

## Exploitation en 4 heures : votre fenêtre de patching est morte

📅 15 mai 2026 • 🔄 Mis à jour le 17 mai 2026 • ⌚ 7 min de lecture • ≡ 1138 mots  
• 👁 28 vues • ❤

L'exploitation de PraisnAI en 3h44 le 11 mai 2026 enterre définitivement les fenêtres de patching mensuelles.

Pourquoi le tempo défensif doit basculer en heures, pas en semaines.

Trois heures et quarante-quatre minutes. C'est le temps qu'il a fallu, le 11 mai 2026, pour qu'une CVE PraisnAI fraîchement publiée soit scannée à l'échelle planétaire. Ce chiffre n'est pas un record isolé : c'est la nouvelle normalité. Et il rend obsolète à peu près tous les plans de patching que j'audite encore en 2026.

---

## **Le mythe des 90 jours est mort. Celui des 30 jours aussi.**

---

Pendant une décennie, le standard implicite du patching a été de 30 jours pour les vulnérabilités critiques, parfois 90 pour les autres. Beaucoup de référentiels — y compris ceux que je vois encore dans les politiques de sécurité chez mes clients — codifient toujours ces fenêtres. Elles étaient bâties sur l'hypothèse que la "weaponisation" d'une CVE, c'est-à-dire le temps entre la publication d'un advisory et l'apparition d'un exploit fonctionnel utilisable à grande échelle, prenait au moins quelques semaines.

En 2026, cette hypothèse n'a plus aucun sens. Les chiffres compilés par les équipes threat intel le confirment, et les exemples des deux dernières semaines sont presque caricaturaux : PraisonAI scannée en 3h44, Ivanti EPMM exploitée en quelques heures, Cisco Catalyst SD-WAN livrée avec un acteur étatique déjà à l'œuvre avant même la publication du correctif. Les attaquants ont industrialisé la chaîne entre advisory et exploitation. Les défenseurs, eux, restent calés sur des cycles trimestriels.

---

## **Pourquoi c'est allé si vite**

---

Trois facteurs convergent. D'abord, l'écosystème offensif a internalisé l'IA. Là où il fallait un développeur expérimenté pour transformer un advisory technique en sonde fonctionnelle, on a aujourd'hui des chaînes automatiques où un LLM lit le diff GitHub,

---