

Cómo se realizó una observación

MUESTRA. Redactada para una observación real de nuestro propio dominio machinewitness.eu, con sus parámetros reales, con la misma herramienta que cada descripción. Se publica como ejemplo y por eso no está firmada. La descripción que usted encargue se redacta para la observación que indique, se revisa y se firma.

1. Objeto de esta descripción

Esta descripción expone cómo se realizó una observación: la dirección <https://machinewitness.eu/robots.txt> el día **2026-09-28** (UTC), obtenida de forma independiente por 2 testigos. Explica el método con los parámetros reales de esa observación, expone cada paso de modo que un tercero pueda repetirlo sin nosotros e indica las objeciones que suelen plantearse contra el método. No dice nada sobre el contenido obtenido ni sobre ningún asunto en el que se utilice.

Los parámetros de abajo se leen de los paquetes de los testigos y se recalculan cuando es posible; nada en ellos se ha tecleado a mano.

WITNESS-1

Testigo	witness-1
Obtenido a las (UTC)	2026-09-28T03:12:53+00:00
Solicitud	GET, sin cookies, redirecciones seguidas, HTTP/1.1
Dirección final	https://machinewitness.eu/robots.txt
Estado HTTP	200
TLS	TLSv1.3 TLS_AES_256_GCM_SHA384
Certificado, titular	CN=machinewitness.eu
Certificado, emisor	C=US, O=Google Trust Services, CN=WE1
Certificado válido	Sep 20 12:45:20 2026 GMT hasta Dec 19 13:45:17 2026 GMT
Cadena de certificados (SHA-256 de cada DER)	aec098364a0137418a01ed8935ac9d0070ffb370429a16df8b70a8cd73cd5e53 · 1dfc1605fbad358d8bc844f76d15203fac9ca5c1a79fd4857ffaf2864fbeb96 · 76b27b80a58027dc3cf1da68dac17010ed93997d0b603e2fadbe85012493b5a7
Integridad del contenido	Almacenado íntegramente; no se aplicó ningún límite de bytes a esta observación.
SHA-256 de los bytes entregados	ae6ef8da7be4397f9d37a57be7b522f3f2b84d2b6af133ba26e8379acab36e32
SHA-256 de la procedencia	5819ce2ac282b3651477abf7db916cf1f3afd22a8aa3a0abef9fa9c580dc9c6
Hoja	c6ae3444c00b163a044151ea59f634a6b0b39a4a3e29ca5b57f61abab78ec1ca
Posición en el árbol del día	índice 44,025 de 642,761 hojas, 20 pasos hasta la raíz
Raíz del día	f1437269e4708b3dcc32a0c2947bdec58c1dbe084739ff912f14a03cacad48a6
Anclajes de la raíz del día	sello de tiempo RFC 3161 (FreeTSA) · OpenTimestamps (Bitcoin) · sello cualificado de tiempo eIDAS (GLOBALTRUST)
Paquete	machinewitness-extract-witness-1-2026-09-28-d0199b52c7cc.zip, SHA-256 3188c88a114c739d32122f7a89c6f4fa06c96c70eb1105886a2c4710c5633fb7

WITNESS-2

Testigo	witness-2
Obtenido a las (UTC)	2026-09-28T03:01:20+00:00
Solicitud	GET, sin cookies, redirecciones seguidas, HTTP/1.1
Dirección final	https://machinewitness.eu/robots.txt

Estado HTTP	200
TLS	TLSv1.3 TLS_AES_256_GCM_SHA384
Certificado, titular	CN=machinewitness.eu
Certificado, emisor	C=US, O=Google Trust Services, CN=WE1
Certificado válido	Sep 20 12:45:20 2026 GMT hasta Dec 19 13:45:17 2026 GMT
Cadena de certificados (SHA-256 de cada DER)	aec098364a0137418a01ed8935ac9d0070ffb370429a16df8b70a8cd73cd5e53 · 1dfc1605fbad358d8bc844f76d15203fac9ca5c1a79fd4857ffaf2864fbebfb96 · 76b27b80a58027dc3cf1da68dac17010ed93997d0b603e2fadbe85012493b5a7
Integridad del contenido	Almacenado íntegramente; no se aplicó ningún límite de bytes a esta observación.
SHA-256 de los bytes entregados	ae6ef8da7be4397f9d37a57be7b522f3f2b84d2b6af133ba26e8379acab36e32
SHA-256 de la procedencia	a20e01fd655b9927a077d667fbcf68afe2a2600552863b6ac8ae84297974a730
Hoja	10f22b28b520a9ea35e4e1be81f525f7aa04035157b9d1cb22ced67375f3fd52
Posición en el árbol del día	índice 24 de 586,787 hojas, 20 pasos hasta la raíz
Raíz del día	d568805980fcd63d4fbd54959ec16efda605b13fc1cae54b88fb94a7ebdc55d
Anclajes de la raíz del día	sello de tiempo RFC 3161 (FreeTSA) · OpenTimestamps (Bitcoin) · sello cualificado de tiempo eIDAS (GLOBALTRUST)
Paquete	machinewitness-extract-witness-2-2026-09-28-d0199b52c7cc.zip, SHA-256 d69c6efb8da19d2a951419677ace17a25d2e079ff7799e2fea49ef21e5ed5f6

