Cinvestav

CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS DEL INSTITUTO POLITÉCNICO NACIONAL

Manual de uso: Aplicación para la generación de huellas criptográficas

V1.0

Ciudad de México, 2 de abril de 2023.

CONTROL DE VERSIONES

Versión	Comentario/Descripción	Responsable de Creación / Actualización / Revisión	Fecha de Creación / Actualización / Revisión
1.0	Creación	Arturo Díaz Pérez	03/02/2023
1.1	Adecuación del contenido	Juan Armando Barró Lugo	02/04/2023

Autorizaciones y Responsables

Elaboró

Fecha	Puesto	Nombre
03/02/2023	Líder de Proyecto	Arturo Díaz Pérez

Revisó

Fecha	Puesto	Nombre
22/03/2023	Líder de Proyecto	Dr. Arturo Díaz Pérez

TABLA DE CONTENIDO

CONTROL DE VERSIONES **2**

AUTORIZACIONES Y RESPONSABLES **2**

ELABORÓ **2**
REVISÓ **2**

RESUMEN **4**

OBJETIVO **5**

INFORME DE VERIFICACIÓN DOCUMENTAL Y NORMATIVA DEL ÁREA ISN **6**

1 INTRODUCCIÓN **6**
2 REQUERIMIENTOS **6**
3 METODOLOGÍA. **6**
3.1 PASO 1: GENERAR LLAVES 6
3.2 PASO 2: RECOPILACIÓN DE INVENTARIO 7
3.3 PASO 3: HUELLAS CRIPTOGRÁFICAS ORIGINALES 8
3.4 PASO 4: CONSTANCIA DE HUELLAS CRIPTOGRÁFICAS ORIGINALES 8
3.5 PASO 5: HUELLAS CRIPTOGRAFICAS INICIALES, INTERMEDIAS Y FINALES 9
3.6 PASO 6: RESGUARDO DE HUELLAS 9
3.7 PASO 7: VALIDACION DE HUELLAS CRIPTOGRÁFICAS 10

Resumen

En este documento, el Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional presenta el manual para la utilización de la aplicación para la generación de huellas criptográficas. Esta aplicación será utilizada para la validación de integridad de los componentes que integran el SIVEI durante la jornada electoral.

Objetivo

Validar que el sistema de voto electrónico por internet (SIVEI) que operará el día de la Jornada Electoral, corresponda al software auditad. La validación respecto a la correspondencia del software auditado y el utilizado en la operación del SIVEI, se tendrá que realizar al inicio, durante y al final de la operación del sistema.

Manual de uso: Aplicación para la generación de huellas criptográficas

v.1.1

1 Introducción

Especialistas del ente auditor deberán llevar a cabo un procedimiento técnico para verificar que los programas auditados se encuentren operando (sin cambios en su interior) desde el inicio y hasta el cierre de operación del SIVEI.

La validación de las aplicaciones se realizará mediante huellas criptográficas para cada evento considerado por el INE (simulacros y jornada electoral). En este proceso, un software desarrollado por el ente auditor automáticamente creará las huellas criptográficas a las aplicaciones del SIVEI inicializadas por el PROVEEDOR (mediante el algoritmo SHA3-256) y serán firmadas digitalmente utilizando el algoritmo RSA para la creación de llaves pública/privada. Este modelo garantiza que solo se validarán las huellas criptográficas creadas por el PROVEEDOR.

2 Requerimientos

A continuación, se listan los requerimientos de la aplicación ***2103Proveedor.jar*** para la generación de huellas criptográficas del inventario de componentes del SIVEI, desarrollado por el ente auditor.

