

Co.Property | Mac Mini teammate handoff

Start here

This handoff lets an authorized teammate administer the existing Co.Property office Mac mini from their own computer. The company OS stays on the mini. This package prepares laptop access; it does not migrate the server.

Prepared 20 October 2026, Asia/Kolkata. Internal distribution only. The ZIP contains public host identity and private-network addresses, with no private keys, passwords, access tokens, provider secrets or company records.

LIVE CHECK, 20 October 2026: the company Tailscale peer is online at 100.103.153.38. Strictly pinned, key-authenticated SSH succeeded. Account coproperty, hostname coproperty-Mac-mini, project path and ED25519 fingerprint match. macOS reports 27.0.1. Codex 0.163.0-alpha.10.2 is on PATH and logged in using ChatGPT. The loopback viewer listens on 8790. This verifies the existing operator connection; a new teammate still requires their own enrollment and acceptance.

First access requires two independent owner steps: invite the teammate's own identity/device to the company Tailscale network, then authorize their newly generated SSH public key on the mini. Possession of this ZIP alone grants no access.

Read the owner checklist first, install the client tools, prepare your SSH configuration, send only your .pub file to the owner, verify the mini identity, then test off office Wi-Fi.

Connection card

SSH alias: coproperty-office

Host: coproperty-mac-mini.tail3ab9e5.ts.net

Tailscale IP verified 20 October 2026: 100.103.153.38.

macOS account: coproperty

Project: /Users/coproperty/Projects/coproperty-office-os

ED25519 host fingerprint:

SHA256:UUFmlsdclEmRtyRWNqAdVYUoFxeUkRUGhbzObQPsOPg

Viewer/admin loopback port: 8790

Staff PWA: <https://coproperty-office-os.vercel.app>

Client API: <https://coproperty-office-api.coproperty-office.workers.dev>

The included public host key comes from the operator's existing known_hosts. Its fingerprint was freshly compared with /etc/ssh/ssh_host_ed25519_key.pub over the existing strictly pinned SSH connection on 20 October and matches. Confirm this fingerprint with the teammate through a trusted channel. A changed key requires investigation.

This is ordinary macOS OpenSSH carried over Tailscale. No public port forwarding, second VPN daemon or separate Tailscale SSH server is required. Keep Co.Property in its company-controlled tailnet, separate from other clients.

Owner: grant access

1. Confirm teammate identity and duties. The shared coproperty account exposes its accessible files and sessions; grant this only to trusted technical operators. A restricted separate macOS account needs a separately reviewed setup and adjusted paths.
2. Invite their own identity to the Co.Property tailnet, approve their device if required, and review access to this mini on TCP 22. Do not grant tailnet administrator rights just for SSH. Review actual existing policy before changing it. Do not move the mini into another client or personal network.
3. Confirm mini power/network/unlock, Tailscale connection and Remote Login for the intended account in System Settings > General > Sharing. Preserve FileVault.
4. Receive only the teammate's .pub file over a trusted channel. Compare its fingerprint with them using `ssh-keygen -lf PUBLIC_KEY_FILE`. Never request their private key or passphrase.
5. Using an existing trusted owner session, back up `authorized_keys` privately, ensure `~/.ssh` is mode 700 and `~/.ssh/authorized_keys` is mode 600, and append exactly the reviewed single public-key line. Preserve existing entries. Record key ownership and a review date privately. Never execute received key text as shell commands.
6. On the mini, run `ssh-keygen -lf /etc/ssh/ssh_host_ed25519_key.pub` and confirm the fingerprint through a trusted channel before the teammate installs the supplied pin.
7. Grant GitHub and application access separately as needed. Confirm the intended company Codex/provider account before authorizing agent use. Witness first SSH identity and hotspot tests.

Install the client tools

Install official Tailscale from <https://tailscale.com/download>. Use the teammate's invited identity and select the Co.Property tailnet. Confirm the mini is visible and online. Never use an owner's password or copied Tailscale auth token. On macOS avoid competing app/daemon installations.

macOS: open Terminal and check `ssh -V`. OpenSSH is normally built in. Enable Tailscale launch at login through its supported settings.

Windows: open PowerShell and check `ssh -V` and Get-Command `ssh-keygen`. If missing, install OpenSSH Client through Windows Optional Features. Your laptop does not need OpenSSH Server. Install Tailscale for Windows and accept the company invitation. WSL uses separate SSH configuration; these instructions target native Windows.

Linux: follow the official Tailscale distribution instructions and install your distribution's `openssh-client` package. Complete the supported sign-in flow with the intended company identity.

Use the Tailscale app device list or `tailscale status` where the CLI is available. If no company peer appears, ask the owner to confirm the active tailnet, membership, device approval and mini availability before modifying SSH. Software installation may require your computer administrator.

