

Mac Mini Setup + Laptop Sync Guide

Prepared for Jeff Parker **Goal** Get your Mac mini running as the always-on Affirmology backend server. You interact with it primarily from your laptop through Claude Desktop. Files exchange seamlessly between the two machines via Tailscale + SSHFS. The nightly corpus build runs on the Mac mini; the laptop is your control surface.

Approximate time 90 minutes if everything cooperates. Done once, never again.

Architecture Overview

Where	Role
Mac mini (always on, plugged in)	Runs the corpus scraper nightly. Hosts the SSD with all the data. Runs Claude Code for direct backend tinkering.
Laptop (your daily driver)	Claude Desktop, code editing, monitoring. Reads/writes the Mac mini's files through a mounted folder that looks like a local drive.
SSD (Affirmology)	Lives on the Mac mini, holds the corpus database and raw scraped text. The Mac mini accesses it directly; the laptop accesses it through the SSH mount.
Tailscale	Private, encrypted network between your laptop and Mac mini. Survives Wi-Fi changes, hotel networks, traveling. No port forwarding, no DDNS.

Phase 0: First Boot With Wireless Peripherals Only

You don't need a wired keyboard, wired mouse, or CD drive. None of those have shipped with a Mac mini in over a decade. The M-series Mac mini's Setup Assistant pairs Bluetooth devices during the welcome screens.

What you need on hand: - HDMI monitor (your switcher is fine) - Power cable for the Mac mini - One wireless keyboard or wireless mouse with Bluetooth (NOT a USB receiver dongle — has to be real Bluetooth). Apple Magic Mouse/Keyboard, Logitech MX, Keychron, etc. all qualify. - Wi-Fi password handy

What to do: 1. Plug the Mac mini into power and HDMI. Don't power it on yet. 2. Put your wireless device in **pairing mode** (usually: hold the power button for 3 seconds until the LED blinks). 3. Power on the Mac mini. 4. The Apple logo appears, then "Hi" → welcome screen. The Setup Assistant will display a message asking you to pair a Bluetooth device. Your wireless mouse/keyboard should show up in the list. Select it, click "Pair." Once paired, navigation works. 5. The rest of Phase 1 below proceeds normally.

If you somehow get stuck mid-setup with no paired input device: - Power off the Mac mini (hold power button 10 seconds) - Power back on with a different wireless device in pairing mode - Setup Assistant re-tries the pairing prompt

You will NOT need a CD drive. macOS hasn't installed from CD since 2009. Everything's over the internet via Recovery Mode if anything goes wrong (Cmd+R at boot).

If you really want a wired-input backup option: USB-C to USB-A adapter (\$8 at any electronics store) plus your old laptop's USB keyboard/mouse if you have one. Not required.

Phase 1: Mac Mini One-Time Setup (45 minutes)

Step 1.1: Initial macOS setup

1. Power on the Mac mini, plug into Ethernet (preferred) or Wi-Fi.
2. Complete the macOS welcome flow. Important choices:
3. Sign in with the same Apple ID as your laptop (makes file transfer and Continuity easier).
4. Set the computer name: System Settings → General → Sharing → Local hostname. Name it **affirmology-mini** (the rest of this guide assumes this exact name).
5. Skip Touch ID and Siri if you want — neither is needed.
6. Plug in the Affirmology SSD.

Step 1.2: Enable Remote Login (SSH)

1. System Settings → General → Sharing.
2. Toggle **Remote Login** ON.
3. Click the next to it. Make sure "Allow full disk access for remote users" is on if you want remote backups (not required for our purposes).
4. Note the SSH access line at the top — should look like `jeffreyparker@affirmology-mini.local`. You'll use this from the laptop in Phase 2.

Step 1.3: Install Homebrew + core tools

Open Terminal on the Mac mini. Paste:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Follow the prompts (it asks for your password, takes ~5 minutes). When it finishes, it prints two lines about adding Homebrew to your shell — paste those into Terminal too.