Ambos testigos recibieron los mismos bytes: el SHA-256 de los bytes entregados es idéntico en ambos paquetes. Sus hojas difieren, como deben: cada hoja vincula también la hora de obtención propia del testigo y su propio registro de procedencia.

2. Obtención

Cada testigo es una máquina distinta en un proveedor distinto de un país distinto, con sus propias claves. Ninguno puede escribir en el otro. Cada uno obtiene la dirección con una simple solicitud HTTP GET, identificándose con el agente de usuario publicado MachineWitnessBot/0.1 (+https://machinewitness.eu/bot; evidence archive; contact: bot@machinewitness.eu), sin enviar cookies y siguiendo las redirecciones; la dirección que respondió finalmente se registra como dirección final. El testigo registra el estado, todas las cabeceras de respuesta, la versión y el cifrado TLS, el certificado del servidor y el SHA-256 de cada certificado de la cadena presentada y, cuando se ha registrado, su propia dirección y la dirección que respondió. No inicia sesión, no hace clic, no rellena nada ni ejecuta los scripts de la página. Lo registrado es, por tanto, lo que esta dirección entregó a esta solicitud en este momento, no lo que pudiera haber mostrado a otro visitante en otro momento.

3. Almacenamiento y huella

Los bytes entregados se almacenan tal como se recibieron. Su huella SHA-256 se calcula exactamente sobre esos bytes. Si los bytes almacenados están completos consta a su vez en la procedencia (truncated), y la línea de integridad de arriba lo repite. El registro de procedencia (dirección final, cabeceras, datos TLS, hashes de los certificados, direcciones) se serializa como JSON compacto con claves ordenadas y recibe a su vez una huella.

4. La hoja

La observación se reduce a un único valor, la hoja:

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

El byte de prefijo `0x00` separa las hojas de los nodos internos del árbol, de modo que ningún nodo interno pueda presentarse como hoja. Para esta descripción se ha recalculado la hoja de cada testigo a partir de sus campos registrados y coincide con la hoja del paquete.

5. El árbol y la raíz del día

Cada observación que un testigo realiza en un día UTC se convierte en una hoja del árbol de Merkle de ese día. Los nodos internos son SHA-256(`0x01` || izquierda || derecha). El valor único de la cima, la raíz del día, cambia si cambia cualquier byte de cualquier observación de ese día. El paquete contiene el camino desde esta hoja hasta la raíz: los hashes hermanos y su lado. Plegándolos en orden se reproduce la raíz. La raíz se publica en el registro público de este archivo bajo una dirección fija para el día.

6. Anclajes

Esa misma noche, la raíz del día se entrega a terceros ajenos a este archivo: una autoridad de sellado de tiempo RFC 3161, los calendarios de OpenTimestamps, que la agregan en la cadena de bloques de Bitcoin, y, desde el 31 de julio de 2026, GLOBALTRUST (e-commerce monitoring GmbH, Austria), un prestador cualificado de servicios de confianza incluido en la Lista de confianza de la UE, que emite un sello cualificado de tiempo electrónico. Cada anclaje acredita que la raíz existía en el momento que indica. A partir de ahí, una raíz no puede sustituirse sin que los anclajes dejen de coincidir. El sello cualificado de tiempo es el que goza de la presunción establecida en el artículo 41, apartado 2, del Reglamento (UE) n.º 910/2014 en cuanto a la exactitud de su fecha y hora y a la integridad de los datos a los que están vinculadas.

