

Wie eine Beobachtung entstanden ist

MUSTER. Geschrieben für eine echte Beobachtung unserer eigenen Domain machinewitness.eu, mit ihren tatsächlichen Parametern, mit demselben Werkzeug wie jede Verfahrensbeschreibung. Sie ist als Beispiel veröffentlicht und deshalb nicht unterschrieben. Eine bestellte Verfahrensbeschreibung wird für die Beobachtung geschrieben, die Sie nennen, gelesen und unterschrieben.

1. Gegenstand dieser Beschreibung

Diese Beschreibung legt dar, wie eine Beobachtung entstanden ist: die Adresse `https://machinewitness.eu/robots.txt` am Tag **2026-09-28** (UTC), unabhängig abgerufen von 2 Zeugen. Sie erklärt das Verfahren mit den tatsächlichen Parametern dieser Beobachtung, legt jeden Schritt so dar, dass ein Dritter ihn ohne uns wiederholen kann, und nennt die Einwände, die gegen das Verfahren üblicherweise erhoben werden. Zum abgerufenen Inhalt und zu einer Sache, in der er verwendet wird, sagt sie nichts.

Die Parameter unten sind aus den Paketen der Zeugen gelesen und, wo möglich, nachgerechnet; nichts davon wurde von Hand eingetippt.

WITNESS-1

Zeuge	witness-1
Abgerufen um (UTC)	2026-09-28T03:12:53+00:00
Anfrage	GET, keine Cookies, Weiterleitungen gefolgt, HTTP/1.1
Endgültige Adresse	<code>https://machinewitness.eu/robots.txt</code>
HTTP-Status	200
TLS	TLSv1.3 TLS_AES_256_GCM_SHA384
Zertifikat, Inhaber	CN=machinewitness.eu
Zertifikat, Aussteller	C=US, O=Google Trust Services, CN=WE1
Zertifikat gültig	Sep 20 12:45:20 2026 GMT bis Dec 19 13:45:17 2026 GMT
Zertifikatskette (SHA-256 je DER)	aec098364a0137418a01ed8935ac9d0070ffb370429a16df8b70a8cd73cd5e53 · 1dfc1605fbad358d8bc844f76d15203fac9ca5c1a79fd4857ffaf2864fbeb96 · 76b27b80a58027dc3cf1da68dac17010ed93997d0b603e2fadbe85012493b5a7
Vollständigkeit	Vollständig gespeichert; für diese Beobachtung galt keine Byte-Grenze.
SHA-256 der ausgelieferten Bytes	ae6ef8da7be4397f9d37a57be7b522f3f2b84d2b6af133ba26e8379acab36e32
SHA-256 der Herkunftsdaten	5819ce2ac282b3651477abf7db916cf1f3afd222a8aa3a0abef9fa9c580dc9c6
Blatt	c6ae3444c00b163a044151ea59f634a6b0b39a4a3e29ca5b57f61abab78ec1ca
Position im Baum des Tages	Index 44,025 von 642,761 Blättern, 20 Schritte zur Wurzel
Wurzel des Tages	f1437269e4708b3dcc32a0c2947bdec58c1dbe084739ff912f14a03cacad48a6
Anker der Tageswurzel	RFC-3161-Zeitstempel (FreeTSA) · OpenTimestamps (Bitcoin) · qualifizierte r eIDAS-Zeitstempel (GLOBALTRUST)
Paket	machinewitness-extract-witness-1-2026-09-28-d0199b52c7cc.zip, SHA-256 3188c88a114c739d32122f7a89c6f4fa06c96c70eb1105886a2c4710c5633fb7

WITNESS-2

Zeuge	witness-2
Abgerufen um (UTC)	2026-09-28T03:01:20+00:00
Anfrage	GET, keine Cookies, Weiterleitungen gefolgt, HTTP/1.1