Then install everything we need:

```
brew install python@3.11 git ffmpeg sqlite gh uv  
brew install --cask iterm2 visual-studio-code  
brew install --cask tailscale  
brew install --cask google-cloud-sdk  
brew install --cask claude  
brew install sshfs # for the laptop-mounts-Mac-mini direction
```

The `c laude` cask installs Claude Code on the Mac mini.

Step 1.4: Clone the affirmology-agent repo

```
mkdir -p ~/CLAUDE/AFFIRMولوجY
cd ~/CLAUDE/AFFIRMولوجY
```

Now you have two options:

Option A (preferred): if the code is already in GitHub:

```
git clone https://github.com/YOUR-USERNAME/affirmology-agent.git
```

Option B (works today, less ideal long-term): copy over the laptop's folder via the SSH-mount in Phase 2, then optionally push to GitHub later.

Either way, you should end up with `~/CLAUDE/AFFIRMولوجY/affirmology-agent/` populated.

Step 1.5: Set up Python venv on the Mac mini

```
cd ~/CLAUDE/AFFIRMولوجY/affirmology-agent
python3.11 -m venv .venv
source .venv/bin/activate
pip install --upgrade pip
pip install -e ".[dev]"
pip install trafilatura tenacity pypdf google-generativeai google-cloud-aiplatform youtube-transcript-api yt-dlp httpx weasyprint
```

This installs all the Affirmology dependencies plus the corpus build dependencies.

Step 1.6: Configure Tailscale on the Mac mini

1. Open the Tailscale app (it lives in the menu bar after install).
2. Click "Sign In" → use your Google or GitHub account. (Use the same identity you'll use on the laptop in Phase 2.)
3. Once connected, the Tailscale icon shows green. The Mac mini now has a private IP like `100.x.y.z` and a hostname like `affirmology-mini.tailfee.ts.net`.

Step 1.7: Set the Mac mini to never sleep

System Settings → Battery (or Energy Saver): - "Prevent automatic sleeping when the display is off" → ON - "Wake for network access" → ON - "Start up automatically after a power failure" → ON

System Settings → General → Software Update → Automatic Updates → click `-` - Turn OFF "Install macOS Updates" (auto-restarts kill long jobs) - Leave the others on if you want, off if you want maximum stability

Step 1.8: Set up launchd for the nightly scraper

Create the launchd plist that runs the scraper every night between 11pm and 9am:

```

mkdir -p ~/Library/LaunchAgents
cat > ~/Library/LaunchAgents/com.affirmology.nightly.plist <<'EOF'
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/
PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>Label</key><string>com.affirmology.nightly</string>
  <key>ProgramArguments</key>
  <array>
    <string>/bin/bash</string>
    <string>-c</string>
    <string>cd ~/CLAUDE/AFFIRMولوجY/affirmology-agent &&&
source .venv/bin/activate &&& caffeinate -i -m -s env PYTHONPATH=src
python -m affirmology.corpus.run --data-dir /Volumes/Affirmology/corpus --
traditions all --mode scrape --max-sources-per-tradition 200 --max-priority 5 --
per-source-timeout-seconds 3600 --stop-after-seconds 36000 --max-cost-usd 25 > /
Volumes/Affirmology/corpus/logs/nightly_$(date +%Y-%m-%d).log 2>&1</string>
  </array>
  <key>StartCalendarInterval</key>
  <dict>
    <key>Hour</key><integer>23</integer>
    <key>Minute</key><integer>0</integer>
  </dict>
  <key>RunAtLoad</key><false/>
  <key>StandardOutPath</key><string>/Volumes/Affirmology/corpus/logs/launchd-
stdout.log</string>
  <key>StandardErrorPath</key><string>/Volumes/Affirmology/corpus/logs/
launchd-stderr.log</string>
</dict>
</plist>
EOF

launchctl load ~/Library/LaunchAgents/com.affirmology.nightly.plist
launchctl list | grep affirmology

```

The last command should print one line confirming the job is registered. It will fire every night at 11pm and run for up to 10 hours.

Phase 2: Laptop ↔ Mac Mini Sync (30 minutes)

Step 2.1: Install Tailscale on the laptop

```
brew install --cask tailscale
```

Open the Tailscale app, sign in with the same account. Both machines should now appear in the Tailscale admin console (<https://login.tailscale.com/admin/machines>) with green dots next to them.

