

🔊 LUISTER DIT VERSLAG

-10%

1,0x

+10%

+25%

1,0x

↓ Download naar telefoon

▶ 📖 Lees mee met de podcast

Periode: 9 juli t/m 5 augustus 2026. 255 berichten, 32 sessies, 144 bestanden.

Wat werkt goed

- Je queue-systeem werkt echt. Bots pakken taken op en maken ze **zelfstandig** af.
 - Bewijs is de norm: HTTP 200 checks, exit codes, Telegram message-IDs.
 - Claude is eerlijk. Bij de Chrome-cookie-vraag zei hij: "die policy bestaat niet". Drie keer gecheckt, geen blinde fix.
 - Oorzaken worden gefixt, geen symptomen. Voorbeeld: de Sport-categorie verdween steeds. De fout zat in het generator-script en is daar opgelost. Nu blijft hij staan.
-

Wat je tegenhoudt

- Claude beweert soms iets **zonder te checken**. Twee keer zei hij "ik kan geen SSH naar de Rijswijk-pc's", terwijl hij dat script zelf al gebouwd had.
 - Bijna alle frustratie komt uit dit patroon. Niet uit gebrek aan kunnen.
 - Veel werk zit in losse shell-commando's en live edits. Weinig staat in git. Fixes kunnen dus **stil verdwijnen**.
 - Achtergrond-klussen draaien zonder eindcheck. Zo schreef een oude batch in hetzelfde bestand als een nieuwe.
-

Snelle winst

- Maak van vaste shell-reeksen een **skill**: bouwen-en-checken, status-ronde, publiceren. Eén commando in plaats van steeds opnieuw typen.
 - Zet een hook bij sessie-start die test wat deze machine echt kan: SSH, VPS, scherm. Dan is "ik kan niet" een **gemeten feit**.
 - Zet in MEMORY.md een korte lijst: "deze scripts bestaan al en werken". Dat voorkomt de SSH-fout.
-

Op termijn

- Elke queue-taak krijgt een **meetbare eis**: HTTP 200, aantal rijen, bestand-hash of exit code. Claude blijft proberen tot het lukt, of meldt "geblokkeerd" met bewijs. Mislukte taken gaan terug in de queue met alle context erbij.
- Grote batches (zoals de 22 podcasts) niet meer één voor één. Verdeel over parallelle subagents. Elk zijn eigen output-map, elk zijn eigen check.
- Bouw een vaste lijst van wat elke machine kan, plus een korte testset. Dan controleert Claude tegen echte metingen, niet op het oog.