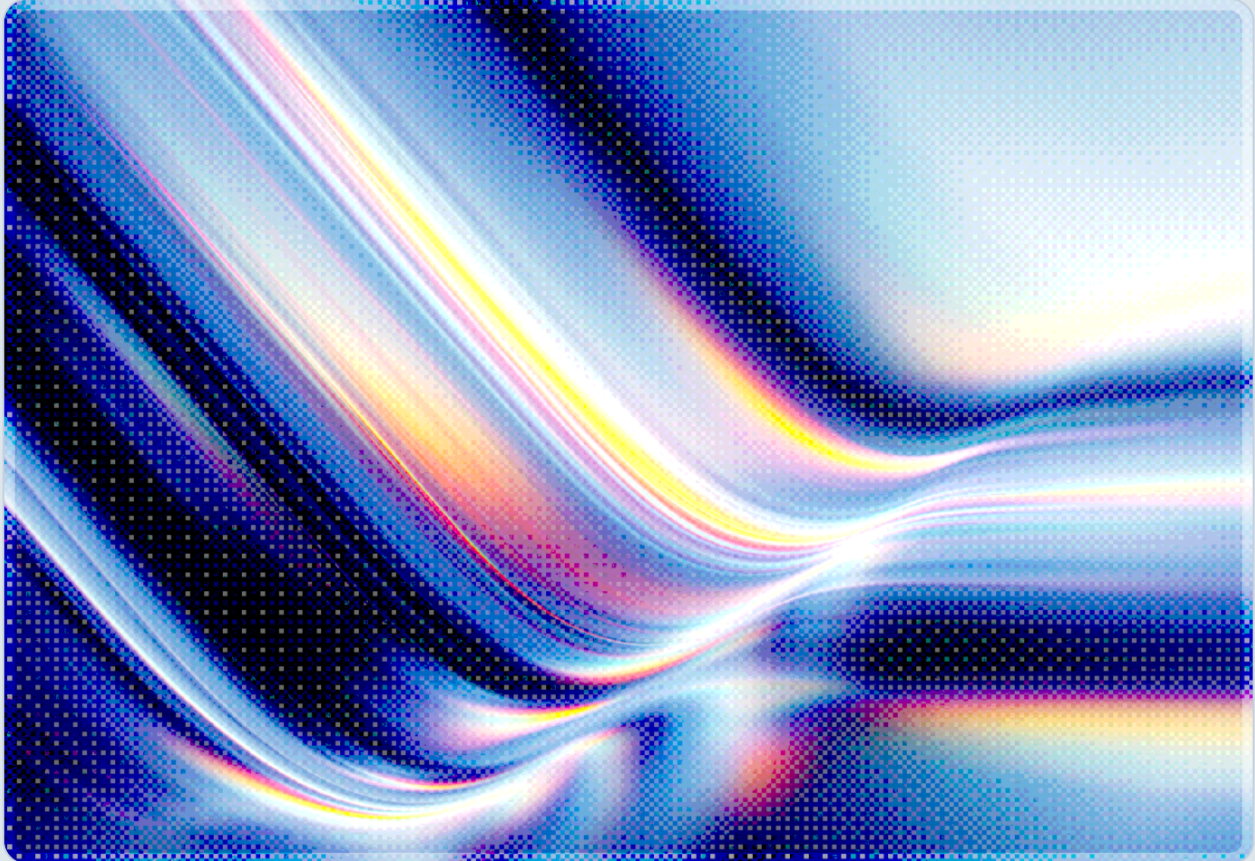


15,465 MCP Servers and No Consistent Governance Signal: Marketplace Supply Chain Risk

Security Implications and Guidance for Public MCP Registries,
Examining OX Security's 'O Governance' Finding

2026-10-07

 AI-assisted Rapid Research



CSAI

cloud
security
alliance®

© 2026 Cloud Security Alliance. Some rights reserved.

You may download, store, display, view, print, redistribute, and link to this document in its original, unmodified form, provided that attribution to the Cloud Security Alliance is maintained and all trademark and copyright notices remain intact.

This document may not be modified or altered. You may quote portions of the document as permitted by the Fair Use provisions of the United States Copyright Act, provided that attribution is given to the Cloud Security Alliance.

This document may be shared on professional and social media platforms in its original form with attribution.