Prepare your own SSH key

Extract the ZIP fully and inspect the scripts before running them. The helpers create a dedicated key with a passphrase prompt, back up existing config/known_hosts, append the supplied alias and install the supplied public host pin when absent. They do not install Tailscale or authorize server access.

macOS/Linux, from the extracted folder:

```
```sh
bash scripts/setup-macos-linux.sh
```
```

Windows PowerShell, from the extracted folder:

```
```powershell
powershell -NoProfile -ExecutionPolicy Bypass -File .\scripts\setup-windows.ps1
```
```

Use this process-scoped invocation only if company policy permits it; otherwise use the manual route. Do not change the machine-wide execution policy.

Manual route: generate an ED25519 key at ~/.ssh/coproperty-office-teammate with ssh-keygen -t ed25519 -a 64 -f ~/.ssh/coproperty-office-teammate. Choose a passphrase. After reviewing conflicts, append config/ssh-config.txt to your SSH config and the verified config/known_hosts public entry to known_hosts. Unix directory mode is 700; key/config/known_hosts mode is 600. Windows uses your user-profile ACLs; ask an administrator to restrict them if SSH reports open permissions.

Send only coproperty-office-teammate.pub to the owner. Keep the file without .pub and its passphrase on this computer. Existing keys are preserved. Existing aliases stop the helper for manual review; check any SSH Include files and broad Host rules yourself before setup. Backups remain locally in ~/.ssh.

First SSH login and identity

After the owner confirms Tailscale membership and key authorization:

```
```sh
ssh coproperty-office
```
```

Enter your local key passphrase when prompted. Stop on a changed host key. A password prompt alone does not prove the intended public key authenticated.

Inside the mini shell:

```
```sh
whoami
scutil --get LocalHostName
sw_vers -productVersion
cd /Users/coproperty/Projects/coproperty-office-os
pwd
command -v codex
codex --version
codex login status
```
```

Expected account: coproperty. Expected LocalHostName: copropertys-Mac-mini. Verify the project path with the owner. scripts/check-connection.sh collects these basic observations on macOS/Linux; review output before sharing it.

Codex login status shows runtime authorization, not proof of the correct company tenant. Confirm identity with the owner before agent work. Repeat SSH and your actual remote editor from a phone hotspot with office Wi-Fi disconnected. Record date and result; office-LAN success alone is insufficient travel proof.

Codex and remote editor

On a supported Codex desktop build, open Settings > Connections > SSH, choose the concrete coproperty-office alias, connect, and select /Users/coproperty/Projects/coproperty-office-os. Follow the official remote-connections reference for your installed release. Desktop/platform support may vary; the terminal SSH path remains the baseline.

The recorded mini setup made codex discoverable through ~/.local/bin and ~/.zshenv. Its bundled binary was /Applications/ChatGPT.app/Contents/Resources/codex-cli/bin/codex. Recheck command -v codex; do not reinstall or copy another user's authentication directory to fix PATH.

Terminal alternative:

```
```sh
ssh -t coproperty-office
cd /Users/coproperty/Projects/coproperty-office-os
codex
```
```

The last two commands run in the remote shell. SSH, Codex subscription, repository permission and provider identity are separate boundaries.

VS Code option: install Microsoft's Remote - SSH extension, connect to coproperty-office and open the same directory. Coordinate remote editor-server installation with the owner. Coordinate overlapping sessions with the owner before making changes.

Private admin and office TV

The documented viewer listens on the mini's loopback port 8790. Confirm the service is running and your local port is free, then:

```
```sh
ssh -N -o ExitOnForwardFailure=yes \
-L 127.0.0.1:8790:127.0.0.1:8790 coproperty-office
```
```

Keep the terminal open. Browse <http://127.0.0.1:8790/admin> or [/records](http://127.0.0.1:8790/records); [/tv](http://127.0.0.1:8790/tv) is the display view. Ctrl-C ends the tunnel. macOS/Linux may use `scripts/open-admin-tunnel.sh`.

The viewer validates Host/Origin for port 8790; do not substitute another local port without reviewing its contract. If occupied, close only your own conflicting viewer/tunnel. Do not kill an unknown process.

This tunnel reaches the real company mini. Admin changes may change the physical office TV, so coordinate them. The staff PWA has its own protected cloud route and separately provisioned login. A tunnel is not remote desktop control, and a web page rendering does not prove physical TV operation.

Company OS architecture

The business operating model has five areas: Managing, Selling, Planning, Delivering and Learning. The staff dashboard, mobile PWA, office TV and Hermes act on shared company state.

Provider observations + company sources -> authoritative mini / SQLite -> bounded publication -> Cloudflare Worker and D1/KV -> Vercel staff PWA and safe display projections. Operator computers administer the mini using Tailscale and OpenSSH. Cloud pages must not depend on reaching a private mini IP.

Local stores have distinct purposes: knowledge.sqlite supports cited retrieval; operations.sqlite retains events, identity links and operational evidence; company-records.sqlite preserves historical exports; portfolio.sqlite stores observed provider portfolio/reservation coverage. Inspect installed source for current OSDB files/schema and freshness. Historical counts are not current finance or cleaning completion.

