PROTOCOLO DE VERIFICACIÓN

Sistema RedPluma – Infraestructura de Certificación de Integridad de Datos

Versión: 2.0
Fecha: 24/02/2026
Autor: RedPluma / Ing. Benjamín Abraham
Tipo de documento: Protocolo de Verificación

Índice de contenido

1. Introducción

El presente protocolo establece el procedimiento técnico para verificar la integridad de datos certificados mediante el sistema RedPluma.

Su objetivo es permitir que cualquier tercero (perito, auditor o parte interesada) pueda comprobar de forma independiente que un conjunto de datos **no ha sido alterado desde su certificación**.

El proceso se basa en:

* funciones hash criptográficas
* estructuras de Merkle
* pruebas OpenTimestamps (OTS)
* anclaje en la red Bitcoin

2. Elementos necesarios

Para realizar la verificación, se requiere:

Datos originales

* archivo o dataset (ej: `dataset.zip`)

Archivos de certificación

* `report.json`
* `merkle_root.txt`
* `merkle_root.txt.ots`
* `merkle_proofs.json`

Herramientas

* herramienta de hashing (SHA-256)
* cliente OpenTimestamps (`ots`)
* acceso a explorador blockchain (ej: mempool)

3. Alcance de la verificación

El presente protocolo permite validar:

- ✔ Integridad del archivo
- ✔ Correspondencia con el certificado
- ✔ Pertenencia al Merkle Tree
- ✔ Anclaje en blockchain
- ✔ Existencia en un momento determinado

No permite validar:

❌Veracidad del contenido
❌Autoría
❌Contexto

4. Procedimiento de verificación

4.1 PASO 1 – Verificación del hash del dataset

4.1.1. Calcular hash

Aplicar doble SHA-256 sobre el archivo:

bash
sha256sum dataset.zip
sha256sum <resultado anterior>

4.1.2. Comparar

Comparar el resultado con:

json
report.json → "zip_sha256"

✔ Resultado esperado

* Si coincide → integridad del dataset confirmada
* Si no coincide → el archivo fue modificado

4.2 PASO 2 – Verificación del Merkle Root

4.2.1. Obtener root

Abrir:

plaintext
merkle_root.txt

4.2.2. Validar contra proofs

Usar `merkle_proofs.json`:

* reconstruir el árbol
* verificar que cada hash conduce al mismo root

✔ Resultado esperado

* Todos los caminos llevan al mismo Merkle Root

4.3 PASO 3 – Verificación individual de archivos (opcional)

Para cada archivo:

1. recalcular hash
2. aplicar la proof correspondiente
3. verificar coincidencia con root

 Esto demuestra que cada archivo pertenece al dataset original

4.4 PASO 4 – Verificación OTS

4.4.1. Ejecutar

bash
ots verify merkle_root.txt.ots

4.4.2. Resultado esperado

Salida similar a:

plaintext
Success! Bitcoin block XXXXX

Esto indica:

* el hash fue anclado en Bitcoin
* existe desde ese bloque

4.5 PASO 5 – Validación en blockchain

4.5.1. Obtener block height

Desde resultado OTS

4.5.2. Consultar blockchain

Ejemplo: mempool.space

Verificar:

* existencia del bloque
- fecha del bloque
-

✔ Resultado esperado

* el bloque existe
* la fecha es consistente

4.6 PASO 6 – Validación temporal

Conclusión: El dataset existía al menos desde la fecha del bloque Bitcoin

5. Interpretación de resultados

✔ Caso válido

Si todos los pasos coinciden:

* el archivo es íntegro
* no fue modificado
* existía en la fecha indicada

❌Caso inválido

Si falla alguno:

* hash distinto → archivo alterado
* merkle inconsistente → manipulación
* OTS inválido → no anclado

6. Propiedades garantizadas

Este proceso garantiza:

* integridad criptográfica
* trazabilidad
* verificabilidad independiente
* resistencia a manipulación

7. Independencia del sistema

El procedimiento puede realizarse:

* sin acceso a RedPluma
* sin acceso a sus servidores

Solo con los archivos generados

8. Reproducibilidad

El proceso puede ser repetido por:

* cualquier perito
* cualquier auditor

 obteniendo el mismo resultado

9. Consideraciones técnicas

9.1. Determinismo

* orden de archivos fijo
* reglas de construcción definidas

9.2. Hashing

* doble SHA-256
* irreversible

9.3. OpenTimestamps

* agrupa múltiples pruebas
* utiliza blockchain Bitcoin

10. Conclusión

El presente protocolo permite verificar de manera objetiva, independiente y reproducible la integridad de datos certificados mediante RedPluma.

Su aplicación garantiza que cualquier alteración posterior al momento de certificación será detectable, proporcionando un mecanismo robusto para la validación de información digital.