Cert-IX

SERVEUR MCP · SÉCURITÉ DES DÉPENDANCES

DepCheck MCP

Guide de l'utilisateur

Tout ce qu'il vous faut pour connecter DepCheck à votre agent de code IA et vérifier les dépendances à la recherche de vulnérabilités connues avant qu'elles n'arrivent dans votre code.

Outil gratuit pour développeurs · Model Context Protocol

ENDPOINT :
`https://mcp.cert-ix.com/depcheck`

Sommaire

1 Présentation

Tout ce qu'il vous faut pour connecter DepCheck à votre agent de code IA et vérifier les dépendances à la recherche de vulnérabilités connues avant qu'elles n'arrivent dans votre code.

- **Endpoint:** `https://mcp.cert-ix.com/depcheck`
- **Transport:** streamable HTTP (MCP)
- **OSV · CISA KEV · EPSS**

2 Connectez votre agent

D'abord, demandez une clé gratuite sur cert-ix.com/contact — nous vous envoyons une clé de la forme cixmcp_… (aucun compte nécessaire). Pointez ensuite votre agent vers l'endpoint avec cette clé.

Claude Code

```
claude mcp add --transport http depcheck https://mcp.cert-ix.com/depcheck \
  --header "Authorization: Bearer <your-key>"
```

Tout client MCP (fichier de configuration)

```
{
  "mcpServers": {
    "depcheck": {
      "type": "http",
      "url": "https://mcp.cert-ix.com/depcheck",
      "headers": { "Authorization": "Bearer <your-key>" }
    }
  }
}
```

> Fonctionne avec Claude Code et tout client qui parle MCP via HTTP. Une fois votre clé active, votre agent peut appeler les outils immédiatement.

3 Les cinq outils

`check_package`

Arguments: `ecosystem` `name` `version`

Vérifie une version de paquet à la recherche de vulnérabilités connues. Renvoie les avis affectant exactement cette version et la version sûre la plus proche vers laquelle mettre à niveau.

`scan_dependencies`

Arguments: `manifest_content` `manifest_name`

Analyse un manifeste entier d'un coup — go.mod, package.json, package-lock.json, requirements.txt, pyproject.toml, Cargo.toml, Cargo.lock, pom.xml, composer.json. Le serveur hébergé prend le corps du fichier, jamais un chemin. Jusqu'à 2000 dépendances par analyse.

`suggest_safe_version`

Arguments: `ecosystem` `name`

Obtient la version la plus récente d'un paquet ne présentant aucun avis connu.

`get_advisory`

Arguments: `id`

Récupère le détail complet de n'importe quel avis (GHSA-…, CVE-…, RUSTSEC-…, GO-…) : description, CVSS, plages affectées et références.

`get_cve_intel`

Arguments: `id`

Consulte les renseignements d'exploitation d'une CVE — statut CISA KEV et score EPSS — pour décider quoi corriger en premier. Un problème listé dans CISA KEV ou à EPSS élevé est activement attaqué et devrait passer en tête de file.

4 Lire un verdict

Champ	Signification
`verdict`	VULNERABLE ou OK pour la version exacte vérifiée.
`advisories`	Chacun avec un id, une gravité et un score CVSS.
`exploitation`	Signalé lorsque le problème figure sur CISA KEV ou présente un EPSS élevé — corrigez-les en premier.
`recommended_min_version`	La version la plus proche au-dessus de la version actuelle qui écarte les avis.

5 Comment vos données sont traitées

- Rien n'est conservé. Votre manifeste est analysé en mémoire pour en extraire les noms et versions de paquets, puis supprimé — nous ne conservons ni vos fichiers ni vos résultats.
- Seules les coordonnées sont vérifiées — nom et version du paquet, jamais le contenu de vos fichiers. Les recherches sont brièvement mises en cache en mémoire pour la rapidité.
- Miroir souverain d'abord. Les vérifications s'exécutent contre notre propre miroir d'avis hébergé sur une infrastructure de l'EU, de sorte que la plupart des recherches ne quittent jamais Cert-IX.

> Repli en sortie : lorsque le miroir ne peut pas répondre avec certitude pour un paquet, le nom et la version de ce paquet sont recherchés via l'API publique OSV.dev (exploitée par OpenSSF). Les coordonnées du paquet — et non le contenu des fichiers — peuvent donc quitter notre infrastructure sur ce chemin. Gardez-le à l'esprit si vous vérifiez des noms de paquets privés ou internes.

6 Usage équitable et limites

- Les requêtes sont limitées en débit par clé (environ 20 par seconde). L'analyse d'un manifeste complet compte pour une seule requête.
- Gardez votre clé privée — une clé par équipe ou par utilisateur. Pour la faire tourner, demandez-en une nouvelle et nous retirons l'ancienne.

7 Dépannage

Réponse	Ce que ça signifie
`401 unauthorized`	Clé manquante ou invalide — vérifiez votre en-tête Authorization: Bearer.
`429 too many requests`	Vous avez atteint la limite de débit — patientez et réessayez.
`scan timed out`	Un manifeste très volumineux ou lourdement chargé en avis — découpez-le en morceaux plus petits.

8 Besoin d'une clé ou d'aide ?

Demandez une clé, posez une question ou signalez un problème — nous serons ravis de vous aider.

Contactez-nous: cert-ix.com/contact · https://mcp.cert-ix.com/depcheck