

OPERATOR FIELD GUIDE NO. 03

FREE — NO SIGN-UP WALL

The Daily Brief That Checks Up On You

A morning brief with a second agent whose only job is catching the first one lying

One email every morning with what actually matters today, what you promised people and haven't done, and what you're waiting on. Verified, so it doesn't tell you to chase something you already closed.

45 minutes

TIME TO BUILD

Paid tier

FOR CONNECTORS

No code

SKILL LEVEL

Free version

INCLUDED

WHY THE SECOND AGENT IS THE WHOLE GUIDE

Most morning briefs die in week two

Everyone builds a morning brief. Most people stop using theirs inside two weeks, and the reason is always the same: **it tells you to do things you've already done.** You closed that loop in a Slack thread on Friday afternoon. The brief doesn't know that, because it read your calendar and your notes but not the resolution. So Monday morning it confidently tells you to follow up. Do that four times and you stop trusting it. Stop trusting it and you stop opening it.

Bea hit exactly this:

"At the beginning it was just giving me like rough, you have to do this, but then actually there was a whole other thread on Slack about that. It was very clear that action was done."

BEATRICE ALIPRANDI — EP 11

The fix is not a better brief. It's a **second agent that reviews the first one's output before it reaches you**, and deletes anything already handled.

"So I asked another agent to check on a daily basis before it sends the email with the to-do list and priorities."

BEATRICE ALIPRANDI — EP 11

THE REUSABLE IDEA

That two-stage pattern works far beyond briefs. Any agent that produces a to-do list benefits from a second one whose only instruction is *prove each of these is still outstanding*.

WHAT YOU NEED

Five sources, three of them optional

YOU NEED	OPTIONS THAT WORK
A meeting notes source	Granola, Otter, Fathom, Fireflies, or your own written notes in a doc
A calendar	Google Calendar, Outlook
Email	Gmail, Outlook
A chat tool (optional, but this is where the value is)	Slack, Teams
A task list (optional)	Notion, Todoist, Linear, Asana, a plain doc

The more of these it can read, the better the supervisor works, because completion evidence usually lives in chat rather than anywhere formal.

No connectors? It still works. Skip to the manual version near the end. You paste, it reasons. Same output, more clicking.

PICK YOUR MODEL

The supervisor is long-context work

As of August 2026:

WHAT YOU'RE DOING	ON CLAUDE	ON CHATGPT
Building it	Fable 5	GPT-5.6 Sol
The daily brief run	Sonnet 5	GPT-5.6 Terra
The supervising agent	Sonnet 5 or Opus 5	GPT-5.6 Terra
Auditing your setup	Fable 5 , high effort	GPT-5.6 Sol
Do not use	Haiku	Luna

The supervisor reads across a lot of scattered text looking for evidence. That's long-context work, which is where the cheap tiers fall apart badly. Luna scores around 41% on long-context recall against Sol's 91%. **A supervisor that misses evidence is worse than no supervisor**, because it gives you false confidence in the brief.

STEP 1

Build the brief

The instinct is to ask for a summary of everything. Don't. A brief that lists everything is a second inbox.

THE BRIEF PROMPT

You are my chief of staff. Every weekday morning you write me one short brief.

WHO I AM: [role] at [type of company]

WHAT I ACTUALLY OWN: [e.g. finance, legal, people ops, investor relations]

WHO I CANNOT LET DOWN: [e.g. my CEO, the board, two key customers]

WHAT A GOOD DAY LOOKS LIKE: [one line]

Read, for the last 3 days: my calendar, my email, my meeting notes, and my messages. Then write the brief below.

RULES, absolute:

1. Maximum 10 items across the whole brief. If there are more, you chose wrong. Rank ruthlessly.
2. Every item must have a source. Name the meeting, the thread or the email. No item without a source.
3. Never invent a commitment. If I did not clearly say I would do something, it does not go in the promises section.
4. Do not summarise my meetings. I was there. Only tell me what needs an action or a decision from me.
5. No greeting, no motivational line, no sign-off.

THE BRIEF:

TODAY, IN ORDER

Up to 3 things. What genuinely has to happen today and why today. If something has a hard external deadline, say so.

I SAID I WOULD

Things I committed to and have not visibly done. For each: what I said, where I said it, who is expecting it, and how long ago. Quote me.

WAITING ON OTHERS

What I asked for and have not received. Who, what, how long. Mark anything older than 5 working days as chase.