1. Contar con JAVA 8, si no se tiene instalado dirigirse a la página http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html, en la página encontrarán una lista de archivos para su instalación, el Proveedor debe de seleccionar el adecuado para su sistema operativo. Una vez obtenido el archivo para la instalación de JAVA es necesario seguir los pasos de instalación.
2. Contar con permisos de escritura y lectura del sistema de archivos dentro del equipo donde será ejecutada la aplicación o sobre las carpetas que serán utilizadas en el inventario (carpeta ubicada en la ruta del software utilizado por el proveedor).
3. El aplicativo ***2103Proveedor.jar*** deberá ser ejecutado en un entorno con acceso al ambiente de producción (Alpha, Bravo y Charlie, Console) donde se encuentren instalados los scripts del SIVEI.
4. Contar con el aplicativo ***2103Proveedor.jar,*** el cual le será proporcionado al proveedor por parte del ente auditor.

3 Metodología.

A continuación, se describe el proceso para la validación de los componentes que conforman el SIVEI.

3.1 Paso 1: Generar llaves

El Proveedor ejecutará la aplicación para crear las llaves pública y privada. Para ejecutar la aplicación es el siguiente comando:

java -jar 2103Proveedor.jar generarLlaves Llavero/ Original

Donde ***java -jar 2103Proveedor.jar*** es el ejecutable creado para el Proveedor, ***generarLlaves*** indica la acción que el software debe realizar, ***Llavero/*** es la ruta donde se guardarán las dos llaves de forma local (para no comprometer la seguridad se sugiere crear esta carpeta) y ***Original*** es el nombre de la actividad que se está realizando.

La llave privada se quedará almacenada de manera local en la carpeta indicada en la ejecución de la aplicación (***Llavero/***), esta llave quedará al resguardo del personal del PROVEEDOR, el sistema no la enviará a ninguna entidad involucrada, el proveedor es responsable de resguardarla, debido a que será usada para firma de los archivos del inventario durante todo el evento. (se recomienda al proveedor conservar la llave y por ningún motivo compartirla con terceros).

Las llaves se utilizarán posteriormente en el proceso de firma digital y validación, los cuales permitirán garantizar que solo se validarán las huellas criptográficas creadas por el PROVEEDOR.

Nota: El software ***generarLlaves*** detecta si en la ruta ***Llavero/*** ya existen las llaves. Si es el caso muestra un mensaje que ya existen las llaves y no se generan. Esto ocurre porque las llaves se deben de generar una sola vez, inicialmente la carpeta **Llavero/** no contendrá alguna llave generada.

3.2 Paso 2: Recopilación de inventario

Actor responsable: Proveedor

El Proveedor deberá organizar los archivos de los cuales se obtendrá la huella digital y procederá a organizarlos en una carpeta llamada ***Inventario***, la cual deberá incluir los archivos previamente acordados con el INE y el ente auditor.

Para el primer simulacro se acordó que los archivos serían los siguientes:

- Script de concentración de hashes *(Hashes.sh).* Script desarrollado por el Proveedor para obtener los hashes de los componentes del SIVE en la infraestructura.
- Resultado del script de hashes *(hashes_inventario.txt)*. Resultado con los hashes de los componentes del SIVEI obtenido por el script *Hashes.sh*. Cada que se vaya a recopilar el inventario, el proveedor debe ejecutar el script *Hashes.sh* para generar este archivo .txt.

A sí mismo, se invita a tomar las siguientes consideraciones:

- Los archivos deben usar la misma nomenclatura (siempre usar los nombres que se indican entre paréntesis) durante toda la jornada (o simulacro).
- El script *Hashes.sh* deberá ser ejecutado una vez que los eventos hayan sido sellados e inicializados en la infraestructura del SIVEI.
- Es pertinente que el script *Hashes.sh* no calcule hashes de ninguna base de datos o script cuyo contenido pueda cambiar durante el tiempo a lo largo de la jornada electoral/simualcro.
- Se deben eliminar las estampas de tiempo del archivo *hashes_inventario.txt* (las primeras 2 líneas), con el fin de evitar que el resultado del calculo hash cambie con cada ejecución.