Endgültige Adresse	https://machinewitness.eu/robots.txt
HTTP-Status	200
TLS	TLSv1.3 TLS_AES_256_GCM_SHA384
Zertifikat, Inhaber	CN=machinewitness.eu
Zertifikat, Aussteller	C=US, O=Google Trust Services, CN=WE1
Zertifikat gültig	Sep 20 12:45:20 2026 GMT bis Dec 19 13:45:17 2026 GMT
Zertifikatskette (SHA-256 je DER)	aec098364a0137418a01ed8935ac9d0070ffb370429a16df8b70a8cd73cd5e53 · 1dfc1605fbad358d8bc844f76d15203fac9ca5c1a79fd4857ffaf2864fbeb96 · 76b27b80a58027dc3cf1da68dac17010ed93997d0b603e2fadbe85012493b5a7
Vollständigkeit	Vollständig gespeichert; für diese Beobachtung galt keine Byte-Grenze.
SHA-256 der ausgelieferten Bytes	ae6ef8da7be4397f9d37a57be7b522f3f2b84d2b6af133ba26e8379acab36e32
SHA-256 der Herkunftsdaten	a20e01fd655b9927a077d667fbcf68afe2a2600552863b6ac8ae84297974a730
Blatt	10f22b28b520a9ea35e4e1be81f525f7aa04035157b9d1cb22ced67375f3fd52
Position im Baum des Tages	Index 24 von 586,787 Blättern, 20 Schritte zur Wurzel
Wurzel des Tages	d568805980fc63d4fbd54959ec16efda605b13fc1cae54b88fb94a7ebdc55d
Anker der Tageswurzel	RFC-3161-Zeitstempel (FreeTSA) · OpenTimestamps (Bitcoin) · qualifizierte r eIDAS-Zeitstempel (GLOBALTRUST)
Paket	machinewitness-extract-witness-2-2026-09-28-d0199b52c7cc.zip, SHA-256 d69cce6fbdda19d2a951419677ace17a25d2e079ff7799e2fea49ef21e5ed5f6

Beide Zeugen haben dieselben Bytes erhalten: der SHA-256 der ausgelieferten Bytes ist in beiden Paketen gleich. Ihre Blätter unterscheiden sich, wie sie es müssen: jedes Blatt bindet auch die eigene Abrufzeit des Zeugen und seine eigenen Herkunftsdaten.

2. Abruf

Jeder Zeuge ist eine eigene Maschine bei einem eigenen Anbieter in einem eigenen Land, mit eigenen Schlüsseln. Keiner kann in den anderen schreiben. Jeder ruft die Adresse mit einer einfachen HTTP-GET-Anfrage ab, weist sich mit dem veröffentlichten User-Agent MachineWitnessBot/0.1 (+https://machinewitness.eu/bot; evidence archive; contact: bot@machinewitness.eu) aus, sendet keine Cookies und folgt Weiterleitungen; die Adresse, die zuletzt geantwortet hat, wird als endgültige Adresse festgehalten. Der Zeuge zeichnet den Status auf, alle Antwort-Header, TLS-Version und Verschlüsselung, das Serverzertifikat und den SHA-256 jedes Zertifikats der vorgelegten Kette sowie, wo erfasst, seine eigene Adresse und die Adresse, die geantwortet hat. Er meldet sich nicht an, klickt nicht, füllt nichts aus und führt die Skripte der Seite nicht aus. Aufgezeichnet ist also, was diese Adresse dieser Anfrage in diesem Moment ausgeliefert hat, nicht, was sie einem anderen Besucher zu einem anderen Zeitpunkt gezeigt haben mag.

3. Speicherung und Fingerabdruck

Die ausgelieferten Bytes werden so gespeichert, wie sie empfangen wurden. Ihr SHA-256-Fingerabdruck wird über genau diese Bytes gebildet. Ob die gespeicherten Bytes vollständig sind, ist selbst in den Herkunftsdaten festgehalten (truncated), und die Zeile zur Vollständigkeit oben gibt es wieder. Die Herkunftsdaten (endgültige Adresse, Header, TLS-Daten, Zertifikats-Hashes, Adressen) werden als kompaktes JSON mit sortierten Schlüsseln serialisiert und ihrerseits mit einem Fingerabdruck versehen.

