La vulnérabilité MCP comme cas pur du port de promotion

Comment un SDK conçu pour le prototype local est devenu infrastructure d'agents sans procédure de transfert de juridiction

I. Un fait précis, mal nommé : du cyber à l'agency-grade

Le 15 avril 2026, OX Security publie une advisory technique qui décrit une vulnérabilité « by design » dans le Model Context Protocol d'Anthropic, référencée CVE-2026-30623. La primitive est simple : le SDK officiel implémente le transport STDIO en exécutant la commande déclarée par la configuration avant de vérifier qu'elle lance bien un serveur MCP valide. Si la commande échoue à initier un serveur, elle s'est déjà exécutée. Cela suffit à transformer toute configuration mal contrôlée, locale ou exfiltrée, en primitive d'exécution arbitraire de code. Le défaut affecte le SDK officiel dans tous les langages supportés (Python, TypeScript, Java, Rust).

L'ampleur, telle que documentée par OX dans son advisory, mérite d'être lue précisément. Au moment de la publication, l'équipe identifie environ 7 000 serveurs MCP publiquement accessibles porteurs de la primitive. Sa projection vers l'ensemble des déploiements, plus exposée, dérive un ordre de grandeur d'environ 200 000 instances. Le SDK officiel cumule plus de 150 millions de téléchargements selon les registres standards. Quatorze CVE liés ont été assignés à la date de l'advisory, et plus de trente vulnérabilités d'exécution à distance documentées sur des produits intégrant MCP : LiteLLM, LangFlow, Windsurf, Cursor, Flowise, DocsGPT, GPT Researcher. La timeline est elle-même informative. OX Security contacte Anthropic le 7 janvier 2026. Neuf jours plus tard, l'éditeur met à jour son fichier SECURITY.md pour préciser que les adaptateurs STDIO doivent être utilisés avec précaution, sans modification architecturale. Cinq mois d'échanges plus tard, Anthropic confirme la position dans une formule textuelle qui mérite d'être citée intacte : *« STDIO execution model represents a secure default, sanitization is the developer's responsibility »*. La publication de l'advisory suit, le 15 avril.

La couverture cyber-sécurité de l'événement est correcte et abondante. The Hacker News, The Register, Tom's Hardware, Hackaday, des newsletters spécialisées : tous décrivent la primitive technique, énumèrent les produits affectés, recommandent des mitigations applicatives (gateways, audit trails, sanitization explicite, SSO-integrated auth). Cloudflare publie une *enterprise MCP reference architecture* dans la foulée. Cisco intègre l'épisode dans son *State of AI Security 2026*. Tout cela est utile, et rien de tout cela ne suffit.

La couverture cyber traite MCP comme un problème de durcissement d'infrastructure logicielle, comme on a traité log4j en 2021 ou OpenSSL Heartbleed en 2014. C'est insuffisant. ***MCP n'est pas une librairie passive : c'est un protocole qui définit la surface***

d'orchestration d'agents capables d'action contextuelle autonome sur des systèmes externes. Le saut n'est pas *enterprise-grade*, il est *agency-grade*. Cette distinction n'est pas anecdotique. Elle décrit le passage d'un système qui traite de l'information à un système qui modifie le monde opérationnel : commit dans des dépôts, exécution dans des CI/CD, écriture dans des bases, ouverture de tickets, déclenchement de workflows. Entre l'industrialisation logicielle classique et l'orchestration d'artefacts décisionnels distribués agissant ainsi, il y a une rupture qu'on peut, sans excès, qualifier de civilisationnelle. Un événement de cette nature appelle une analyse autre que cyber-classique. Reste à nommer ce que la couverture cyber rate.

II. Publication, intégration, promotion : trois opérations distinctes

La question utile n'est pas *« qui a fauté ? »*. Elle est *« quelle opération institutionnelle n'a pas eu lieu ? »*. Cette reformulation ouvre la seule porte par laquelle la réponse d'Anthropic, *« expected behavior »*, peut être lue comme rigoureuse plutôt que comme une démission.