This document was generated with AI assistance and has not undergone official CSA review and approval processes.

Key Takeaways

- OX Security's September 2026 study of 15,465 servers listed in three public Model Context Protocol (MCP) registries concluded that the ecosystem lacks consistent authorization models, operator identity, and lifecycle accountability. Its "0 Governance" title is the vendor's framing, not a measured count [1][2][3].
- The infrastructure percentages refer to the 5,095 unique hostnames behind those listings, not to all 15,465 servers: 15.6% resolved to infrastructure outside the United States, 2.3% no longer resolved, and 0.45% sat on home networks or consumer tunneling services. Six expired domains could be bought for \$4 to \$12 per year. These are vendor-reported results that we could not audit against the underlying dataset [1][3].
- Independent research reports that registries contain unverified lookalikes of verified brand servers, and ClawHub's experience shows that a skill marketplace for agents can host hundreds of malicious entries [4][5][6].
- A registry listing is a discovery convenience, not an assurance signal. Enterprises should treat each MCP server connection as a third-party code and data dependency and govern it through an allowlist, not through developer discretion.

Background

The Model Context Protocol, introduced by Anthropic in November 2024, gives AI agents and applications a standard way to connect to external tools and data sources [10]. Its adoption has produced a set of public registries and marketplaces where publishers list servers and users, or the agents acting for them, find and configure them. The protocol was designed to make tool discovery and invocation easy. The specification defines an optional OAuth-based authorization framework for HTTP transports [11], but whether a given server enforces it is left to that server's implementation, and vetting of publishers falls largely to the registries themselves [3].

OX Security Research published "15,465 MCP Servers. 0 Governance." in September 2026. The study examined servers listed across three public registries: the official MCP registry, the Cline marketplace, and the GitHub MCP registry [1][2]. The researchers narrowed the listings to 5,095 unique hostnames for infrastructure analysis, then checked where those hosts resolved and whether the domains were still

live [1][3]. The sources do not explain what happened to the remaining listings, which may be local or package-based servers with no hostname or may share hosts, so readers should not apply the hostname percentages to the full 15,465. The report is vendor research. We have not been able to review its full dataset or methodology, so the figures below are the authors' reported results and should be read as indicative of ecosystem conditions rather than as an audited census.

The "O Governance" in the title is a characterization, not a measured count. Coverage of the report describes it as finding that none of the listings demonstrated a consistent, verifiable authorization control, and that a registry entry alone does not reveal whether a server enforces access control at all [3]. That is a narrower and better-supported claim than a statement that no server in the sample had any protection. Some individual servers likely do, and the official registry performs namespace-ownership checks for publishers, so we have retitled this note to reflect what the evidence supports.

Security Analysis

What the infrastructure data shows

The hostname analysis supports three distinct risks. The first is data residency. Of the 5,095 hostnames, 15.6% (796) resolved outside the United States, including 19 in China and 18 in Russia, which together account for about 4.6% of the foreign-hosted set [3]. The sources we reviewed do not provide the full country distribution. OX notes that MCP has no protocol-level concept of geographic region, so an enterprise can enforce residency controls on its own cloud workloads while its agents connect to servers that sit outside those controls [2]. Foreign hosting is not malicious in itself, and the relevant question for any enterprise is jurisdiction relative to its own obligations rather than relative to the United States. The concern is that the agent's tool traffic, which may carry prompts, file contents, and credentials, crosses jurisdictions that no one at the enterprise decided to approve.

The second risk is dangling infrastructure. About 2.3% of hostnames no longer resolved, and six of the unresolved domains were unregistered and available for roughly \$4 to \$12 per year [1][3]. An agent or a user configuration that still points at such a name could be redirected to an attacker who registers it. This resembles the abandoned-dependency takeover pattern seen in package ecosystems, applied to a tool endpoint that the agent trusts to return instructions as well as data. The sources we reviewed describe this as a possibility and do not report an exploited case.

The third risk is enterprise network exposure. Roughly 0.45% of hostnames were associated with home networks or consumer tunneling tools [1]. The figure is small, but such servers are likely to sit outside organizational control, with limited assurance of uptime, patching, or logging, and a developer's laptop or

home router may be the operator of record for a tool that an enterprise agent calls. The table below consolidates the reported values; note that every percentage is a share of hostnames.

