

GetProofAnchor - Evidence Report

This document provides a concise, human-readable summary of the digital evidence package.

Verification status: VERIFIED ✓ (eIDAS) · Bitcoin content anchor attested

Evidence Identification	—
Proof ID	73782224-b5a4-4cfb-b7cf-a82879ed1ff7
Captured at (UTC)	2026-10-31T10:26:36Z
Source URL	https://getproofanchor.com/
Final URL	https://getproofanchor.com/
Page title	GetProofAnchor — Forensic Web Evidence Capture (eIDAS)

Verification summary:

- ✓ Sealed artifacts can be independently verified against the qualified timestamp
- ✓ This package contains: screenshot, page HTML, offline MHTML snapshot, HTTP archive (HAR), capture video, TLS certificate evidence and integrity metadata
- ✓ Bitcoin content anchor (anchor v3 inclusion proof): attested (per OTS calendar attestation, block 973966; not independently verified)
- ✓ eIDAS qualified timestamp status: ok

This evidence is designed to be independently verifiable rather than merely trusted. This report, proof.json and manifest.json are generated when the package is downloaded and are not sealed. The sealed record is timestamp/eidas_payload.json with timestamp/eidas.tsr and the files whose digests it lists. Compare this report with the output of the independent verifier (see README.md).

What this package proves

This package proves the existence and integrity of the captured data. It demonstrates that the captured content existed in the recorded form at the recorded time, and that any later modification of a sealed artifact would be detectable during verification.

How integrity is protected

Each captured artifact has its own SHA-256 cryptographic hash. The qualified timestamp seals a payload that lists these hashes together with the source URL, final URL and capture time. The package also includes an append-only chain record and, where present, a Bitcoin anchor. As a result, any modification to a sealed artifact produces a mismatch that can be detected. This report, proof.json and manifest.json are generated at download time and are not sealed; the page title and publication status shown here are not part of the sealed record.

Cryptographic Identifiers	—
Fingerprint (SHA-256)	f2f61ccc0d586a6d200021618aa24ffcb23d7402877c568458098d72d9bdd552
Content SHA-256	119552abcbe2981dd03d58dde349a7ab8d1fe27c4c4306c32f450dbb8710c7a6
Page HTML SHA-256	0e32dc3002618d76652f2d88a8b5f087386f7b55412c1c155c080f6126758e8c

Cryptographic Identifiers	—
Screenshot SHA-256	efce2aaa2c1fc7bffc93137ef743b9406b1c70a5a04ee22b288f2f279a4a1e23
Chain sequence	562
Chain head hash	13ef7a89adfb7943a39e8d93ec28b483105c79f835f4bb409c4f110a5ab2e95f

Blockchain anchoring

What is anchored (anchor v3). This proof has a Merkle inclusion proof in anchor/inclusion.json. Its leaf is SHA-256 over the proof ID and the SHA-256 of timestamp/eidas_payload.json — the exact payload the qualified timestamp signs, which lists the digest of every sealed artifact. The content anchor's receipt is anchor/anchor_receipt.ots; its status is attested (per OTS calendar attestation, block 973966; not independently verified). The leaf is number 0 of 1 in an RFC 6962 Merkle tree whose root (5345e9dd212f4dd5f6cace006ddaf7ab7cad93b682f880d3b0e9356eb307eacc) is part of the anchor payload committed by the OpenTimestamps receipt. Offline check: recompute the leaf from timestamp/eidas_payload.json, follow the audit path to the root, compare the root with the anchor payload, check that SHA-256 of the canonical anchor payload equals the digest in the .ots receipt, then run ots verify with a Bitcoin node (README.md, Step 3). The Bitcoin anchor proves existence no later than the anchoring block, which may be later than the capture time; the qualified electronic timestamp remains the legal time evidence.

Content-binding anchor (v3)	—
Status	attested (per OTS calendar attestation, block 973966; not independently verified)
Anchor payload (SHA-256)	eb0aee4789de2ceb3b12d542fb2d7a48aee666f758f647bc85eeaeafd834b88f
Bitcoin block	973966
Transaction ID	518e951a0bb6fbc9624e4726dfa641ef6cd106249d6847b5f218f41e00a543f
Anchor created at	2026-10-31T10:27:02Z
Attestation recorded by server at	2026-10-31T11:47:59Z
Receipt file	anchor/anchor_receipt.ots
Inclusion proof	leaf 0 of 1 (anchor/inclusion.json)

Qualified timestamp

This package may also include a qualified electronic timestamp issued by an independent trust service provider.

eIDAS Qualified Timestamp	—
Status	ok
Timestamp time	2026-10-31T10:26:38Z
Provider / TSA	SK ID Solutions AS

Capture mode — Assisted (consent resolution)

This capture was performed in assisted mode. The capture engine automatically detected and resolved a consent dialog in order to access the public content of the page. The resolution method, click target, and outcome are recorded below for full transparency. No authentication credentials were used.