Step 2.2: Test SSH from the laptop to the Mac mini

```
ssh jeffreyparker@affirmology-mini
```

(The `.local` suffix isn't needed when both machines are on the same Tailscale network — Tailscale resolves the hostname automatically.)

Type `yes` on the first connection to accept the host key. Enter your Mac mini password. You should land in the Mac mini's shell. Confirm:

```
hostname  
exit
```

`hostname` should print `affirmology-mini`. Then `exit` brings you back to your laptop.

Step 2.3: Set up SSH key auth (no password every time)

On your laptop:

```
# Generate a key if you don't have one  
[ -f ~/.ssh/id_ed25519 ] || ssh-keygen -t ed25519 -N "" -f ~/.ssh/id_ed25519  
ssh-copy-id jeffreyparker@affirmology-mini
```

Now `ssh jeffreyparker@affirmology-mini` logs in without a password.

Step 2.4: Install macFUSE + SSHFS for the file mount

The cleanest way to make the Mac mini's files appear as a local drive on your laptop. Skip this if you'd rather copy files over manually via `scp` (works fine, just less convenient).

```
brew install --cask macfuse  
brew install gromgit/fuse/sshfs-mac
```

After install, restart your laptop once. macFUSE requires a kernel extension that needs a reboot to activate.

Step 2.5: Create the mount point on your laptop

```
mkdir -p ~/affirmology-mini-mount
```

Then mount the Mac mini's home folder so it appears under that path on your laptop:

```
sshfs jeffreyparker@affirmology-mini:/Users/jeffreyparker ~/affirmology-mini-  
mount -o defer_permissions,reconnect,allow_symlinks,allow_other
```

Now `~/affirmology-mini-mount/` on your laptop is the Mac mini's home directory. You can browse it in Finder, open files in any editor, drag files in and out, everything.

To unmount cleanly when you're done:

```
umount ~/affirmology-mini-mount
```

Step 2.6: Make the mount automatic at laptop startup (optional)

Create a launchd plist on your **laptop**:

```
cat > ~/Library/LaunchAgents/com.affirmology.mount.plist <<'EOF'  
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/  
PropertyList-1.0.dtd">  
<plist version="1.0">  
<dict>  
  <key>Label</key><string>com.affirmology.mount</string>  
  <key>ProgramArguments</key>  
  <array>  
    <string>/bin/bash</string>  
    <string>-c</string>  
    <string>/usr/local/bin/sshfs jeffreyparker@affirmology-mini:/Users/  
jeffreyparker /Users/jeffreyparker/affirmology-mini-mount -o  
defer_permissions,reconnect,follow_symlinks,allow_other</string>  
  </array>  
  <key>RunAtLoad</key><true/>  
  <key>KeepAlive</key><true/>  
</dict>  
</plist>  
EOF  
launchctl load ~/Library/LaunchAgents/com.affirmology.mount.plist
```

Now every time you boot the laptop and Tailscale connects, the Mac mini's files appear at `~/affirmology-mini-mount/` automatically.

Phase 3: Using This From Day to Day (15 minutes to learn)

Working pattern 1: Claude Desktop on the laptop, reading Mac mini files

You can open the mounted folder in Claude Desktop just like any local folder. Cowork sees `~/affirmology-mini-mount/CLAUDE/AFFIRMولوجY/affirmology-agent/` as a regular local path. You can read, write, and run any file tools against the Mac mini's content without leaving your laptop.

When you ask Claude to run something on the Mac mini, the cleanest pattern is:

1. From the laptop, open Terminal
2. `ssh jeffreyparker@affirmology-mini`
3. Run the command there
4. `exit` when done

Or, for a one-off command:

```
ssh jeffreyparker@affirmology-mini "cd ~/CLAUDE/AFFIRMOMETRY/affirmology-agent &&
source .venv/bin/activate && PYTHONPATH=src python -m affirmology.corpus.status
--data-dir /Volumes/Affirmology/corpus"
```

That runs the status check on the Mac mini and prints the result on your laptop, all without leaving your laptop's terminal.