Finding	Reported value	Source
Servers listed across three registries	15,465	[1]
Unique hostnames analyzed	5,095	[1][3]
Hostnames resolving outside the US	15.6% (796), including 19 in China and 18 in Russia	[1][3]
Hostnames no longer resolving	2.3%	[1][3]
Hostnames on home or consumer tunnel networks	0.45%	[1]
Unregistered domains available for takeover	6, at \$4–\$12 per year	[1][3]

Listings carry no attestation

A structural issue sits beneath these infrastructure findings. In the listings OX reviewed, nothing in a registry entry binds a published source repository to the code running behind a live endpoint. OX describes the absence of operator identity and of any attestation that the running server matches its published source [1]. Discovery in MCP is typically unauthenticated, so a registry cannot tell a consumer whether the server validates callers before executing tools [3]. A consumer who reads a clean repository has therefore verified a repository, not the service the agent will call. Remote servers compound this problem, and UpGuard found that some depend on an individual's GitHub account rather than a verified organization [6].

Impersonation and typosquatting

Registry size lowers the barrier to impersonation. UpGuard analyzed 18,000 Claude Code settings files from public GitHub repositories and found misspelled server names in real configurations. It also reported between 3 and 15 unverified lookalike servers for each official brand server, and estimated that 10–16% of servers across the registries it studied were lookalikes of verified brands [6]. UpGuard's figures are also vendor research, and a lookalike is not necessarily malicious; the sources we reviewed do not

report a confirmed malicious typosquat. UpGuard observed that MCP.so hosted more than 17,000 servers, while GitHub's curated registry listed 57 and was described as highly moderated [6]. That comparison is consistent with an inverse relationship between registry openness and the confidence a listing deserves, though it rests on a single contrast rather than a broad data set.

Precedent from agent skill marketplaces

ClawHub, the skill marketplace for the OpenClaw agent, shows how this plays out once attackers pay attention. Koi Security audited all 2,857 skills then listed and identified 341 malicious entries (about 11.9%), 335 of them tied to a single campaign tracked as ClawHavoc that delivered the Atomic Stealer macOS infostealer through fake prerequisites [4][5]. Later Koi reporting, which we could not confirm on the page cited as [4], put the malicious count at 824 as the marketplace grew past 10,700 skills. The count rose, but the share fell to about 7.7%. CSA's own synthesis of subsequent disclosures found that attackers had begun engineering skills that pass the marketplace's scanning pipeline [7].

MCP registries and skill marketplaces differ in mechanics, since an MCP server is typically a running service while a skill is installed content. They share the property that an autonomous agent executes third-party logic, often with broad privileges depending on how it is configured. This note cites no malicious-server campaign in an MCP registry comparable to ClawHavoc, so the precedent is an analogy rather than evidence. It is nonetheless reasonable to expect similar adversary interest in MCP registries as they grow.

Persistent approvals widen exposure

OX tested a malicious server against Claude Code paired with an older model, which [2] identifies as Haiku 3.5. The server first requested a harmless file, the user granted an "always-allow" permission, and the server then requested sensitive files, including `.env`, without a further prompt [1][2]. According to OX, the attack did not succeed against newer Opus-class models in its tests [1]. This is one scenario from one vendor test, and "always-allow" is a deliberate user choice. It still illustrates two points. Persistent permission grants widen the blast radius of any later compromise of the server. Model-side resistance to injection varies by version, so it should be treated as a mitigating factor that may change rather than a boundary on which policy can rest.

Tool descriptions are a second injection channel that registry vetting does not address. CSA's ShareLock analysis describes an attack that splits an adversarial instruction across the descriptions of several MCP tools, keeping each fragment below per-tool detection thresholds. The underlying research reports high

attack success against the models it tested and no detection by the five detectors evaluated [8]. A server that looks benign in isolation can therefore contribute to a malicious outcome when combined with others, which weakens review models that evaluate servers one at a time.

Recommendations

Immediate Actions

Organizations should inventory every MCP server that agents, IDE assistants, and developer tooling connect to, including remote endpoints configured in local settings files. Many organizations lack a complete list, and OX's central recommendation is visibility into connected agent infrastructure [1]. In parallel, teams should disable or review persistent "always-allow" permissions for MCP tools, and remove servers whose hostnames no longer resolve or whose publisher cannot be identified. Existing configurations should also be searched for near-miss spellings of vendor server names.