Assisted Capture Details	—
Capture profile	assisted
Verdict	success_assisted
AI vision calls	0
TCF stub injected	No
Consent dialogs resolved	1

Consent Resolution #1	—
Method	js_strong
Button clicked	accept all
URL before	https://getproofanchor.com/
URL after	https://getproofanchor.com/
Dialog disappeared	Yes
Page content loaded	No

Note: Assisted capture interacts only with publicly accessible consent mechanisms. The capture engine interacted with the consent dialog as logged above. The screenshot is a full-page render produced by the capture engine. It may differ from what a visitor saw on screen — for example, fixed or sticky overlays may have been hidden or de-duplicated while the page was stitched into a single image. The authoritative record of the page is the set of hashed artifacts in this package (page HTML, MHTML snapshot, HAR and video, where present); the interventions performed are listed in `capture/capture_meta.json`.

Capture interventions

The capture engine recorded the following interventions in `capture/capture_meta.json` (`interaction_summary`). They affect the screenshot; the hashed page HTML, MHTML, HAR and video remain the authoritative record of the page.

- Consent dialog: 1 click(s) performed by the capture engine (details in the consent resolution tables / `consent_log`).
- A stylesheet hiding consent/cookie dialogs was injected into the page.
- Overlay-removal pass (assisted profile): fixed-position modal or overlay elements were hidden immediately before the screenshot. Individual elements are not itemised in this version.

Independent verification

This evidence can be verified without relying on GetProofAnchor. The package includes `README.md`, which explains how to verify file hashes, chain integrity, blockchain anchoring, and, where present, qualified timestamping artifacts.

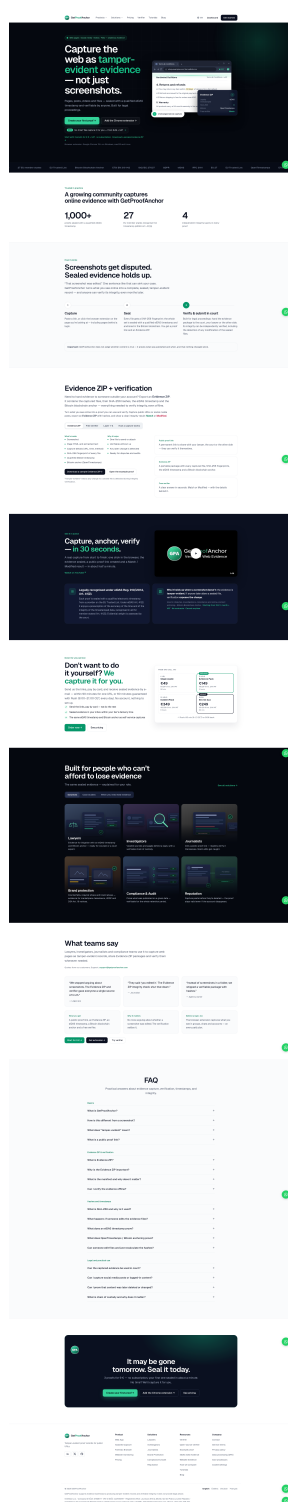
Important note

This system proves the existence and integrity of data. It does not prove the factual truthfulness of the captured content itself. Its purpose is to provide a technically verifiable foundation for working with digital evidence.

Key principle: the system is designed to be verified, not merely trusted.

Visual Capture (Screenshot)

The image below shows the captured web page at the time of evidence creation. The screenshot is a full-page render produced by the capture engine. It may differ from what a visitor saw on screen — for example, fixed or sticky overlays may have been hidden or de-duplicated while the page was stitched into a single image. The authoritative record of the page is the set of hashed artifacts in this package (page HTML, MHTML snapshot, HAR and video, where present); the interventions performed are listed in capture/capture_meta.json.



Captured at (UTC): 2026-10-31 10:26:36

Network-layer Forensic Evidence

Network-layer evidence recorded during this acquisition session, comprising server attribution via the TLS certificate presented at acquisition, multi-resolver DNS resolution, RDAP registry snapshot, a full HTTP transaction trace (HAR). The acquisition procedure was designed with reference to the identification and collection guidance of ISO/IEC 27037 § 6.4–6.5 and the network-trace guidance of SWGDE 21-F-001, to the extent of the artifacts listed on this page; this is not a certification of compliance.

