PROYECTO CRIPTOGRAFICO

DESCRIPCIÓN BREVE

DOCUMENTO BASADO EN EL ANALISIS Y DISEÑO DEL PROYECTO

kevin acuña paredes
Martín José Morata García
AMykhaylov Krutkova, Oleksandr

PROYECTO CRIPTOGRACIA CON PYTHON

Resumen Ejecutivo

El proyecto tiene como objetivo desarrollar un sistema criptográfico implementado en Python, con el propósito de proporcionar soluciones seguras de cifrado y descifrado de datos. La importancia radica en la creciente necesidad de salvaguardar la confidencialidad y la integridad de la información en entornos digitales. La aplicación de técnicas criptográficas se vuelve esencial en un contexto donde la ciberseguridad es una preocupación central.

- Objetivos principales del proyecto.
    - Implementación de Algoritmos Criptográficos: Desarrollar y poner en práctica algoritmos criptográficos robustos, como AES y RSA, para garantizar la seguridad de los datos.
    - Facilidad de Uso: Crear una interfaz de usuario intuitiva que permita a los usuarios, incluso sin experiencia técnica, aprovechar las capacidades criptográficas de manera sencilla y segura.
    - Seguridad y Resistencia: Evaluar y garantizar la resistencia del sistema frente a posibles ataques criptográficos, identificando y mitigando posibles vulnerabilidades.
- Alcance del proyecto.
    - El proyecto abarcará desde la investigación y selección de algoritmos criptográficos apropiados hasta la implementación y validación de estos en un entorno Python. Además, se incluirá la creación de una documentación completa y comprensible para facilitar la implementación y comprensión del sistema por parte de los usuarios finales.

Introducción

CIFRADO SIMETRICO

Comenzamos la explicación con lo que debería hacer el usuario que quiera encriptar el archivo deseado.

1. Insertar el archivo en el botón que muestra el texto “Insertar Archivo”.

2. Seguido de eso encontrará una casilla de titulo “Clave”, la cual tendrá que introducir la contraseña que desea

3. Tras introducir la clave, el usuario deberá pulsar el botón “Encriptar” que se encuentra justo debajo.

4. Cuando lo presiones, le saldrá un resultado nuevo que le dirá, “Archivo Encriptado:”.

5. Y después le saldrá otro botón que le dirá “Descargar archivo”, le tiene que presionar y le descargará el archivo ya cifrado.

Luego de saber como encriptar seguiremos con lo contrario, la desencriptación

1. Comenzamos con ir al lugar donde muestra de titulo “Desencriptar archivo”.

2. Le mostrará de nuevo un botón de texto “Insertar archivo encriptado”, a la cual tiene que presionar e insertar el archivo encriptado.

3. Al finalizar de insertar debe ir a la casilla de titulo “Clave” e insertar la contraseña que debe saber, ya sea porque creó el archivo encriptado o el usuario que creó el archivo le compartió la contraseña.

4. Tras introducir la contraseña debe presionar el botón “Desencriptar”.

5. Aparecerá otro contenido de titulo “Archivo Desencriptado”.

6. Y por ultimo le saldrá otro botón de texto “Descargar archivo desencriptado”, la cual al presionarle le descargará el archivo desencriptado y sin problemas.

CIFRADO ASIMETRICO

Claves Públicas:

En este manual, vamos a describir cual es el proceso de uso del apartado "Claves Públicas" de la aplicación. Es importante aclarar que esto es una parte **no obligatoria**, para la encriptación asimétrica, ya que las claves pueden ser generadas y compartidas mediante diferentes métodos. Esto tan solo facilita el trabajo del usuario.

La pestaña de "Claves públicas", consta de 3 apartados.

1. **Descarga de claves:**

   El usuario puede descargar diferentes claves **públicas** que otros usuarios carguen al servidor.

2. **Carga de claves:**

   El usuario puede cargar diferentes claves **públicas** que han sido generadas anteriormente. Para identificar las claves públicas, se utilizará su nombre. Cuando una clave es introducida al servidor, se comprobará que el nombre introducido no esté repetida en ninguna de las claves en el servidor.

Por supuesto, diferentes métodos serán utilizados para comprobar que los archivos subidos no sean maliciosos o indebidos.

3. **Generar nuevas claves:**

Consta de un simple botón, que generará un nuevo par de claves, **pública** y **privada**, y las descargará para el usuario inmediatamente.

A continuación, entraremos un poco más en detalle sobre cómo funciona la aplicación.

Esta sería una vista previa de cómo lucirá la página de la base de datos.

Descarga de claves

Podemos observar que el primer apartado, "Descarga de claves", consta de 2 elementos, un select, que permite seleccionar la clave que se desea descargar, y un botón, para ejecutar el código de descarga.