MEETINGS THAT NEED PREP

Only meetings where turning up cold would be bad. One line on what to prepare.

QUIETLY GOING WRONG

Anything drifting that nobody has escalated. A thread that died, a decision nobody made, a question asked twice with no answer. Say "nothing" if there is nothing. Do not manufacture something here.

RULE 3 EXISTS FOR A REASON

Without it these briefs invent commitments out of politeness. Somebody says "we should look at that" in a meeting and by Tuesday morning it's on your list as a promise you made. Quoting yourself is the guard against it.

"Quietly going wrong" is the section that earns its keep. It's also the one that will produce nonsense if you let it, which is why it's explicitly allowed to say nothing.

Build the supervisor

This is the part almost nobody builds. It runs *after* the brief, on the brief's own output, before you ever see it.

THE SUPERVISING AGENT PROMPT

You are a verification agent. You did not write the brief below. Your job is to find everything in it that is already handled, and cut it.

You are adversarial. Assume the brief is out of date. Your default assumption is that some of this is already done.

For EVERY item in the brief:

1. Search my messages, email and meeting notes from the last 5 days for evidence that it is already complete, already answered, cancelled, or handed to someone else.
2. Look especially at: replies in the same thread, later messages from me, later meetings on the same subject, and anything sent after the item was created. Completion evidence is usually in chat, not in a task list.
3. Classify each item:
 - DONE – clear evidence it is complete. Quote the evidence.
 - MOOT – no longer relevant, cancelled, or someone else owns it now. Quote the evidence.
 - STILL OPEN – no evidence of completion anywhere.
 - UNCLEAR – ambiguous evidence.

RULES:

- To mark anything DONE or MOOT you must quote the exact message or line that proves it. No quote, no removal. This is not negotiable.
- If you are unsure, mark it UNCLEAR and leave it in. A stale item is annoying. A dropped item is a broken promise.
- Do not add new items. Do not rewrite items. You only verify and cut.

Output:

1. The cleaned brief, with DONE and MOOT items removed entirely.
2. A short section at the bottom, "Removed as already handled", listing what you cut and the quote that justified it.
3. UNCLEAR items kept in the brief but marked with a question mark.

THE ASYMMETRY IN THE SECOND RULE IS DELIBERATE

A stale item wastes ten seconds. A dropped item loses you someone's trust. So the supervisor is instructed to err towards leaving things in, and it needs hard evidence to remove anything.

The "Removed as already handled" section is not decoration. It's how you learn to trust the thing. For the first fortnight you'll read it and check. Then you'll stop, because it'll be right.

Have it delivered

On Claude

Ask, in a Project where both prompts live:

SCHEDULED TASK

Set this up as a scheduled task, every weekday at 07:00 my time.

Stage 1: run the daily brief prompt.

Stage 2: pass that output to the supervising agent prompt and let it verify and cut.

Stage 3: put the final cleaned brief in my inbox as a draft addressed to me, subject line "Brief – [date]".

Only the verified version reaches me. I never see the unverified draft.

The draft-in-inbox thing. Bea's lands as a draft she has to click send on: *"it is a draft in my inbox, and I have to send it to myself every morning. It's a small click that I can live with."* Ceci has the same annoyance on a different workflow: *"I have a similar workflow for something completely different and it sits in my drafts."*

So expect the click. Two ways around it if it bothers you:

- **Deliver it somewhere else.** Post to a private Slack channel or DM yourself. Chat delivery usually avoids the send-permission problem entirely, and honestly a brief reads better in Slack than in email.
- **Reframe it.** The click is a one-second review gate. You glance at the subject and the first line before it becomes a real email. That's not the worst design.

On ChatGPT

Use Tasks for a weekday 07:00 run. Chain the two stages in a single instruction:

"First produce the brief. Then act as the verification agent on your own output and give me only the cleaned version."

It's a weaker version of the pattern — one model checking itself is less rigorous than a genuinely separate pass — so be explicit: *"Adopt the verification role completely. Assume the brief you just wrote is out of date and try to prove it."*

Install as a skill on Claude

Folder `daily-brief`, file `SKILL.md`:

```

---
name: daily-brief
description: Produce a verified morning brief – today's priorities, promises
  the user made and hasn't kept, what they're waiting on, meetings needing
  prep, and things quietly going wrong. Runs a second verification pass to
  remove anything already handled. Use when the user asks for their daily
  brief, morning brief, what's on today, or what they're forgetting.
---

```

Both prompts underneath, clearly labelled Stage 1 and Stage 2. Zip, rename to `.skill`, upload.