7. Dos testigos

El segundo testigo realiza su propia obtención, construye su propio árbol y toma sus propios anclajes. Las dos raíces no son idénticas ni pretenden serlo: cada testigo observa según su propio calendario y registra su propia hora de obtención. Lo que permiten es una comparación: para la misma dirección el mismo día, el paquete de cada testigo muestra lo que ese testigo recibió, y ambos pueden contrastarse. Ambos testigos recibieron los mismos bytes: el SHA-256 de los bytes entregados es idéntico en ambos paquetes. Sus hojas difieren, como deben: cada hoja vincula también la hora de obtención propia del testigo y su propio registro de procedencia.

8. Repetir los pasos sin nosotros

1. **Bytes.** `sha256sum payload-<id>.bin` da el valor que figura en `leaf-<id>.json` como `payload_sha256`.
2. **Procedencia.** `sha256sum provenance-<id>.json`, el archivo tal cual, da `provenance_sha256`.
3. **Hoja.** La fórmula del apartado 4, con los cinco campos de `leaf-<id>.json`, da `leaf`. Sirve cualquier herramienta que calcule SHA-256.
4. **Raíz.** Plegar el camino de `proof-<id>.json` como se describe en el apartado 5; el resultado es la raíz de `<día>.root.txt` y del registro público.
5. **Anclajes.** `openssl ts -verify` para los dos sellos RFC 3161 frente a las cadenas de certificados de los emisores; `ots verify` para la prueba de OpenTimestamps.

Cada paquete incluye estos pasos en un archivo `VERIFY.txt`. Ninguno necesita software de este archivo.

9. Objeciones habituales y la respuesta técnica

El operador podría haber cambiado el registro después.

Un cambio en cualquier byte cambia la hoja, la raíz y los anclajes del día. Los anclajes se emitieron la noche en que se selló el día y están en manos de terceros. Un registro modificado deja de coincidir con ellos, y esto puede comprobarse sin nosotros (apartado 8, pasos 4 y 5).

El registro muestra lo que el testigo quiso almacenar, no lo que envió el servidor.

Los bytes se almacenan tal como se recibieron y reciben su huella antes de que les ocurra cualquier otra cosa. La integridad del contenido consta como campo sellado, no se afirma después. Cuando se aplicó un límite de bytes, el registro lo indica para esa observación.

El servidor pudo mostrar otra cosa a otros visitantes.

Es cierto y se dice, no se oculta: el registro abarca lo que esta dirección entregó a esta solicitud, identificada por el agente de usuario publicado, en el momento indicado. No afirma nada sobre otras solicitudes.

La hora de obtención depende del reloj del propio testigo.

La hora de obtención la registra el testigo. Los anclajes fijan de forma independiente el último momento en que existía el día completo, incluida esta observación. El sello cualificado de tiempo indica su hora bajo la supervisión descrita en el apartado 6.

Una sola máquina podría equivocarse.

Son dos, en proveedores distintos, con claves separadas, y ninguna puede escribir en la otra (apartado 7).

10. Lo que esta descripción no es

No contiene ninguna afirmación sobre asunto alguno, ninguna valoración de la conducta de nadie ni ninguna opinión sobre las perspectivas de un procedimiento. Describe un procedimiento y los parámetros de una observación. Lo que de ella se derive lo deciden las partes y el tribunal.

Parte B · Respuestas técnicas por escrito, ronda 1

Preguntas recibidas por escrito sobre la observación anterior, respondidas por escrito. Las respuestas se remiten al método publicado y no dicen nada sobre el asunto.

El extracto da dos raíces del día distintas para el 28 de septiembre de 2026, una por testigo.

¿Cuál de las dos es la correcta?

Ambas. Cada testigo construye su propio árbol sobre sus propias observaciones de ese día y ancla su propia raíz; los dos árboles contienen conjuntos distintos de hojas, registradas en momentos distintos, por lo que sus raíces difieren por construcción. Una raíz se comprueba frente al registro y los anclajes del testigo que la generó. Lo que los dos testigos tienen en común no es la raíz, sino la observación: para esta dirección ambos paquetes llevan el mismo SHA-256 de los bytes entregados.

¿Cómo puede comprobarse que el robots.txt no se modificó después de almacenarse?

Recalculando, sin nosotros: el SHA-256 de payload-<id>.bin debe coincidir con payload_sha256 en el archivo de la hoja; la hoja debe poder recalcularse a partir de sus cinco campos (apartado 4); el camino de proof-<id>.json debe plegarse hasta la raíz del día; y la raíz del día debe coincidir con los anclajes emitidos esa misma noche (apartado 8). Si se hubiera cambiado un solo byte después del 28 de septiembre de 2026, fallaría una de estas cuatro comprobaciones.

Lugar, fecha

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

MUESTRA, sin firmar.