Précisons un mot qui va revenir. Par *juridiction*, je n'entends pas ici un tribunal, ni un trust domain, ni un simple intended use. J'entends le périmètre institutionnel, opérationnel et décisionnel dans lequel un artefact est réputé fonctionner sans hypothèses supplémentaires explicitées. MCP livré pour le prototype local opère dans une juridiction où les hypothèses (configuration de confiance, sanitization manuelle, audit applicatif) sont celles du développeur seul. MCP déployé comme infrastructure d'agents en production opère dans une juridiction où ces mêmes hypothèses doivent être portées par l'écosystème. Ce sont deux juridictions distinctes au sens où je l'emploie ici.

Le déplacement de MCP entre ces deux juridictions s'est fait en trois temps successifs et bien identifiables.

> *Anthropic publie* : ***fin 2024, MCP est conçu comme protocole d'intégration locale pour Claude Desktop***. Sa juridiction d'origine est étroite et claire. Un développeur expérimenté connecte localement un modèle Claude à des outils qu'il contrôle lui-même, avec les responsabilités opérationnelles qui en découlent. Le SDK officiel est publié, le SECURITY.md précise les conditions d'usage, le code est ouvert. C'est conforme à ***la juridiction E***.
>
> *Les frameworks intègrent* : mi-2025, l'écosystème agentique amorce sa promotion. Les frameworks émergents (LangGraph, CrewAI, AutoGen) intègrent MCP comme couche d'outils par défaut. Les vendors qui livrent en production cliente (LiteLLM, LangFlow, Windsurf, Cursor) le déploient dans leurs offres. Chacun de ces intégrateurs opère dans sa propre juridiction applicative. C'est ***la juridiction I***.
>
> *L'écosystème promeut* : début 2026, la concordance de signaux entre Anthropic, OpenAI (Apps SDK et Connectors en avril 2025), Google (Gemini API et Vertex AI Agent Builder en mars 2026), Cloudflare (reference architecture en avril 2026), AAIF (MCP Dev Summit North America au printemps 2026), et la cohorte des frameworks

> récents transforme l'artefact en standard de fait pour le runtime agentique. À ce stade de promotion, on parle d'environ 9 400 serveurs publics dans les registres standards au deuxième trimestre 2026, en croissance de l'ordre de 58 % trimestre sur trimestre selon les enquêtes industrielles disponibles. Les enquêtes auprès des équipes IA enterprise placent l'adoption à plus de trois sur quatre, avec MCP cité comme standard agent par défaut chez environ deux tiers des CTO interrogés. Ces chiffres sont des signaux faibles de surveys industriels, pas des mesures auditées. Mais leur convergence indique une promotion réussie. C'est ***la juridiction S***.

Trois opérations, trois acteurs, trois juridictions. Cette trichotomie est exactement la structure que le protocole Twingital v3 désigne sous le nom *E/I/S* en propriété intellectuelle (exogène / interne / systémique).

Transposée du registre IP au registre épistémique-procédural, elle décrit ici la distribution de la responsabilité dans la promotion d'un artefact technique vers une juridiction opérationnelle plus exigeante. La doctrine n'a pas à être importée ; elle est ce que le cas MCP révèle quand on regarde de près qui fait quoi.

Quand OX Security contacte Anthropic le 7 janvier 2026 et reçoit *« expected behavior »* comme réponse, l'éditeur protège correctement la juridiction E.

Quand LiteLLM ou Cursor intègrent MCP, ils opèrent dans la juridiction I.

La juridiction S, celle où l'écosystème entier déploie MCP comme infrastructure agentique, n'a pas de protecteur explicite. Personne n'est en charge de la juridiction S parce qu'elle n'a pas été instituée.

La réponse *« expected behavior »* est rigoureuse à E. Elle est inopérante pour S, qui n'a jamais été promu institutionnellement, seulement par adoption convergente. La formule *« sanitization is the developer's responsibility »* est, lue dans ce cadre, la formulation textuelle exacte de quelque chose d'autre : l'absence d'opération institutionnelle de promotion vers la juridiction S.

