

Opnieuw beginnen zonder de ruis — is dat haalbaar?

Jamal, 14-8-2026: "Stel dat we een hele nieuwe set-up willen maken. Een nieuwe gBrain, een nieuwe skills-lijst, nieuwe wachters. De hooks heb je nog niet gecontroleerd. Elke bot maakt ook hooks en andere dingen die problemen kunnen geven. Welke issues kunnen er nog meer ontstaan behalve met de wachters? Kan je uitgebreid research doen en alle bots checken wat er bij hun loopt en draait, en met een plan komen? Dus alsof ik helemaal opnieuw met Claude zou beginnen, maar dan wel met alle skills en alle andere dingen die nodig zijn om goed te kunnen draaien, maar zonder alle ruis. Is dat haalbaar?"

Het korte antwoord: ja, maar niet zoals je het nu voor je ziet. En de reden waarom is de verrassendste uitkomst van dit onderzoek.

In het kort

1. **Je had gelijk over de hooks.** Die had ik niet geteld. Het zijn er 57 over de vloot, en ze zijn gevaarlijker dan wachters: een hook draait bij elke actie, niet één keer per dag.
2. **Maar de grote verrassing: er is bijna niets dood.** Van de 309 scripts op de MacBook zijn er maar 6 die nergens worden aangeroepen én al 30 dagen niet zijn gebruikt.
3. **Dat betekent dat opruimen het probleem niet oplost.** Ik dacht eerst dat 143 scripts dood waren. Toen ik het beter mat, bleken 137 daarvan gewoon handgereedschap dat wel degelijk gebruikt wordt. Als ik dat niet had nagemeten, had ik je een verkeerd plan gegeven.

4. **Het probleem is niet rommel. Het is omvang en verspreiding.** 627 scripts, 218 geplande taken, 57 hooks, 144 skills, 33 MCP-servers, verdeeld over zeven machines die elkaar niet kennen.
5. **Eén skill is 1,1 GB** en dat is 99% van de hele skills-map. Daar zit 704 MB aan node_modules in.
6. **Helemaal opnieuw beginnen is haalbaar, maar niet door alles weg te gooien.** De waarde zit in 560 geheugenbestanden en 63 regels die je in twee jaar hebt opgebouwd. Dat is precies wat je NIET opnieuw wilt doen.

1. De volledige telling

	MACBOOK	SJAKIE	IMAC	PACO	VPS	TOTAAL
Hooks	24	15	18	?	—	57
Skills	119	25	0	?	—	144
Scripts	309	122	171	?	25	627
Geplande taken	72	45	41	26	34	218
MCP-servers	19	14	?	?	—	33+
Wachters	20	8	7	5	4	44

Niet gemeten: **Coco** (geen SSH-toegang), **Julio** en **Lenovo** (offline). De echte getallen liggen dus hoger.

2. De hooks: het gat dat je zelf aanwees

Hooks zijn stukjes code die automatisch meedraaien bij elke actie. Op de MacBook staan er 24:

MOMENT	AANTAL	WAT ER ONDER ANDERE GEBEURT
PreToolUse	7	machine-check, deploy-lock, backup-vooraf, zelfherstart-guard, beeld verkleinen
UserPromptSubmit	4	context inladen
SessionStart	4	sessie-context, MCP-check
PostToolUse	4	deploy-lock vrijgeven, werklog, telegram-nummering
Stop	4	telegram-afdwinger, checkpoint, obsidian, slack-herinnering
PreCompact	1	opslaan vóór het samenvatten

Waarom dit belangrijker is dan de wachtters: eenwachter draait één keer per dag of per minuut. Een hook draait bij **elke** actie. Zeven PreToolUse-hooks betekent dat er bij elke opdracht zeven stukjes code langskomen die allemaal iets kunnen blokkeren of vertragen.

En er staat iets vreemds tussen: vier hooks met de tekst `null || true`. Dat is een constructie die altijd slaagt, ook als er niets gebeurt. Zulke hooks kunnen stil niets doen zonder dat er ooit een foutmelding komt. Dat is precies de vorm van storing die deze week al vier keer langskwam.

3. De verrassing: bijna niets is dood

Dit is de belangrijkste uitkomst, en hij gaat in tegen wat ik verwachtte.

Eerste meting: 143 van de 309 scripts (46%) wordt door niets aangeroepen. Dat leek een berg dood gewicht.

Tweede meting, strenger: hoeveel van die 143 zijn óók 30 dagen niet in een sessie gebruikt?

Zes.

De andere 137 zijn handgereedschap. Ze staan niet in een geplande taak, maar ze worden wel

degelijk gebruikt — door mij, of door jou, wanneer dat nodig is.

Wat dat betekent voor je vraag: grote schoonmaak levert bijna niets op. Er is geen berg rommel om weg te gooien. Alles doet iets.

Dat maakt het probleem lastiger, niet makkelijker. Je kunt niet snoeien. Je moet samenvoegen.