4. Das Blatt

Die Beobachtung wird auf einen einzigen Wert zurückgeführt, das Blatt:

```
leaf = SHA-256( 0x00 || "mwleaf-v1|" url "|" observed_at "|" payload_sha256 "|" status "|"
provenance_sha256 )
```

Das vorangestellte Byte 0x00 trennt Blätter von inneren Knoten des Baums, sodass kein innerer Knoten als Blatt ausgegeben werden kann. Für diese Beschreibung wurde das Blatt jedes Zeugen aus seinen aufgezeichneten Feldern nachgerechnet und stimmt mit dem Blatt im Paket überein.

5. Baum und Wurzel des Tages

Jede Beobachtung, die ein Zeuge an einem UTC-Tag macht, wird ein Blatt im Merkle-Baum dieses Tages. Innere Knoten sind SHA-256(0x01 || links || rechts). Der einzelne Wert an der Spitze, die Wurzel des Tages, ändert sich, sobald sich irgendein Byte irgendeiner Beobachtung dieses Tages ändert. Das Paket enthält den Pfad von diesem Blatt zur Wurzel: die Geschwister-Hashes und ihre Seite. Faltet man sie der Reihe nach, ergibt sich die Wurzel. Die Wurzel ist im öffentlichen Protokoll dieses Archivs unter einer festen Adresse für den Tag veröffentlicht.

6. Anker

Noch in derselben Nacht wird die Wurzel des Tages an Stellen außerhalb dieses Archivs übergeben: an einen RFC-3161-Zeitstempeldienst, an die OpenTimestamps-Kalender, die sie in die Bitcoin-Blockchain einbringen, und seit dem 31. Juli 2026 an GLOBALTRUST (e-commerce monitoring GmbH, Österreich), einen qualifizierten Vertrauensdiensteanbieter auf der EU-Vertrauensliste, der einen qualifizierten elektronischen Zeitstempel ausstellt. Jeder Anker belegt, dass die Wurzel zu dem Zeitpunkt existierte, den er nennt. Danach lässt sich eine Wurzel nicht mehr austauschen, ohne dass die Anker nicht mehr passen. Der qualifizierte Zeitstempel ist derjenige, der die Vermutung nach Artikel 41 Absatz 2 der Verordnung (EU) Nr. 910/2014 für die Richtigkeit seines Datums und seiner Zeit und für die Unversehrtheit der mit ihm verbundenen Daten genießt.

7. Zwei Zeugen

Der zweite Zeuge ruft selbst ab, bildet seinen eigenen Baum und nimmt seine eigenen Anker. Die beiden Wurzeln sind nicht identisch und sollen es nicht sein: jeder Zeuge beobachtet nach seinem eigenen Zeitplan und zeichnet seine eigene Abrufzeit auf. Was sie ermöglichen, ist ein Vergleich: für dieselbe Adresse am selben Tag zeigt das Paket jedes Zeugen, was dieser Zeuge erhalten hat, und beide lassen sich gegeneinander halten. Beide Zeugen haben dieselben Bytes erhalten: der SHA-256 der ausgelieferten Bytes ist in beiden Paketen gleich. Ihre Blätter unterscheiden sich, wie sie es müssen: jedes Blatt bindet auch die eigene Abrufzeit des Zeugen und seine eigenen Herkunftsdaten.

8. Die Schritte ohne uns wiederholen

1. **Bytes.** sha256sum payload-<id>.bin ergibt den Wert, der in leaf-<id>.json als payload_sha256 steht.
2. **Herkunftsdaten.** sha256sum provenance-<id>.json, die Datei genau so, wie sie ist, ergibt provenance_sha256.
3. **Blatt.** Die Formel aus Abschnitt 4 mit den fünf Feldern aus leaf-<id>.json ergibt leaf. Jedes Werkzeug, das SHA-256 berechnet, genügt.
4. **Wurzel.** Den Pfad in proof-<id>.json wie in Abschnitt 5 beschrieben falten; das Ergebnis ist die Wurzel in <Tag>.root.txt und im öffentlichen Protokoll.
5. **Anker.** openssl ts -verify für die beiden RFC-3161-Token gegen die Zertifikatsketten der Aussteller; ots verify für den OpenTimestamps-Nachweis.

Jedes Paket enthält diese Schritte in einer Datei VERIFY.txt. Keiner davon braucht Software dieses Archivs.

9. Übliche Einwände und die technische Antwort

Der Betreiber könnte die Aufzeichnung nachträglich geändert haben.