Le concept de *port de promotion*, introduit dans le second volet du diptyque IA-énergie [*Allouer le kilowattheure-IA*, mai 2026], désignait jusqu'ici le glissement par lequel un test technique se transforme en clé d'admission institutionnelle sans avoir été calibré pour cette charge. L'artefact y était une métrique de benchmark. Ici, l'artefact est un SDK. Le mécanisme est plus large que les benchmarks. Le port de promotion s'élève en cas particulier d'un mécanisme général : la dilution de responsabilité dans les architectures composites, là où l'adoption convergente d'un artefact crée une juridiction systémique que personne n'a explicitement instituée.

Le SDK n'a pas changé d'usage. C'est sa juridiction qui a changé sans procédure de transfert.

III. La concession technique d'abord, puis trois mécanismes et un modèle économique

Une objection se présente, et elle est légitime. Pour cette vulnérabilité précise, un patch côté SDK suffit largement à réduire le risque immédiat : configuration safe par défaut, sandbox du transport STDIO, rejet explicite des commandes qui n'initient pas un serveur valide. OX Security le recommande. L'industrie peut le déployer. Anthropic peut, à terme, l'intégrer en backward-compatible. Reconnaissons cela sans détour : la vulnérabilité CVE-2026-30623, prise isolément, est techniquement résoluble en quelques lignes de code.

Cet article n'est pas écrit pour cette vulnérabilité-là.

Il est écrit pour comprendre pourquoi une procédure de promotion institutionnelle de MCP vers la juridiction agentique enterprise n'a jamais existé, et ne pourra pas exister sans un déplacement doctrinal explicite. ***L'industrie n'a pas oublié d'instituer la promotion : Elle découvre simplement qu'elle peut croître plus vite tant qu'elle ne le fait pas.*** Trois mécanismes causaux verrouillent cette découverte. Pris ensemble, ils forment un système fermé : on ne peut pas en corriger un en laissant les deux autres en place.

1. Premier mécanisme, *asymétrie de bénéfice et compression temporelle stratégique*. La rapidité d'adoption profite aux acteurs qui adoptent vite. Mais ce n'est pas seulement une asymétrie passive entre vainqueurs et perdants d'un cycle technologique. C'est une stratégie active. L'éditeur capte l'effet de standard avant stabilisation ; l'intégrateur capte la position de marché avant que la barrière d'entrée durcie ne s'installe ; le déployeur capte l'avantage applicatif avant ses concurrents. L'absence de procédure de promotion produit un avantage compétitif immédiat : accélération de diffusion, capture de standard, externalisation du coût de durcissement vers les intégrateurs et les déployeurs. La procédure n'est pas seulement manquante par lacune historique ; elle est manquante parce que son absence est productive. La formule d'Anthropic, *« sanitization is the developer's responsibility »*, distribue formellement la charge vers l'aval. Elle est exactement la formulation textuelle de l'externalisation que produit la stratégie. Ironie tenue à distance respectueuse.
2. Deuxième mécanisme, *absence de juridiction unique*. Anthropic publie. Les frameworks intègrent. Les entreprises déploient. La trichotomie observée en section II trouve ici sa cause structurelle : trois acteurs, aucun avec mandat ni autorité pour conduire la promotion procédurée vers la juridiction systémique. Cette absence n'est pas une défaillance bureaucratique à laquelle on pourrait remédier en désignant un responsable. Elle est une propriété structurelle des architectures composites : aucune des trois juridictions ne peut absorber les deux autres sans dénaturer ses propres responsabilités. L'éditeur ne peut pas porter la charge de l'intégration applicative sans devenir intégrateur. Les intégrateurs ne

peuvent pas porter la charge de la promotion systémique sans devenir consortium. L'écosystème ne peut pas porter la charge de la publication sans devenir éditeur.