Hospitable owns provider reservation and availability transactions. Turno evidence needs verified property/provider mapping and complete current captures. Keep native IDs and provenance; do not join on display names. Notion-derived collections preserve the company's operating model and source IDs. Task/audit state is durable; knowledge indexes are rebuildable; KV is cache.

The mini remains authoritative writer. Protected cloud replicas/projections do not imply a completed writable cloud cutover or per-person role/property controls. Shared pilot authentication remains a limitation. TV is a passive safe projection; private finance, contacts, HR and credentials stay off-screen. Stale, partial and unavailable evidence must remain explicit.

Hermes, providers and delivery state

Hermes is the existing company assistant. Its conversation interface and scheduled refresh work are separate responsibilities. The company system, dashboard and display share the same operational records.

Remote access and core checks have passed. Final integration and verification of the latest improvements remain in progress through the ongoing implementation work. See [START-HERE.html](#) and [docs/Status-checklist.md](#) for the current checklist.

Company Drive uses an existing company-owned connection for the verified SOP reading path. Broader search and upload coverage remain to be completed. Confirm company identity before linking any service.

Power, services and recovery

Live power readback on 20 October confirms AC sleep=0 and autorestart=1; FileVault is On. Existing com.coproperty.awake runs caffeinate -dis. Existing com.coproperty.tailscale-login opens the installed Tailscale app at user login. The viewer setup uses com.coproperty.wallboard. Live launchctl readback confirms the awake, Tailscale-login and wallboard labels; the viewer listens on 127.0.0.1:8790. Inspect current state before changing services and avoid duplicates.

FileVault remains enabled. User LaunchAgents run after login; autorestart and keep-awake do not establish pre-unlock availability. Attended cold-boot recovery is unverified. Arrange a person at the mini before testing restart; do not reboot it unattended for diagnosis.

Useful read-only checks after verifying target identity:

```
```sh
pmset -g custom
fdesetup status
launchctl list | grep coproperty
lsof -nP -iTCP:8790 -sTCP:LISTEN
```
```

A listener proves a process, not fresh provider data. Inspect logs locally and redact before sharing. Use the project's documented SQLite online-backup/restore procedure; a raw copy of an open DB is not a proven consistent backup.

Keep company databases and credentials out of shared code and handoff packages. Do not overwrite the runtime with an old ZIP or use a private data-migration archive for laptop onboarding. That separate archive can contain company data and credentials. This access kit intentionally excludes it.

Troubleshooting and revocation

Hostname not found / no company peer: verify Tailscale running, invited identity, active Co.Property tailnet, device approval and MagicDNS. Owner checks mini power/network/unlock. The initial probe failed here; the later 20 October recheck passed with the company peer online and SSH authenticated. The old IP is context only; owner must reconfirm before a fallback config with the same pin is prepared.

Timeout: check peer visibility, TCP 22 policy, Remote Login, firewall and mini power. Do not open public router ports.

Permission denied: check exact authorized .pub key, IdentityFile, intended account, file ownership/permissions and allowed Remote Login user. Never transfer the private key as a fix.

Host key changed: stop. Owner verifies the physical mini fingerprint and any legitimate rebuild. Preserve prior evidence. Never disable strict checking or blindly trust ssh-keyscan output.

Codex absent: verify installed path and noninteractive PATH with owner. Do not copy auth directories or blindly reinstall the company runtime. Tunnel failure: check local 8790 occupancy, SSH and remote viewer separately. Stale UI needs producer/replica freshness checks.

Revocation: owner removes/suspends the teammate's Tailscale identity/device and removes only their `authorized_keys` entry, preserving others. Terminate existing SSH/editor/tunnel sessions separately; key removal does not close active sessions. Review GitHub/application grants and exposed credentials. Remove the local alias/pin only when unused; never restore a whole old config over unrelated later changes.

Acceptance and references

Owner checklist: identity approved; tailnet invitation accepted; device approved; TCP 22 permitted; teammate public key authorized; host fingerprint independently confirmed.

Teammate checklist: SSH key login; correct account/hostname/project; project location confirmed; intended Codex identity/PATH verified where needed; actual remote editor working; private tunnel if needed; repeat SSH/editor off office Wi-Fi. Record tester, date, environment and pass/fail privately.

Package validation is distinct from teammate installation and live mini acceptance. No invitation, policy edit, server authorization or remote installation was performed for this package. Windows helper execution requires a Windows test; none was available here.

Official references checked during preparation:

<https://tailscale.com/docs/how-to/quickstart>

<https://tailscale.com/download>

<https://learn.chatgpt.com/docs/remote-connections>

https://learn.microsoft.com/en-us/windows-server/administration/openssh/openssh_install_firstuse

This is a dated access and orientation guide. The technical owner retains detailed verification records separately.

Package contents: PDF + editable Markdown guide, README-FIRST.md, owner checklist, local setup helpers, SSH alias/public host pin, connection/tunnel helpers, validation notes and SHA256SUMS.txt. Share privately with the intended teammate.