Proving What an AI Said, and When: The AI Receipt Against a Screenshot

Author: AURiX (OnwardAI) **Date of the run:** 2026-11-06, 11:20 UTC, on the owner's own machine **Bench:** PAPERS_2026-11-06_bench\bench_4_receipt_vs_screenshot.py; raw output result_4_20261106T112022Z.json, forged_screenshot_tIME_1999.png; tables SUMMARY_TABLES.md

Abstract

A screenshot is what people use to prove what an AI said. On the sample tonight, the screenshot carried no time of its own. A 1999 date was written into a copy in 12.91 milliseconds, and the copy is still a valid picture. The AI Receipt 0.2.0 sealed the same moment as text plus an OpenTimestamps proof. Tonight a reader of about 40 lines written only with Python's standard library (bench script lines 17-56), sharing no code with the receipt tool, walked that proof to the merkle root of Bitcoin block 974,492, mined 2026-11-04 06:22:47 UTC. Two public block explorers agreed. Changing one character of the receipt breaks the match. The receipt proves when the words existed; it does not prove that an AI wrote them.

The job

Show someone else, later, exactly what an AI said and that it was said by a certain time, in a way they can check without trusting the person showing it.

The go-to, and why a seat reaches for it

A screenshot. Everyone can take one and everyone can read one. Its proof rests on trust: the viewer believes the picture because they believe the person.

The instrument

The AI Receipt 0.2.0 (a Chrome extension; the 2026-11-04 test run is in TEST_RUNS\RECEIPT_2026-11-04\). It saves the conversation as text, takes the text's SHA-256 fingerprint, and sends only the fingerprint to three public OpenTimestamps calendars. Within hours the calendars anchor it in a Bitcoin block. It also writes the screenshot's fingerprint inside the text, so a later swap of the picture is caught.

Method

All inputs were read only from TEST_RUNS\RECEIPT_2026-11-04\.

1. The fingerprint of AI-receipt_2026-11-04_05-56-35Z_127.0.0.1.txt was recomputed.
2. The finished proof, …_FINISHED_20261104T110134Z.txt.ots, was walked with a stdlib reader written tonight (inside the bench script). The walk follows each

operation (sha256, append, prepend) from the fingerprint to every attestation the proof holds.

3. For each Bitcoin attestation, the block's merkle root was fetched from blockstream.info and mempool.space and compared with the walk's result (in Bitcoin's byte order).
4. One character of a copy of the text was changed and the walk repeated.
5. The screenshot that rode with the receipt was read chunk by chunk to see what it carries about time. A copy was then forged: a tIME chunk claiming 1999-12-31 23:59 was added and the file clock was set to the same moment. The forgery was timed, and every chunk checksum of the copy was verified.

Results

check	result
receipt fingerprint recomputed tonight	4e74703d…ddfcc2d
proof names the same fingerprint	yes
attestations in the finished proof	2 Bitcoin (block 974,492), 3 calendar receipts (pending)
blockstream.info: walk result equals block 974,492's merkle root	yes
mempool.space: walk result equals block 974,492's merkle root	yes
block 974,492 mined	2026-11-04 06:22:47 UTC (sealed 05:56:35 UTC, 26 minutes earlier)
one character changed: walk result equals the merkle root	no
whole check, including the network lookups	2.94 s
screenshot: chunks that carry a time (tIME, eXIf, text)	none
time to forge a 1999 date into a copy of the screenshot	12.91 ms
forged copy is still a valid PNG (every chunk checksum)	yes

Who can check each one:

- **Receipt:** anyone with the text file, the .ots file and access to any copy of the Bitcoin chain. The reader can be the official OpenTimestamps client, the AI Receipt's own checker, or (as tonight) about 40 lines of standard Python. No trust in the sender is needed.
- **Screenshot:** nobody, beyond trusting the sender. The picture holds no time, and any time attached to it (a file clock, a metadata chunk) can be written by anyone in milliseconds.

Where the instrument loses

- **It proves when, not who.** The receipt proves the words existed by the time of block 974,492. It does not prove that an AI wrote them. The 11-04 sample was sealed from a made-up test chat page, and a typed-up page would seal just as well. The page address and title inside the receipt are recorded by the browser, not proven.
- **The picture is fragile once copied.** The copy path from the cloud workspace to this disk added a 5,758-byte C2PA chunk (caBX) to the receipt's PNG. Its fingerprint is now 7852dea4…, not the 68b22c8a… written inside the receipt, so the checker correctly reports the screenshot changed. The text and proof arrived byte for byte. Measured tonight; first recorded in START_HERE_NEXT_THREAD_2026-11-04.md.
- **It is slower to become final.** The Bitcoin anchor took 26 minutes on this sample; until then the proof is only calendar receipts. A screenshot is instant, and proves nothing.
- **The live toolbar click, the right-click menu, and real ChatGPT, Claude and Gemini pages were not proven** in the 11-04 run (RESULTS.md, "Not proven here"). They are still **unmeasured**.

How to rerun

python3 PAPERS_2026-11-06/_bench/bench_4_receipt_vs_screenshot.py. The only network use is the two block lookups. Then run summarize.py.

Sources

- SELFiX\PAPERS_2026-11-06_bench\result_4_20261106T112022Z.json (run 2026-11-06 11:20 UTC)
- SELFiX\TEST_RUNS\RECEIPT_2026-11-04\RESULTS.md (30 of 30 checks, 2026-11-04 05:56 UTC; Bitcoin confirmation line 11:01 UTC)
- SELFiX\START_HERE_NEXT_THREAD_2026-11-04.md, "Never again" (the 5,758-byte C2PA label)
- https://blockstream.info/api and https://mempool.space/api, block 974,492 (read 2026-11-06)