3. Troisième mécanisme, *absence de métrique commune*. Aucun benchmark partagé ne permet de qualifier qu'un SDK est *prêt pour la juridiction agentique enterprise*. Aucun test public ne distingue un artefact bien conçu pour le prototype local d'un artefact bien conçu pour l'orchestration de production. La critique RAISE du benchmark sans empreinte, déjà mobilisée pour les évaluations de modèles, trouve son extension naturelle ici : il n'y a pas davantage de benchmark pour le durcissement procédural d'un SDK promu infrastructure que pour le coût ressource d'un modèle. Le déficit métrologique de la couche modèle se reproduit, sous une autre forme, à la couche agent.

À ce stade, il est utile de hiérarchiser ce qu'on sait. Les faits durs sont la date de l'advisory (15 avril 2026), le CVE (2026-30623), les produits affectés (LiteLLM, LangFlow, Windsurf, Cursor, Flowise, DocsGPT, GPT Researcher), et la réponse textuelle d'Anthropic. Sur ces points, le débat est documenté et stable. Les estimations industrielles, secondaires mais sérieuses, sont les ordres de grandeur que produisent OX et les registres : environ 7 000 serveurs publics confirmés, environ 200 000 estimés, environ 150 millions de téléchargements du SDK officiel.

Ces chiffres sont des projections fondées et utiles, mais ils ne valent pas comme mesures auditées. Les signaux faibles, mobilisés pour caractériser la dynamique d'adoption, sont les enquêtes auprès des équipes IA enterprise et des frameworks : adoption au-delà des trois quarts, MCP comme standard par défaut chez la majorité des CTO interrogés, intégration built-in dans la quasi-totalité des frameworks agent récents. Ces signaux indiquent une trajectoire, pas un état mesuré. La thèse du présent article ne dépend d'aucun de ces chiffres pris isolément, mais de leur convergence : si la projection des 200 000 serveurs s'avérait exagérée d'un facteur deux et si l'adoption enterprise s'avérait surestimée de quinze points, la conclusion resterait inchangée. C'est précisément l'intérêt d'une thèse qui porte sur le mécanisme, pas sur l'amplitude.

L'objection économique se formule alors clairement, et elle est forte. Toute cette analyse de la promotion comme modèle économique est correcte, mais elle s'applique aussi à Internet, à OpenSSL, à Kubernetes. ***La dilution de responsabilité dans les architectures composites est une caractéristique permanente du logiciel open*** ; sa permanence n'a pas empêché la civilisation de tenir. Nommer ne change rien.

Concession à part : oui, la dilution est permanente dans le logiciel open. Mais sa permanence n'a pas davantage supprimé son coût :

- Heartbleed a coûté à OpenSSL plusieurs années de gouvernance reconstruite,

- log4j a coûté à la Apache Software Foundation un cycle complet de durcissement procédural et un changement de doctrine sur les dépendances systémiques invisibles ,
- Kubernetes a fini par développer une procédure de certification CNCF qui n'existait pas dans ses premières années.

Le coût se paie. La seule question est s'il se paie *ex post*, après que la dette systémique se soit accumulée, ou *ex ante*, par une procédure de promotion explicite. Pour MCP, et pour les couches agent qui le suivront, la fenêtre *ex ante* reste ouverte. Pour combien de temps, c'est le seul vrai sujet.

log4j logge. OpenSSL chiffre. Kubernetes orchestre des conteneurs. MCP orchestre des agents agissant contextuellement sur des systèmes externes. Le glissement vers l'agency-grade prend ici sa portée : ce qu'un compromis Heartbleed permet d'exfiltrer reste de l'information ; ce qu'un compromis MCP permet d'orchestrer reste de l'action. La distinction n'est pas accessoire : Elle change le coût attendu de la dette.