STEP 4

Make it fail loudly

A brief that arrives looking normal but was built from half the data is genuinely dangerous, because it reads as complete.

FAILURE INSTRUCTION

If you cannot reach any source – calendar, email, messages, notes – do not produce a normal brief. Put this at the very top, before anything else:

"INCOMPLETE BRIEF. Could not reach: [source]. Treat the sections below as partial."

If the supervising agent cannot run, mark the brief clearly as UNVERIFIED at the top. Never send a brief that looks verified when it is not.

"If they can't reach the information, make it say: Hi, I'm the fairy, I couldn't reach the information and tag you so that you actually know when it's failing. Otherwise, it's not a process that you are keeping."

BEATRICE ALIPRANDI – EP 11

STEP 5

Audit it, because this one can read everything

This workflow has the broadest data access of anything in this series. It reads your calendar, your email, your messages and your notes. **That's most of your working life in one automation.**

Point your strongest model at it once a month:

SECURITY AUDIT

Review my daily brief automation. Tell me exactly: what can it read, what can it write or send, whose credentials it uses, and what happens if a connector breaks or a permission changes. Flag anything where the access is wider than the job needs. Be specific about the risk.

THIS IS NOT THEORETICAL

This is not theoretical. On the episode Bea ran a sweep like this and it found a real problem: an authentication flaw in one of her own scheduled tasks, left over from reconnecting broken connectors.

The lesson isn't that the tool was unsafe. It's that **reconnecting a broken connector can silently leave things in a worse state than before**, and you won't notice by looking at the output. Run the audit on Claude Fable 5 or ChatGPT Sol. Cheap reviewers rubber-stamp.

THE GENERAL CADENCE

Every three prompts, check for bugs. Every ten, check for security.

NO CONNECTORS

The free version

If you can't or won't connect anything, this still works. Every morning, paste in:

- your calendar for today and yesterday
- anything you wrote in your notes app yesterday
- the last two days of messages from your two or three most important channels

Then use both prompts in sequence. Two minutes of copying, and you get 90% of the value. **The supervisor is actually the more important half, and it works fine on pasted text.**

HONEST LIMITS

When not to do this

- **Don't let it write to anyone but you.** A brief that reads is low risk. A brief that starts sending your chase-ups is a different thing entirely, and one bad inference becomes an awkward message to your CEO.
- **Don't run it on a day you're off.** Coming back to five briefs is worse than coming back to none.
- **Don't let it own your task list.** It surfaces; you decide. Give it write access to your tasks and within a fortnight your list is full of things you never agreed to.
- **Kill it if it overlaps something else.** Bea retired a weekly roundup for exactly this reason: *"I don't use it that much anymore because I have the daily brief and it like overlaps."* Two automations telling you similar things means you'll ignore both.

TROUBLESHOOTING

When it misbehaves

"It's still telling me to do done things."

The supervisor isn't reading the right places. Name them explicitly: *"Check the thread the item came from, and any message I sent after it."*

"It's cutting things that aren't done."

Tighten the evidence rule: *"You removed an item without quoting proof. Never remove anything without a verbatim quote."*

"It's too long."

The 10-item cap is being ignored. Reply: *"You gave me 19 items. The limit is 10. Cut the 9 least important and tell me what you cut."*

"It invented a promise I never made."

Rule 3. Reply: *"Quote where I committed to that. If you can't, remove it and never infer a commitment again."*

"It's useless on Mondays."

It's only looking back three days and missing the weekend. Widen to five for Monday runs.

THE POINT OF THIS

Build the supervisor

The brief isn't the clever part. Anyone can ask an AI what's on today.

The clever part is the second agent that assumes the first one is wrong. Most automations fail not because they can't do the work, but because they can't tell what's already been done, so they slowly become noise and you quietly stop reading them.

Build the supervisor. It's the difference between a tool you use in month six and one you abandoned in week two.

This guide came out of Top of the Ops, Season 1 Bonus — ["Ride or Die AI Workflows"](#), where Bea and Ceci went through every automation they actually run. The rest of the Operator Field Guides live on the [Free Guides](#) page.

Built something better? Tell us. We'll publish the improved version and credit you.