Una vez atendido las consideraciones previamente descritas, el proveedor procederá a colocar *Hashes.sh* y *hashes_inventario.txt* en la carpeta ***Inventario/.***

3.3 Paso 3: Huellas criptográficas originales

Actor responsable: Proveedor

El Proveedor ejecutará la aplicación para las firmas criptográficas iniciales. Para ejecutar la aplicación se deberá utilizar el siguiente comando:

java -jar 2103Proveedor.jar firmarArchivos Inventario/ Llavero/LlavePrivada S1-Original Ciudad

Donde ***java -jar 2103Proveedor.jar*** es la aplicación de generación de huellas criptográficas y firmas digitales, ***Inventario/*** es la ruta donde se encuentran los archivos de las bases de datos y aplicaciones vacías e inicializadas, ***Llavero/LlavePrivada*** es la ruta donde se encuentra la llave privada (SK), ***S1-Original*** es el nombre que se le da al lote de firmas iniciales y ***Ciudad*** es el nombre de la ciudad en la que fue realizada la generación de las firmas.

La aplicación en forma automática realizará las siguientes acciones:

1. Se obtendrán las huellas criptográficas de cada documento que se encuentre en el inventario de archivos usando el algoritmo SHA3-256. Si el archivo es superior a 1Mb, se realiza una división del archivo en 4 partes iguales (*chunks*). Cada *chunk* es procesado en paralelo obteniendo su hash (H) y finalmente se realiza un hash de la concatenación de estos 4. Este proceso se puede expresar con la siguiente formula:

$$\mathrm{H}(H(\mathrm{Chunk}_1) + H(\mathrm{Chunk}_2) + H(\mathrm{Chunk}_3) + H(\mathrm{Chunk}_4))$$

Nota: Si los datos no pueden ser divididos en 4 partes iguales, el chunk número 4 se queda con el sobrante.

2. Cada huella criptográfica será firmada digitalmente utilizando la llave privada SK mediante el algoritmo RSA. En el proceso de validación de huellas posterior, se validará esta firma para garantizar que las huellas validadas han sido generadas únicamente por el Proveedor. E

Finalmente, el Proveedor enviará tanto la **LlavePublica** y el archivo **JSON** (el nombre de este archivo corresponde al nombre del lote especificado en el comando de ejecución, p. ej. S1-Original.json) al Ente Auditor por medio de correo electrónico.

Nota: Para el simulacro 1, la validación de los componentes del SIVEI será realizada a grano grueso, es decir, se calculará el hash únicamente de 2 archivos (*Hashes.sh, y hashes_inventario.txt*), donde *hashes_inventario.txt* contiene a su vez los hashes de todos los componentes del SIVEI. En caso de que algun componente cambie se podra detectar el cambio durante la valdiacion, sin embargo no sera posible detectar que componente especifico fue el que cambió. Esto sera atentido para los proximos simulacros.

3.4 Paso 4: Res gurdo seguro de huellas criptográficas Originales y generación de Constancia de generación de huellas criptográficas Originales.

Actor responsable: Auditor

El Auditor generará la constancia de huellas criptográficas iniciales. Para ello se utilizara un aplicativo desarrollado por el ente auditor el cual tomará el archivo **JSON** generado por la aplicación ***2103proveedor.jar,*** y generará un reporte con las huellas criptográficas originales.

Esta actividad tiene como unico objetivo la creacion de una constancia que resguarde las huellas criptograficas originales. En este sentido, el reporte hace constar que las huellas reportadas en el son las huellas criptograficas originales de los archivos del inventario y cuyas cuales se utilizaran como punto de comparacion para validar las huellas criptograficas que se generaran al inicio del evento, durante el evento y al finalizar el evento de la jornada electoral. Durante esta etapa no se aplica ningun tipo de mecanismo de validacion de huellas o firmas, siendo esto realizado mas adelante en el proceso de validacion (Ver 3.7).