Conséquence systémique : tant que la promotion institutionnelle reste implicite, la même dette se reproduira à chaque couche montée vers le runtime agentique. Aujourd'hui MCP. Demain les protocoles de mémoire d'agent long-horizon. Après-demain les frameworks de planification multi-step. Ensuite l'orchestration de sous-agents. La couche agent est mature avant que sa procédure de promotion ne le soit, et son absence est, pour l'instant, économiquement productive ou du moins perçue comme telle en l’absence avérée d’un « accident » industriel.

Le port de promotion n'est pas un oubli. C'est une stratégie.

IV. Quatre conditions d'une promotion explicitante

À quoi ressemblerait une procédure de promotion institutionnelle pour un SDK agent, sans tomber dans la sur-certification ni dans la pseudo-conformité que la profession sait produire en quantité ?

La question est cardinale parce qu'elle est immédiatement récupérable.

Disons-le d'entrée : le but n'est pas de produire un certificat. Le but est d'empêcher qu'un changement de régime d'usage soit traité comme une simple adoption technique. La différence est fondamentale :

- Une certification dit *« ce SDK est sûr »,*
- Une procédure de promotion dit *« ce SDK est conçu pour la juridiction X, et son déploiement dans la juridiction Y suppose les durcissements Z »*.

La première est récupérable par les consultants Governance/Risk/Compliance. La seconde reste opérationnelle.

Disons-le encore plus clairement avant de détailler les conditions : la procédure de promotion n'est pas une couche de conformité. C'est un mécanisme de limitation explicite des hypothèses transférées. Le sujet n'est pas la gouvernance administrative ; il est l'architecture des hypothèses opérationnelles qu'un artefact emporte avec lui quand il quitte sa juridiction d'origine. Tout le reste relève de la rédaction de checklists par des cabinets spécialisés dans la production de checklists.

Trois précédents partiels éclairent la praticabilité, mobilisés comme repères et non comme modèles transposables :

1. FIPS 140-3 pour les modules cryptographiques organise quatre niveaux explicitement publiés, pas une certification monolithique ; la déclaration précise du niveau et du périmètre fait carrière comme objet juridique.
2. Common Criteria pour les composants logiciels critiques fonctionne sur la base de *protection profiles* déclarés, qui spécifient le périmètre d'évaluation.
3. Les Quality Management System Regulations de la FDA et leur alignement ISO 13485 pour les dispositifs médicaux logiciels reposent sur la déclaration explicite de l'*intended use*, qui détermine le périmètre régulatoire applicable.

Aucun de ces trois dispositifs n'est transposable tel quel à MCP. Tous indiquent que l'explicitation du périmètre de validité est une opération institutionnelle praticable et opérée ailleurs. Pas un horizon utopique.

Pour MCP et la couche agent dans son ensemble, quatre conditions paraissent nécessaires.

1. **Condition 1, déclaration explicite de juridiction d'origine et de juridiction promue.** Tout SDK ou protocole agent doit déclarer publiquement le périmètre d'usage pour lequel il a été conçu, et le périmètre auquel il prétend désormais s'appliquer. Si Anthropic avait publié, au moment de la promotion de MCP comme standard agent enterprise mi-2025, un document précisant explicitement *« la juridiction d'origine de MCP est l'intégration locale prototypale ; son extension à des déploiements enterprise suppose une couche additionnelle de durcissement détaillée ci-après »*, la dette systémique observée en 2026 aurait été matériellement plus faible. C'est la sortie directe de la critique RAISE : le périmètre de validité doit être déclaré au moment de la promotion, pas reconstruit après l'incident.
2. **Condition 2, gates institutionnels distincts par niveau de juridiction.** Distinguer publiquement plusieurs niveaux de promotion : prototype local, outil interne, production enterprise, infrastructure agentique critique sectorielle. À chaque niveau, exigences explicites de durcissement, traçables et vérifiables. Il ne s'agit pas de créer une autorité nouvelle, mais de rendre visible ce qui est déjà observé dans les déploiements. Un SDK peut être adapté pour l'un et inadapté pour l'autre, et cette adaptation a un coût qu'il faut nommer.

3. **Condition 3, responsabilité partagée mais explicite selon les trois juridictions.** La responsabilité de l'éditeur, de l'intégrateur, et du framework ou hub de distribution doit être distinguée procéduralement. Aucune des trois ne disparaît, aucune n'absorbe les deux autres. Concrètement : l'éditeur déclare la juridiction d'origine et les hypothèses de sécurité ; l'intégrateur déclare la conformité de son déploiement applicatif à ces hypothèses ; le hub de distribution ou le consortium de standard valide que la convergence de légitimité ne dépasse pas le périmètre déclaré. Si une couche promeut au-delà du périmètre, elle en porte explicitement la responsabilité.
4. **Condition 4, compensation explicite du modèle économique d'accélération.** C'est la condition que la critique économique rend cardinale. Si l'absence de procédure produit un avantage compétitif, sa présence produit un coût. La procédure n'est viable que si elle compense ce coût par un effet de structure : réduction du risque enterprise mesurable et assurable, accès aux marchés régulés via AI Act et équivalents nationaux, prime de confiance vérifiable dans les contrats B2B. Sans cette compensation, la procédure reste perpétuellement contournée. Le sujet n'est pas seulement de demander la procédure ; c'est de la rendre économiquement viable face à la stratégie d'accélération qui prospère sans elle.

Une procédure de promotion ne dit pas qu'un SDK est sûr. Elle dit dans quelle juridiction il peut être déployé comme infrastructure, sous quelles hypothèses, et qui assume la responsabilité de chaque transfert. C'est moins ambitieux qu'une certification. C'est plus opérationnel.

Cette procédure n'a rien d'utopique. Elle existe partout où l'industrie a tiré les conséquences d'un Heartbleed sectoriel. Elle n'existe pas encore pour la couche agent parce que cette couche est trop récente, et parce que son absence est, à court terme, productive pour ceux qui y déploient le plus vite.

V. La dette récurrente, la fenêtre AI Act, et le triptyque

Le règlement européen sur l'intelligence artificielle active ses pouvoirs d'exécution sur les modèles GPAI le 2 août 2026, soit dans environ soixante-dix-neuf jours à la date où j'écris.

L'AI Safety Institute britannique évalue Claude Mythos Preview, modèle « frontière » dont les capacités cyber-offensives sont qualifiées d'unprecedented par l'institut lui-même.

Le Center for AI Standards and Innovation américain signe avec Google DeepMind, Microsoft et xAI un accord d'évaluation pré-publication. La couche modèle s'instrumente, lentement et imparfaitement, à l'échelle régulatoire.

La couche agent ne l'est pas encore. C'est précisément la fenêtre d'action.

La vulnérabilité MCP n'est pas un événement isolé. C'est le premier d'une série prévisible si la procédure de promotion institutionnelle des artefacts de la couche agent n'est pas instaurée. Les couches suivantes sont déjà identifiables :

- protocoles de mémoire d'agent long-horizon,
- frameworks de planification multi-step,
- orchestration de sous-agents,
- dispositifs de capacité d'action contextuelle étendue.

Chacun de ces artefacts traversera, comme MCP, un trajet de la juridiction prototypale vers la juridiction systémique enterprise. Chacun rencontrera la même asymétrie de bénéfice, la même absence de juridiction unique, la même absence de métrique commune. Et si rien ne change, chacun produira sa propre dette d'incident.

Anthropic publie un SDK conforme à son usage d'origine. L'industrie l'intègre. L'écosystème le promeut. La procédure de promotion n'existe pas. Tant qu'on ne nomme pas l'opération comme distincte, on continuera à payer son absence à chaque couche.

Nous avons décrit en février les contraintes physiques. Nous avons décrit en mai l'absence de protocole pour les arbitrer. Il reste à décrire l'absence de procédure pour promouvoir les artefacts qui les exécutent.

Les régulateurs discutent encore des modèles. L'industrie déploie déjà les couches qui les instrumentalisent. L'industrie, elle, a déjà cessé d'attendre.