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

El cifrado simétrico es una técnica de cifrado en la que se utiliza la misma clave tanto para cifrar como para descifrar la información. En este enfoque, la entidad que cifra los datos (llamada remitente o emisor) y la entidad que los descifra (receptor) comparten una clave secreta. Este tipo de cifrado ha sido utilizado durante siglos y es fundamental para la seguridad de la información.

Encriptación de Archivos:

Insertar el Archivo:

- **Inicia el proceso seleccionando el archivo deseado mediante la interfaz proporcionada.**
- **La selección de archivos se realiza mediante un formulario HTML con un campo de entrada de tipo "file" y un botón de envío.**

Introducir la Clave:

- Después de seleccionar el archivo, ingresa la clave de encriptación en la casilla "Clave".
- Utiliza un campo de entrada de tipo "password" para ocultar la clave mientras se escribe.

Encriptar el Archivo:

- Al presionar el botón "Encriptar", el sistema utiliza la clave proporcionada para cifrar el archivo.
- El formulario se envía al servidor, donde se procesa la encriptación.

Descargar Archivo Encriptado:

- Una vez encriptado, se proporciona un enlace "Descargar archivo" que permite descargar el archivo encriptado.
- El enlace apunta a una ruta en el servidor que gestiona la descarga del archivo encriptado.

Desencriptación de Archivos:

1. **Insertar Archivo Encriptado:**

- En la sección "Desencriptar archivo", selecciona el archivo previamente encriptado mediante la interfaz.
- Se utiliza un formulario similar al de la encriptación.

2. **Introducir la Clave:**

    - Ingresa la clave asociada al archivo encriptado en la casilla "Clave".
    - Este paso es crucial para descifrar el archivo correctamente.

3. **Desencriptar el Archivo:**

    - Al presionar el botón "Desencriptar", el sistema utiliza la clave proporcionada para descifrar el archivo encriptado.
    - El formulario se envía al servidor, donde se realiza la operación de desencriptación.

4. **Descargar Archivo Desencriptado:**

    - Después de desencriptar, se proporciona un enlace "Descargar archivo desencriptado".
    - Este enlace apunta a una ruta en el servidor que maneja la descarga del archivo desencriptado.

INTERACCION DE LA APLICACIÓN

- **El código proporciona funciones para cifrar y descifrar archivos utilizando cifrado simétrico (AES y DES).**

Funcionamiento del Cifrado de Bloques:

1. **División de Datos:**

- Los datos a cifrar se dividen en bloques de tamaño fijo.
- Cada bloque se trata de manera independiente durante el proceso de cifrado.

2. **Cifrado con la Misma Clave:**
    - Se utiliza la misma clave para cifrar cada bloque de datos.
    - Esto simplifica la gestión de claves pero destaca la importancia de mantener la confidencialidad de la clave.
3. **Aplicación del Algoritmo:**
    - Un algoritmo de cifrado, como AES, se aplica a cada bloque de datos de forma secuencial.
    - La combinación de la clave y el algoritmo garantiza la confidencialidad de la información.

Consideraciones Importantes:

- La elección de un algoritmo criptográfico robusto, como AES, es esencial para garantizar la seguridad del cifrado de bloques.
- La gestión adecuada de claves es crucial; se deben implementar prácticas seguras para generar, almacenar y distribuir las claves de cifrado.

CLAVES PUBLICAS

Las claves públicas son componentes esenciales de la criptografía asimétrica. Este enfoque utiliza un par de claves: una clave pública y una clave privada. La clave pública se comparte abiertamente, mientras que la clave privada se mantiene en secreto. Estas claves están matemáticamente relacionadas de tal manera que la información cifrada con una clave solo puede descifrarse eficientemente con la otra clave del par.

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 ASIMETRICO

El cifrado asimétrico, también conocido como criptografía de clave pública, es un método criptográfico que utiliza un par de claves para realizar operaciones de cifrado y descifrado. A diferencia del cifrado simétrico, donde se utiliza la misma clave para ambas operaciones, el cifrado asimétrico utiliza dos claves distintas: una clave pública y una clave privada.

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 HIBRIDO

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:

Encriptación de Archivos:

1. Generar Par de Claves Asimétricas:
    - Antes de iniciar, genera un par de claves asimétricas (pública y privada) utilizando un algoritmo como RSA o ECC.

2. Insertar Archivo:
    - Selecciona el archivo deseado haciendo clic en el botón "Insertar Archivo".
3. Introducir la Clave Simétrica:
    - Genera una clave simétrica única (clave de sesión) y colócala en la casilla titulada "Clave Simétrica".
4. Encriptar Archivo Simétricamente:
    - Utiliza la clave de sesión para cifrar el contenido del archivo simétricamente.
5. Encriptar Clave Simétrica con Clave Pública:
    - La clave de sesión se cifra asimétricamente utilizando la clave pública del destinatario.
6. Descargar Archivo Encriptado:
    - Después de la encriptación, descarga el archivo cifrado haciendo clic en "Descargar archivo encriptado".

Desencriptación de Archivos:

1. Insertar Archivo Encriptado:
    - En la sección "Desencriptar archivo", selecciona el archivo cifrado mediante el botón "Insertar archivo encriptado".
2. Introducir la Clave Privada:
    - Ingresa la clave privada asociada a la clave pública utilizada durante la encriptación.
3. Descifrar Clave Simétrica con Clave Privada:
    - Utiliza la clave privada para descifrar la clave de sesión cifrada asimétricamente.
4. Desencriptar Archivo Simétricamente:

- Con la clave de sesión descifrada, procede a descifrar simétricamente el contenido del archivo.

5. Descargar Archivo Desencriptado:
    - Finalmente, descarga el archivo desencriptado haciendo clic en "Descargar archivo desencriptado".