TLS / SSL Certificate	—
Captured	Yes
Subject	CN=*.getproofanchor.com
Issuer	CN=YE1,O=Let's Encrypt,C=US
Serial	0x55b54bd923ee02303b410c2eacf961df50e
Not before (UTC)	2026-09-29T00:27:10Z
Not after (UTC)	2026-12-28T00:27:09Z
SHA-256 fingerprint	32443c29dfbce1ba9b6b4608278ee9052542d96e21defc2fee4f3ec44e40b616
TLS version	TLSv1.3
Cipher	TLS_AES_256_GCM_SHA384
Server IP	2a01:28:ca:112::1:3185
Reverse DNS	put06.vas-server.cz
Hostname matches certificate	Yes ✓
Chain length	4

DNS Resolution Snapshot	—
Hostname	getproofanchor.com
Captured at (UTC)	2026-10-31T10:25:35Z
Resolvers used	1.1.1.1, 8.8.8.8, 9.9.9.9
A (consensus)	46.234.101.84
AAAA (consensus)	2a01:28:ca:112::1:3185
NS (consensus)	ns.vas-hosting.com., ns.vas-hosting.cz., ns.vas-hosting.eu.
MX (consensus)	10 put06.vas-server.cz.
TXT (consensus)	"google-site-verification=kVNi8IOBc6Vo8rNHc6azJ0pKxN795KWI4q6or-FqRyo", "v=spf1 a mx ip4:46.234.101.84 ip6:2a01:28:ca:112::1:3185 -all"
Resolver divergence	No (all resolvers agreed)

RDAP / WHOIS Snapshot	—
Domain	getproofanchor.com
Event: registration	2025-11-08T21:01:03Z
Event: expiration	2027-11-08T21:01:03Z
Event: last changed	2026-10-07T07:44:06Z
Event: last update of RDAP database	2026-10-31T10:25:29Z
Status	active

Media Artifacts	—
HAR (HTTP archive)	captured
HAR SHA-256	a3c9d0ed4fa9edd91528fa00c16b950276662641341f485158d1337ad6b94a03
Video (capture session)	captured
Video SHA-256	adab92a79959b634881eec857e92824db8c59c7ec462251b9ea9e9b890ca1a3a
TLS leaf cert PEM	captured
Leaf SHA-256	e386249c7508eaff0dab0808626f806f89edd925d839a51b3aa23490dcfb49ae
TLS chain PEM	captured
Chain SHA-256	075ba300be4f1af57f5b29346dce29665a6f297fd3d5dd4e5cace273c1cc631b

How this is sealed. The following artifact digests are bound directly into the eIDAS timestamp payload (format stamp-6): content, capture_meta, screenshot, page HTML, MHTML snapshot, HAR, video, TLS leaf PEM, TLS chain PEM. The DNS, RDAP and TLS records shown on this page are carried inside capture/capture_meta.json under interaction_summary.network_evidence, and are therefore bound through capture_meta_sha256; the copies under network/ are extracts of that sealed data and can be compared against it field by field. Any change to a bound artifact produces a digest mismatch and invalidates verification.

Evidence Integrity Certificate

This certificate confirms that the captured web evidence has been processed and verified using cryptographic methods.

Certificate Identification	—
Proof ID	73782224-b5a4-4cfb-b7cf-a82879ed1ff7
Captured at (UTC)	2026-10-31T10:26:36Z
Source URL	https://getproofanchor.com/
Final URL	https://getproofanchor.com/

Integrity checks:

- ✓ Content hash verified
- ✓ Page HTML hash verified
- ✓ Screenshot hash verified
- ✓ Offline MHTML snapshot hash verified
- ✓ Chain metadata included
- ✓ Bitcoin content anchor status recorded: attested (per OTS calendar attestation, block 973966; not independently verified)
- ✓ eIDAS timestamp status recorded: ok
- ✓ HAR network trace recorded
- ✓ Capture video recorded
- ✓ TLS server certificate captured
- ✓ TLS full chain captured
- ✓ DNS resolution snapshot recorded
- ✓ RDAP registry snapshot recorded

Status: VERIFIED ✓ (eIDAS) · Bitcoin content anchor attested

Cryptographic guarantee

The integrity of this evidence is protected using SHA-256 cryptographic hashing and independent timestamping mechanisms. Any modification to the recorded digests, the acquisition metadata, the page HTML, the screenshot, the MHTML snapshot or the TLS certificate evidence produces a different digest and invalidates the integrity checks.

This report, proof.json and manifest.json are generated when the package is downloaded and are not sealed. The sealed record is timestamp/eidas_payload.json with timestamp/eidas.tsr and the files whose digests it lists. Compare this report with the output of the independent verifier (see README.md).

GetProofAnchor
Digital Evidence Integrity System
<https://getproofanchor.com>