Eine Änderung an irgendeinem Byte ändert das Blatt, die Wurzel und die Anker des Tages. Die Anker wurden in der Nacht ausgestellt, in der der Tag versiegelt wurde, und liegen bei Dritten. Eine geänderte Aufzeichnung passt nicht mehr zu ihnen, und das lässt sich ohne uns prüfen (Abschnitt 8, Schritte 4 und 5).

Die Aufzeichnung zeigt, was der Zeuge speichern wollte, nicht, was der Server gesendet hat.

Die Bytes werden so gespeichert, wie sie empfangen wurden, und erhalten ihren Fingerabdruck, bevor irgendetwas anderes mit ihnen geschieht. Die Vollständigkeit ist als versiegeltes Feld festgehalten, nicht nachträglich behauptet. Wo eine Byte-Grenze griff, sagt die Aufzeichnung das für diese Beobachtung.

Der Server könnte anderen Besuchern etwas anderes gezeigt haben.

Das trifft zu und wird ausgesprochen, nicht verborgen: die Aufzeichnung erfasst, was diese Adresse dieser Anfrage, ausgewiesen durch den veröffentlichten User-Agent, im angegebenen Moment ausgeliefert hat. Über andere Anfragen sagt sie nichts.

Die Abrufzeit beruht auf der eigenen Uhr des Zeugen.

Die Abrufzeit zeichnet der Zeuge auf. Die Anker legen unabhängig davon den spätesten Zeitpunkt fest, zu dem der ganze Tag, diese Beobachtung eingeschlossen, existierte. Der qualifizierte Zeitstempel nennt seine Zeit unter der in Abschnitt 6 beschriebenen Aufsicht.

Eine einzelne Maschine könnte irren.

Es sind zwei, bei verschiedenen Anbietern, mit getrennten Schlüsseln, und keine kann in die andere schreiben (Abschnitt 7).

10. Was diese Beschreibung nicht ist

Sie enthält keine Aussage zu einer Sache, keine Bewertung des Verhaltens von irgendjemandem und keine Einschätzung der Aussichten eines Verfahrens. Sie beschreibt ein Verfahren und die Parameter einer Beobachtung. Was daraus folgt, entscheiden die Parteien und das Gericht.

Teil B · Schriftliche technische Antworten, Runde 1

Schriftlich eingegangene Fragen zur Beobachtung oben, schriftlich beantwortet. Die Antworten stützen sich auf das veröffentlichte Verfahren und sagen nichts zur Sache.

Der Auszug nennt für den 28. September 2026 zwei verschiedene Tageswurzeln, eine je Zeuge. Welche der beiden ist die richtige?

Beide. Jeder Zeuge bildet seinen eigenen Baum über seine eigenen Beobachtungen dieses Tages und verankert seine eigene Wurzel; die beiden Bäume enthalten unterschiedliche Mengen von Blättern, zu unterschiedlichen Zeiten aufgezeichnet, daher unterscheiden sich ihre Wurzeln schon ihrer Bauart nach. Eine Wurzel wird gegen das Protokoll und die Anker des Zeugen geprüft, der sie gebildet hat. Gemeinsam ist den beiden Zeugen nicht die Wurzel, sondern die Beobachtung: für diese Adresse tragen beide Pakete denselben SHA-256 der ausgelieferten Bytes.

Wie lässt sich prüfen, dass die robots.txt nach der Speicherung nicht verändert wurde?

Durch Nachrechnen, ohne uns: der SHA-256 von payload-<id>.bin muss dem Wert payload_sha256 in der Blattdatei entsprechen; das Blatt muss sich aus seinen fünf Feldern nachrechnen lassen (Abschnitt 4); der Pfad in proof-<id>.json muss sich zur Tageswurzel falten lassen; und die Tageswurzel muss zu den Ankern passen, die in derselben Nacht ausgestellt wurden (Abschnitt 8). Wäre nach dem 28. September 2026 auch nur ein Byte geändert worden, schlug eine dieser vier Prüfungen fehl.

Ort, Datum

Martin Schenk, für MachineWitness (Martin Schenk S.L., Madrid)

MUSTER, nicht unterschrieben.