3.5 Paso 5: Huellas criptográficas Iniciales, intermedias y finales

Actor responsable: Proveedor

El Proveedor ejecutará la aplicación de firmas criptográficas nuevamente como esta especificado en el paso 3. Este proceso se realizara 3 veces durante la duración de los eventos. A continuacion se muestra un ejemplo de los comandos a ejecutarse durante el simulacro/jornada electoral.

Al inicio del evento:

java -jar 2103Proveedor.jar firmarArchivos Inventario/ Llavero/LlavePrivada S1-Inicio Toluca

Durante el evento:

java -jar 2103Proveedor.jar firmarArchivos Inventario/ Llavero/LlavePrivada S1-Intermedio Toluca

Al finalizar el evento:

java -jar 2103Proveedor.jar firmarArchivos Inventario/ Llavero/LlavePrivada S1-Final Toluca

Se recomienda mantener la misma nomenclatura de nombre para los lotes de las firmas. En este ejemplo, ***S1-Inicio*** corresponde al lote de firmas iniciales del Simulacro 1, ***S1-Intermedio*** al lote de firmas intermedias del Simulacro 1 y ***S1-Final*** corresponde al lote de firmas finales del Simulacro 1.

Después de cada ejecución de la aplicación ***2103Proveedor.jar*** (generación de huellas criptográficas), el proveedor deberá compartir el archivo **JSON** resultante al Ente Auditor.

3.6 Paso 6: Resguardo de huellas

Actor responsable: Auditor

Para cada resultado producido por la aplicación ***2103Proveedor.jar***, el Auditor resguardara los resultados para su posterior validación. El resguardo se realizará por medio de una aplicación desarrollada por el ente auditor llamada **2*103Auditor.*** El auditor ejecutará el siguiente comando:

java -jar 2103Auditor.jar /home/user/auditoria/nombreArchivo.json

Este aplicativo indexara la informacion contenida en el json en una base de datos local para su resguardo.

3.7 Paso 7: Validación de huellas criptográficas

Actor responsable: Auditor

El Auditor ejecutará la aplicación de validación de firmas criptográficas antes del inicio de la jornada electoral, durante la jornada y al finalizar la jornada. Esta aplicación realizara la validación de las firmas digitales utilizando la llave publica (proporcionada por el proveedor) y comparando los hashes originales contra los del evento. En otras palabras, se validará que los hashes originales sean los mismos que los hashes obtenidos durante el inicio, a la mitad y al final de la jornada.
Para ejecutar la aplicación el auditor utilizara el siguiente comando:

java -jar 2103INE.jar validar Llavero/LlavePublica Original Evento.

Donde ***java -jar 2103INE.jar*** es la aplicación, ***validar*** es el nombre de la actividad que se está realizando, ***Llavero/LlavePublica*** es la ruta donde se encuentran la llave pública (n/a, si desea descargar la llave del servicio de almacenamiento), **Original** es el nombre del lote de firmas generadas inicialmente y ***Evento*** es el nombre del paquete de firmas que se quiere validar: Inicio, Intermedio, o Final.

Para la validacion se hace uso de la llave pública proporcionada por el proveedor, y se utiliza para validar la firma digital tanto del lote de hashes del evento Original (los hash presentes en la constancia de huellas descrita en al seccion 3.4) y del lote de firmas del evento (hash iniciales, intermedios yo finales). En este contexto, si las firmas digitales no se pueden validar utilizando la llave publica, la aplicación lanzara un error y se reportara en el CSV resultante.

Una vez finalizado la validación, el Auditor generará la constancia de hechos. Para ello se utilizará un aplicativo desarrollado por el ente auditor el cual tomará archivo **CSV** generado por la aplicación ***2103INE.jar,*** para generar un reporte con la validación de las firmas de los archivos originales y el lote de firmas a comparar (iniciales, intermedias o finales).

A t e n t a m e n t e

Dr. José Luis González Compeán
Líder del área de infraestructura y seguridad en la nube