Short-Term Mitigations

Open registry discovery should give way to an internal allowlist or private registry that records, for each approved server, an accountable owner, the pinned version or source commit, the hosting location, and the authentication method. Agent traffic to MCP servers should route through an egress gateway or proxy that enforces the allowlist, applies residency policy by destination, and logs tool calls. Remote servers should require OAuth or equivalent short-lived credentials instead of static keys, with tokens scoped to the minimum tools needed. For servers that must be evaluated, test the live endpoint in a sandbox rather than reviewing only the repository, and review tool descriptions in combination with the other servers an agent will load.

Strategic Considerations

The registries that OX and UpGuard examined did not consistently provide verified publisher identity, signed provenance linking a listing to a build, namespace protection against lookalikes, or lifecycle signals such as deprecation and domain-expiry monitoring. Some partial controls exist, such as the official registry's namespace verification, but they were not reported as a consistent baseline. Enterprises can use procurement and platform-selection leverage to ask registry operators and agent vendors for these capabilities. Residency and operator identity should become attributes of an MCP server record, much

as region and account are attributes of a cloud resource. Because the figures in this note come largely from vendor research, organizations with high-assurance requirements should replicate the hostname and lookalike analyses against the specific registries and servers they use.

CSA Resource Alignment

CSA's research note [ClawHub Under the Microscope: Agentic AI Supply Chain Risk](#) is the closest prior work. It concludes that ClawHub is a persistent, evolving threat surface that may generalize to any marketplace allowing agents to install and run third-party code, and it advises treating each skill installation as an unreviewed code execution event [7]. The OX findings extend that argument from skill marketplaces to MCP registries, where the unit of trust is a running remote service rather than installed content. The same governance posture applies: an approval workflow and an allowlist ahead of any installation or connection.

CSA's [ShareLock: Stealthy Multi-Tool Threshold Poisoning in MCP](#) addresses what a single-server review cannot see, namely adversarial content distributed across several servers' tool descriptions [8]. It supports the recommendation here to evaluate servers as a combined set. In our assessment, the absence of a cross-server isolation boundary in MCP is a design gap that registry vetting alone cannot close.

The [AI Controls Matrix \(AICM\) v1.1](#) provides the control vocabulary for the recommendations above. In our reading, its supply chain, application security, and identity and access management domains map to the server allowlist, provenance requirements, and credential practices described here [9]. Organizations may find it useful to record each approved MCP server as a third-party service dependency within their AICM assessment scope.

References

- [1] OX Security. "[15,465 MCP Servers. 0 Governance.](#)" OX Security Research, September 2026.
- [2] Infosecurity Magazine. "[MCP Is Creating Major Governance Gaps, Researchers Warn.](#)" Infosecurity Magazine, September 28, 2026.
- [3] WorkOS. "[What 15,465 ungoverned MCP servers tell us about the authorization gap.](#)" WorkOS Blog, September 2026.
- [4] Koi Security. "[ClawHavoc: 341 Malicious Clawedbot Skills Found by the Bot They Were Targeting.](#)" Koi Security, 2026.
- [5] The Hacker News. "[Researchers Find 341 Malicious ClawHub Skills.](#)" The Hacker News, February 2026.
- [6] UpGuard. "[Emerging Risks: Typosquatting in the MCP Ecosystem.](#)" UpGuard, July 2, 2026.
- [7] Cloud Security Alliance AI Safety Initiative. "[ClawHub Under the Microscope: Agentic AI Supply Chain Risk.](#)" CSA, July 7, 2026.
- [8] Cloud Security Alliance AI Safety Initiative. "[ShareLock: Stealthy Multi-Tool Threshold Poisoning in MCP.](#)" CSA, June 26, 2026.
- [9] Cloud Security Alliance. "[AI Controls Matrix \(AICM\) v1.1.](#)" CSA, 2026.
- [10] Anthropic. "[Introducing the Model Context Protocol.](#)" Anthropic, November 25, 2024.
- [11] Model Context Protocol. "[Authorization.](#)" Model Context Protocol Specification, 2025-06-18.