Para hacer esto, no ha sido necesario descargar ninguna base de datos, todos los archivos ".key", son almacenados en la ruta relativa "static/public_keys" de la aplicación, en local, siendo cada archivo dentro de esta carpeta una clave independiente. Se pueden cargar la misma clave con diferentes resultados múltiples veces.

Carga de claves

Consta de 3 elementos, un input de tipo texto, en el cual debes escribir el nombre con el que vas a subir tu clave. Un input de tipo archivo, en el cual el usuario deberá subir el archivo de la clave pública, y un botón, que ejecuta el código.

En este apartado se realizarán diferentes comprobaciones para ejecutar que el archivo que se intenta cargar es el debido, entre ellas:

- Comprueba que es una clave pública, y no privada. Si se cargan en claves privadas resultaría en una falla de seguridad.
- Comprueba que el nombre con el que se introduce la clave no existe ya en otro archivo.
- Comprueba que es un archivo ".key", y que es de tipo texto, y no un virus.

Si las comprobaciones no detectan ningún problema y ningún error ocurre, el archivo se cargará a la ruta relativa a la aplicación "static/public_keys", con el nombre "<nombre Introducido>.key"

Generar nuevo par de claves

Este apartado consta de un solo botón. Cuando es presionado, el código genera un nuevo par de claves, público y privado. Tras ello, se introducen en un archivo .Zip, disponible para cualquier Sistema Operativo que pueda utilizar un navegador, y lo descarga al usuario.

Ahora, para que la descarga de los archivos sea posible, es necesario que estos sean almacenados en una ruta temporal primero, en este caso, la relativa a la aplicación "static/temp".

Tras finalizar la descarga, debido a la dificultad de la aplicación, los 3 archivos, "private_key.key", "public_key.key" y "keys.zip" se quedan en esta ruta, y son reemplazados por unos nuevos cada vez que la descarga es requerida.

Cifrado Asimétrico:

En este caso, vamos a describir cual es el proceso de uso del apartado "C.Asimétrico" de la app. Esta pestaña está dividida en 2 apartados: Cifrar y Descifrar.

1. **Cifrar:** Hay 2 modos de cifrado diferente, puedes cargar un archivo .key que tengas en tu equipo local, o puedes elegir una clave pública almacenada de manera local en el servidor.

2. **Descifrar:** Carga el archivo .key de tu clave privada para descifrar el archivo encriptado, anteriormente con una clave pública.

No hay vista previa del aspecto de la página. Aun en desarrollo.

Descifrado

Cuando el usuario pulsa el botón de cifrar, el archivo cifrado se guarda en la ruta local del servidor "static/temp", con el nombre "cifrado_asimetricamente.gpg". Luego, si el usuario ha cargado un archivo .key localmente, este se guarda en la misma ruta como "clave_privada.key" y es cargada al sistema. Tras ello, el archivo resultante por descifrar el archivo, se guardará en la misma ruta, con el nombre "descifrado_asimetricamente.txt", y se enviará al usuario para que se descargue.

Cifrado

El usuario debe cargar el archivo que quiere cifrar, de manera local, tras ello, puede elegir un archivo ".key" localmente para cifrar el archivo. El archivo para cifrar se guarda en la ruta "static/temp" con el nombre "para_cifrar_asim.<extensiónDelArchivo>", y el archivo .key se guarda en la misma ruta, con el nombre "clave_asimetrica.key". Tras ello, el resultado es procesado y este se guarda en la misma ruta con el nombre "cifrado_asimetricamente.gpg", y es enviado al usuario para descargar.

El usuario, alternativamente puede elegir de un select, un archivo .key, almacenada en el servidor, para cifrar el archivo. Si esto es así, no es necesario guardar un archivo .key en la ruta "static/temp", y el proceso es más liviano.

Queda aclarar, que esto está pensado para que las claves puedan ser importadas y exportadas mediante archivos, y que las claves, para que no caduquen, son todas almacenadas en archivos .key en el servidor.

CIFRADO ASIMETRICO

CONCEPTO:

El cifrado híbrido es un enfoque de cifrado que combina dos técnicas de cifrado diferentes: el cifrado simétrico y el cifrado asimétrico. Este método se utiliza comúnmente en sistemas de seguridad de la información, especialmente en el contexto de la seguridad de las comunicaciones en línea.

Aquí hay una descripción básica de cómo funciona el cifrado híbrido:

Cifrado Simétrico:

En el cifrado simétrico, se utiliza una única clave para cifrar y descifrar la información. Aunque es eficiente, surge el desafío de compartir esta clave de manera segura entre las partes involucradas.

Cifrado Asimétrico:

El cifrado asimétrico utiliza un par de claves: una pública y una privada. La clave pública se comparte abiertamente y se utiliza para cifrar datos, mientras que la clave privada, que se mantiene en secreto, se utiliza para descifrarlos. Esto aborda el desafío de la distribución segura de claves.

Cómo Funciona en la Práctica: