Liste de vérification de préparation MCP

Ceci est un point de décision à franchir avant de connecter un serveur MCP à un assistant, afin qu'une intégration utile ne devienne pas discrètement un chemin pour une fuite de données ou pour qu'un étranger agisse avec vos accès. Il s'agit de la connexion, de la confiance que vous étendez, et de l'ampleur des dégâts si un token fuit, indépendamment du fait qu'une tâche unique soit sécuritaire à automatiser (c'est la Liste de vérification de préparation de l'agent). Exécutez-la la première fois que vous activez un serveur, et à nouveau chaque fois que le serveur, ses portées ou son éditeur changent.

Valeur et adéquation

- [ ] Vous pouvez nommer la tâche spécifique que ce serveur permet, et pourquoi elle surpasse le fait de coller les données à la main.
- [ ] Aucune option plus restreinte (une portée en lecture seule, un seul système, un outil existant) ne ferait le même travail avec moins d'accès.
- [ ] Le serveur vaut le risque permanent d'être connecté, pas seulement utile sur le moment.

Accès et autorisation

- [ ] Le serveur demande les portées les plus étroites pour le travail, en lecture seule si possible, un système plutôt que tous.
- [ ] Le serveur accepte uniquement les tokens émis pour lui-même et rejette les tokens dont l'audience est un service différent.
- [ ] Le serveur ne transmet pas votre token à une API en aval ; il utilise ses propres informations d'identification pour les appels en amont.
- [ ] Les serveurs distants utilisent OAuth 2.1 avec PKCE sur HTTPS, avec une correspondance exacte de l'URI de redirection (pas de jokers).
- [ ] Un écran de consentement par client nomme le client, les portées tierces, et où les tokens seront envoyés, et vous l'avez effectivement vu.
- [ ] Chaque serveur a ses propres informations d'identification ; aucun token n'est partagé entre les serveurs.

Confiance et chaîne d'approvisionnement

- [ ] Vous avez vérifié l'éditeur et le nom exact du paquet, et ce n'est pas une imitation d'un serveur populaire.
- [ ] Vous avez lu ou scanné le code du serveur et ses dépendances, et épinglé une version exacte.
- [ ] Vous ré-examinerez et ré-épinglerez lorsque la version changera, pour qu'un échange silencieux de définition d'outil (un rug pull) ne puisse pas passer inaperçu.
- [ ] Les descriptions d'outils et les schémas de paramètres ont été inspectés pour des instructions cachées avant approbation.

Confinement et sécurité d'exécution

- [ ] Un serveur local fonctionne sur stdio ou un socket restreint, dans un bac à sable avec uniquement les fichiers et le réseau dont il a besoin.

- [ ] Vous avez vu la commande de démarrage complète, non tronquée, avant qu'elle ne s'exécute, et elle ne touche pas aux clés SSH, aux fichiers système ou à des points de terminaison réseau inattendus.
- [ ] Les descriptions d'outils et les résultats d'outils sont traités comme du texte non fiable, contrôlable par un attaquant, qui peut contenir une injection de prompt.
- [ ] Aucune action à haut risque ne s'exécute automatiquement sur la base du résultat d'un outil; un humain la révise d'abord.
- [ ] Le serveur ne peut pas atteindre les plages d'IP internes ni le point de terminaison des métadonnées cloud à 169.254.169.254.

Rayon d'impact et responsabilité

- [ ] Vous savez ce qu'un token divulgué ou volé exposerait, et la réponse est limitée, pas "tout".
- [ ] Vous pouvez révoquer ou faire tourner l'identifiant rapidement sans briser des flux de travail non liés.
- [ ] Une personne est responsable de cette connexion, et les mises à jour sont revues avant d'être intégrées dans une configuration partagée.
- [ ] La journalisation des appels d'outils est activée dès le premier jour, ce qui permet d'enquêter réellement sur un incident.

Décision

- [ ] Chaque ligne de sécurité est un oui clair; tout non ou inconnu bloque la connexion jusqu'à ce qu'il soit résolu.
- [ ] La décision est enregistrée avec ses portées, conditions, et un déclencheur pour savoir quand la re-vérifier.