IT SECURITY BRIEF

PraxisOS: what it is, what it can reach, and what leaves the machine

For the IT team reviewing an install request

Prepared by: Stefan Urness, Better Execute **For:** IT teams reviewing a PraxisOS install request **Date:** September 2026

1. WHAT PRAXISOS IS

PraxisOS is a configuration and scheduling layer that runs on **Claude Code**, Anthropic's official command-line agent (npm package @anthropic-ai/claude-code). Better Execute did not build the AI. We built the roles, schedules, memory and guard rails that turn Claude Code into a personal assistant that works on a timetable and reaches its owner on their phone.

Better Execute (betterexecute.ai) was founded by Mike Urness; the product lead is Stefan Urness (stefan@betterexecute.com). PraxisOS is installed for leadership teams at Australian companies including Moddex, whose IT team reviewed the same questions before approving it. References on request.

Two things that are easy to miss. PraxisOS is on our site at betterexecute.ai/ai-personal-assistant, with a page written for IT reviewers at betterexecute.ai/insights/messaging-channel-risk/it-brief, but neither surfaces when you search the name. And the public GitHub projects named PraxisOS are not ours. Our repositories are private; clients receive a release snapshot, not repository access.

2. WHERE THE AI RUNS

Nothing runs locally except scheduling and file handling. The model is Claude, served from Anthropic's API over TLS. These are the same Claude models Microsoft offers inside Copilot and that Anthropic's own Cowork feature uses. Claude models are proprietary to Anthropic; there is no open-source variant and no local model.

Hardware requirement: a normal laptop with 8 GB of RAM and Node.js 22.13 or newer.

The one optional exception: voice-note transcription, if enabled, downloads a 75 MB Whisper model once and runs it on the CPU. It is off unless the user asks for it.

3. WHAT GETS INSTALLED, AND WITH WHAT PRIVILEGES

Component	Where	Privilege
Windows Subsystem for Linux 2 (Ubuntu)	Windows feature	Administrator, once, to enable the feature. The only elevated step.
Node.js 22 LTS	Inside the Linux environment	User
Claude Code CLI (Anthropic)	Inside the Linux environment	User
PraxisOS release snapshot	~/praxisos inside the Linux environment	User. Plain files, readable by IT.
pm2 process manager	Inside the Linux environment	User level. Keeps the assistant running.
Telegram Desktop (optional; the phone app is enough)	Windows	User

No Windows services, no drivers, no scheduled tasks running as SYSTEM, and no listening ports exposed off the machine (a local status page binds to localhost only). Native Windows install is supported but not recommended. The WSL2 route is what we

ask for because it is more reliable and because it gives you the fence described next.

4. WHAT IT CAN REACH: THE FENCED PROFILE WE RECOMMEND

By default WSL2 exposes the Windows drive to Linux at /mnt/c. **We recommend switching that off for this install.** Two documented Microsoft settings in /etc/wsl.conf do it:

[automount]
enabled=false

[interop]
enabled=false
appendWindowsPath=false

With these set, the Linux environment cannot see the Windows filesystem and cannot launch Windows programs. The assistant then only sees what the user deliberately places in its Linux home or connects as an account. Nothing on the C: drive, no network shares, no Outlook data store. This is a tighter boundary than a desktop agent running directly on Windows, which reads whatever folders the user grants on the PC itself.

On top of that, Claude Code has an OS-enforced sandbox (bubblewrap on Linux) that can deny reads of listed paths and restrict outbound domains. These settings can be locked by managed configuration so the user cannot change them. Reference: code.claude.com/docs/en/sandboxing and code.claude.com/docs/en/security.

Stated plainly
Inside that fence, routine work (reading, drafting, organising, research) runs without a per-command prompt, because the assistant's purpose is to do scheduled work while nobody is at the keyboard. Actions that leave the machine with consequences, such as sending an email or changing a calendar, follow the approval model in our IT reviewer brief: the owner approves on the phone channel, and if the approval cannot be delivered the action does not happen. Two safeguards stay on regardless: a hardline block list that refuses operations with no recovery path (destroying filesystem roots, raw device writes, shutdown, fork bombs), and a rule that the assistant cannot rewrite the configuration files those guards are wired from.