Let op, eerlijk: mijn eerste meting gaf per ongeluk "0 scriptnamen gevonden in 2112 sessielogs". Dat kon niet kloppen; de zoekopdracht was stil gestrand op te veel bestanden. Had ik dat niet nagerekend, dan stond hier nu "143 scripts kunnen weg" en had je een opruimactie gekregen die 137 werkende gereedschappen had gesloopt.

4. Waar het gewicht dan wel zit

Eén skill is 1,1 GB. De hele skills-map is 1,1 GB, en `gstack` is daar in zijn eentje verantwoordelijk voor:

ONDERDEEL	GROOTTE
<code>gstack/node_modules</code>	704 MB
<code>gstack/browse</code>	152 MB
<code>gstack/make-pdf, design, bin</code>	3 × 61 MB

Daarin zit één bestand van 197 MB (een kopie van de Claude Agent SDK) en twee wasmbestanden van samen 47 MB.

Ter vergelijking: Sjakie heeft 25 skills die samen **772 kB** zijn. Een verschil van meer dan duizend keer.

5. Welke problemen er nog meer kunnen ontstaan

Je vroeg hier expliciet naar. Buiten de wachters om zie ik vier soorten risico.

1. Hooks die stil niets doen.

Vier hooks eindigen op `null || true`. Die melden nooit een fout. Als ze kapot gaan, blijft alles er normaal uitzien.

2. Dezelfde config op meerdere plekken.

gBrain stond vandaag met twee verschillende tokens ingesteld: één in `.claude.json` (werkte) en

één in de Keychain (was dood). Codex las de dode. Dat was maandenlang onzichtbaar.

3. Machines die elkaar niet kennen.

Elke machine bewaakt zichzelf. Valt een machine helemaal weg, zoals Lenovo gisteren negen uur,

dan is er niemand die het merkt. De wachter die dat had moeten zien stond op die machine.

4. Kennis die alleen lokaal staat.

560 geheugenbestanden op de MacBook. Als een feit daar wel staat en niet in het gedeelde geheugen, weet de rest van de vloot het niet. Dat is in augustus al een keer misgegaan.

6. Is helemaal opnieuw beginnen haalbaar?

Ja. Maar niet als "alles weggooien en van nul af aan".

Want wat is de waarde van de huidige opzet? Niet de scripts. Die zijn in een week te herschrijven.

De waarde zit in:

- 560 geheugenbestanden met wat je hebt geleerd
- 63 team-regels, elk uit een echte fout ontstaan
- de gotcha's: precies de dingen die één keer fout gingen en nooit meer mogen

Dat is twee jaar leergeld. Dat wil je juist niet opnieuw betalen.

Wat wel kan: opnieuw samenstellen

Niet opnieuw bouwen, maar opnieuw kiezen wat er in mag.

Begin met een **lege lijst**. Voeg alleen iets toe als het door deze vier vragen komt:

1. **Wat gaat er kapot als dit er niet is?** Kun je dat niet concreet zeggen, dan hoeft het niet mee.
2. **Bestaat er al iets dat dit doet?** Zo ja: dat uitbreiden, niet iets nieuws ernaast.
3. **Hoe merk ik dat het stilvalt?** Geen antwoord betekent: het gaat een keer stil dood.
4. **Wanneer mag dit weer weg?** Alles krijgt een vervaldatum.

Het huidige systeem is nooit door die vier vragen gegaan. Daarom staat er nu 44 keer een wachter en twaalf keer een marketing-tool.

De volgorde

1. **Eerst de vloot-wachter** (loopt al, jij gaf akkoord). Zonder zicht op wat er stilvalt, is elke volgende stap blind.
2. **Dan de hooks**. 57 stuks, ze draaien bij elke actie, en vier ervan kunnen stil niets doen. Dit is het grootste onzichtbare risico.
3. **Dan het gewicht**. gstack van 1,1 GB naar iets normaal, of eruit als je hem niet gebruikt.
4. **Dan de scripts samenvoegen**. Niet weggooien (er is niets dood), maar bundelen wat hetzelfde doet.
5. **Het geheugen als laatste**. Dat is het kroonjuweel, daar blijf je het langst van af.

Wat het NIET moet worden

Een nieuwe schone opzet naast de oude. Dan heb je twee systemen in plaats van één, en dat is precies hoe we hier zijn gekomen.

Wat ik voorstel

1. **Ga door met de vloot-wachter** (je gaf al go).
2. **Laat mij de 57 hooks uitzoeken** zoals ik de wachters heb uitgezocht: wat doet elk, wat overlapt, welke doen stil niets. Dat is research, geen bouwwerk.

3. **Beslis over gstack:** gebruik je die skill? Zo nee, dan is dat 1,1 GB in één klap.
4. **Neem de vier vragen aan als vaste toets** voor alles wat er vanaf nu bij komt. Ook voor mij.
5. **Geef me toegang tot Coco** en zet Julio en Lenovo een keer aan, dan is het beeld compleet.

En één ding dat ik zelf moet doen: ik heb deze week twee keer een nieuwe wachter gebouwd terwijl

er al 44 stonden. Ik ga die vier vragen op mezelf toepassen voor ik nog iets voorstel.