Working pattern 2: Check on the nightly run from anywhere

From any terminal anywhere (your phone with Termius, a laptop in a hotel), as long as Tailscale is connected:

```
ssh jeffreyparker@affirmology-mini "cat /Volumes/Affirmology/corpus/logs/
heartbeat.json"
```

Shows what the nightly scraper is doing right now.

Working pattern 3: Edit Mac mini files in VS Code on the laptop

Install the **Remote - SSH** extension in VS Code. Then:

1. Cmd+Shift+P → "Remote-SSH: Connect to Host"
2. Type `jeffreyparker@affirmology-mini`
3. VS Code opens a new window editing the Mac mini's files directly. Terminal in this VS Code window is also remote. You can run Claude Code from this terminal.

This is the most powerful pattern for actual development. Files live on the Mac mini, editing happens on the laptop, code runs on the Mac mini.

Working pattern 4: Sync the laptop's code changes to the Mac mini

When you edit a file on your laptop and want it on the Mac mini quickly (without going through Git):

```
rsync -avz --delete \
~/CLAUDE/AFFIRMOMETRY/affirmology-agent/ \
jeffreyparker@affirmology-mini:~/CLAUDE/AFFIRMOMETRY/affirmology-agent/
```

That's a one-way push. For two-way, use Git as the source of truth and `git pull` on both machines.

Phase 4: Daily Routine

Each morning when you sit down with coffee:

```
ssh jeffreyparker@affirmology-mini "cd ~/CLAUDE/AFFIRMOMETRY/affirmology-agent &&
source .venv/bin/activate && PYTHONPATH=src python -m affirmology.corpus.status
--data-dir /Volumes/Affirmology/corpus"
```

That tells you what the night produced: how many new documents, how many new structured records, total spend, recent errors.

If anything went sideways:

```
# Kill any still-running scraper
ssh jeffreyparker@affirmology-mini "pkill -f affirmology.corpus.run"

# Check the most recent log
ssh jeffreyparker@affirmology-mini "ls -t /Volumes/Affirmology/corpus/logs/ |
head -3"

# Tail the most recent log
ssh jeffreyparker@affirmology-mini "tail -100 /Volumes/Affirmology/corpus/logs/
nightly_$(date +%Y-%m-%d).log"
```

Pause or unschedule the nightly job if you need to:

```
ssh jeffreyparker@affirmology-mini "launchctl unload ~/Library/LaunchAgents/
com.affirmology.nightly.plist"

# Re-enable later:
ssh jeffreyparker@affirmology-mini "launchctl load ~/Library/LaunchAgents/
com.affirmology.nightly.plist"
```

Common Issues

"sshfs: command not found" after install. macFUSE needs a reboot to activate the kernel extension. Restart the laptop and try again.

"ssh: Could not resolve hostname affirmology-mini". Tailscale isn't connected on one of the machines. Check the Tailscale menu bar icon on both — both should be green.

"Permission denied (publickey)". SSH key auth not set up. Re-run `ssh-copy-id jeffreyparker@affirmology-mini` from the laptop.

Mount disappears randomly. SSHFS sometimes loses connection. The `reconnect` flag in the mount command helps; if it still happens, `umount ~/affirmology-mini-mount` and re-mount.

The nightly job doesn't fire. Check `launchctl list | grep affirmology` shows the job. If it's missing, the plist didn't load — re-run the load command. Also check `~/Library/LaunchAgents/com.affirmology.nightly.plist` actually exists and has correct XML (no trailing whitespace issues from copy-paste).

What You Have After Setup

- A Mac mini that scrapes every night between 11pm and 9am, automatically, no input from you
- A laptop that can read/write the Mac mini's files as if they were local
- Claude Desktop on the laptop can see and edit Mac mini files

- VS Code Remote-SSH for actual development
- Status checks from anywhere via SSH
- The SSD with all the corpus data lives on the Mac mini, accessible from the laptop seamlessly

When something breaks: SSH in, fix it, exit. When something works: it keeps working without your attention.

End of guide. If anything in here doesn't apply to your exact setup, tell me and I'll write a corrected version.