WHY THIS IS SAFER THAN CLAUDE CODE ON ITS OWN

Claude Code is a general tool. Installed by an individual, every safeguard is a choice that person makes in the moment, and the common pattern is to approve everything so the prompts stop, with no floor underneath and no record of what ran. PraxisOS ships the safeguards pre-written and keeps them out of the assistant's reach:

- A hardline block list evaluated before every action in every mode, refusing operations with no recovery path.
- Enforcement files the assistant cannot edit, so the gates cannot be switched off from inside.
- Input guards and a loop detector that stop runaway or injected instructions.
- An event log of what each role did, readable by IT.
- A paired phone channel: the Telegram bot answers only its owner, with pairing codes, rate limits and lockout.
- Secrets stored with restricted file permissions, and one read-only release token per client, revocable on our side.
- The fenced environment in section 4 as the outer boundary.

A business running PraxisOS is running Claude Code in a more controlled configuration than an employee who installs it alone.

5. NETWORK EGRESS (ALL OUTBOUND ON 443, NO INBOUND)

Host	Purpose
api.anthropic.com	The model
console.anthropic.com, claude.ai	Account login (one-time browser sign-in)
api.telegram.org	The phone channel

Host	Purpose
graph.microsoft.com, login.microsoftonline.com	Mail and calendar, only if the owner connects Microsoft 365. Read-only scopes (Mail.Read, Calendars.Read) in the first phase; see the IT reviewer page
api.github.com, github.com	Release download and updates, using a read-only token scoped to one private repository
nodejs.org, Ubuntu package mirrors	Install time only
huggingface.co	Optional, once, only if voice transcription is enabled

On the Telegram route nothing passes through Better Execute servers. We have no telemetry and no remote access to the machine in the standard setup; the optional support connection (a Tailscale tailnet scoped to one machine) is described on the IT reviewer page and is off unless you ask for it. Updates are files the user pulls; we cannot push anything to an installed machine.

6. DATA AND TERMS

What leaves the machine is what the assistant reads in order to answer: the messages sent to it and the files it is given, transmitted to Anthropic's API over TLS, exactly as with Cowork or the Claude option in Copilot. Telegram messages travel through Telegram's Bot API, which is transport-encrypted but not end-to-end encrypted. If that is a concern, the phone channel can be disabled and the assistant used from the terminal only.

Anthropic's terms, as published September 2026:

Consumer plans (Pro, Max): model improvement is opt-in since March 2026. We switch it off during setup; IT can verify at claude.ai/settings/privacy.

Commercial plans (Team, Enterprise, API): "By default, we will not use your inputs or outputs from our commercial products to train our models." A Team seat can be used instead of a personal one if IT prefers commercial terms; nothing else changes.

Anthropic holds SOC 2 Type II, ISO 27001:2022 and ISO/IEC 42001:2023 (trust.anthropic.com).

7. HOW THIS COMPARES WITH THE ALTERNATIVES YOU NAMED

	PraxisOS (fenced WSL2 profile)	Claude Cowork (desktop app)	Copilot Studio agents
Where the model runs	Anthropic	Anthropic	Anthropic via Microsoft, admin-enabled
What it can read	Only what is placed inside the Linux environment	Folders the user grants on the Windows PC	Microsoft 365 data the agent is scoped to
Runs unattended on a schedule	Yes; that is its purpose	Session-based on the desktop	Yes, within Microsoft 365
Reaches the user on their phone	Telegram	No	Teams
Named roles with memory across days	Yes	Limited	Per agent, built by you

They are not mutually exclusive. Cowork suits one-off desktop tasks; PraxisOS carries the standing work.

8. THE ALTERNATIVE HARDWARE ROUTE

If any foothold on a domain-joined machine is unacceptable, the cleanest answer is a dedicated machine per person (a Mac Mini is enough), not joined to the domain, with nothing on it but the assistant and the accounts that person chooses to connect. The install is then entirely outside your managed estate.

9. WHAT WE OFFER IT

1. **A twenty-minute call**
   With the engineer who does the install. No slides.

2. **First seat only, then inspect**
   IT inspects the machine (files under ~/praxisos, /etc/wsl.conf, listening ports with ss -tlnp, outbound connections) before any other seat goes in.

3. **Your firewall list and the sandbox config**
   The egress list above, and the Claude Code sandbox configuration as a file you keep.

4. **Revocation at any time**
   Delete the Linux distribution and the install is gone. The release token is revoked on our side per client.

Contact
Stefan Urness, Better Execute. stefan@betterexecute.com, +1 303 704 6439.