Informe Laboratorio 4

Sección 1

Sebastián Riquelme
e-mail: sebastian.riquelme1@mail.udp.cl

Mayo de 2023

Índice

1. Descripción de actividades

Para este laboratorio, deberá utilizar Tampermonkey y la librería CryptoJS (con SRI) para lograr obtener los mensajes que le está comunicando su informante. En esta ocasión, su informante fue más osado y se comunicó con usted a través de un sitio web abierto a todo el público https://cripto.tiiny.site/.

Sólo un ojo entrenado como el suyo logrará descifrar cuál es el algoritmo de cifrado utilizado y cuál es la contraseña utilizada para lograr obtener la información que está oculta.

1. Desarrolle un plugin para tampermonkey que permita obtener la llave para el descifrado de los mensajes ocultos en la página web. La llave debe ser impresa por la consola de su navegador al momento de cargar el sitio web. Utilizar la siguiente estructura:

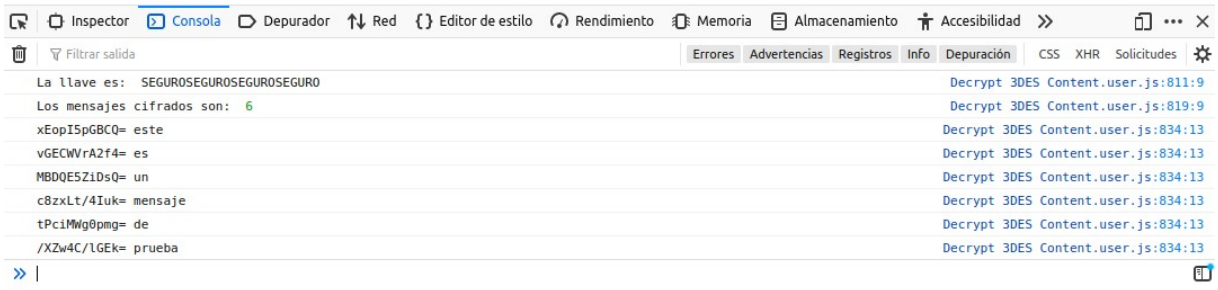
   - La llave es: KEY

2. En el mismo plugin, se debe detectar el patrón que permite identificar la cantidad de mensajes cifrados. Debe imprimir por la consola la cantidad de mensajes cifrados. Utilizar la siguiente estructura: Los mensajes cifrados son: NUMBER

3. En el mismo plugin debe obtener cada mensaje cifrado y descifrarlo. Ambos mensajes deben ser informados por la consola (cifrado espacio descifrado) y además cada mensaje en texto plano debe ser impreso en la página web.

   El script desarrollado debe ser capaz de obtener toda la información del sitio web (llave, cantidad de mensajes, mensajes cifrados) sin ningún valor forzado. Para verificar el correcto funcionamiento de su script se utilizará un sitio web con otro texto y una cantidad distinta de mensajes cifrados. Deberá indicar la url donde se podrá descargar su script.

   Un ejemplo de lo que se debe visualizar en la consola, al ejecutar automáticamente el script, es lo siguiente:



Sin el conocimiento de información secreta, el criptoanálisis se dedica al estudio de sistemas criptográficos con el fin de encontrar debilidades en los sistemas y romper su seguridad. El criptoanálisis es un componente importante del proceso de creación de criptosistemas sólidos. Gracias al criptoanálisis, podemos comprender los criptosistemas y mejorarlos identificando los puntos débiles. Un criptoanalista puede ayudarnos a trabajar en el algoritmo para crear un código secreto más seguro y protegido. Resultado del criptoanálisis es la protección de la información crítica para que no sea interceptada, copiada, modificada o eliminada. Otras tareas de las que pueden ser responsables los criptoanalistas incluyen evaluar, analizar y localizar las debilidades en los sistemas y algoritmos de seguridad criptográfica. Sin el conocimiento de información secreta, el criptoanálisis se dedica al estudio de sistemas criptográficos con el fin de encontrar debilidades en los sistemas y romper su seguridad. El criptoanálisis es un componente importante del proceso de creación de criptosistemas sólidos. Gracias al criptoanálisis, podemos comprender los criptosistemas y mejorarlos identificando los puntos débiles. Un criptoanalista puede ayudarnos a trabajar en el algoritmo para crear un código secreto más seguro y protegido. Resultado del criptoanálisis es la protección de la información crítica para que no sea interceptada, copiada, modificada o eliminada. Otras tareas de las que pueden ser responsables los criptoanalistas incluyen evaluar, analizar y localizar las debilidades en los sistemas y algoritmos de seguridad criptográfica. Sin el conocimiento de información secreta, el criptoanálisis se dedica al estudio de sistemas criptográficos con el fin de encontrar debilidades en los sistemas y romper su seguridad. El criptoanálisis es un componente importante del proceso de creación de criptosistemas sólidos. Gracias al criptoanálisis, podemos comprender los criptosistemas y mejorarlos identificando los puntos débiles. Un criptoanalista puede ayudarnos a trabajar en el algoritmo para crear un código secreto más seguro y protegido. Resultado del criptoanálisis es la protección de la información crítica para que no sea interceptada, copiada, modificada o eliminada. Otras tareas de las que pueden ser responsables los criptoanalistas incluyen evaluar, analizar y localizar las debilidades en los sistemas y algoritmos de seguridad criptográfica. Sin el conocimiento de información secreta, el criptoanálisis se dedica al estudio de sistemas criptográficos con el fin de encontrar debilidades en los sistemas y romper su seguridad. El criptoanálisis es un componente importante del proceso de creación de criptosistemas sólidos. Gracias al criptoanálisis, podemos comprender los criptosistemas y mejorarlos identificando los puntos débiles. Un criptoanalista puede ayudarnos a trabajar en el algoritmo para crear un código secreto más seguro y protegido. Resultado del criptoanálisis es la protección de la información crítica para que no sea interceptada, copiada, modificada o eliminada. Otras tareas de las que pueden ser responsables los criptoanalistas incluyen evaluar, analizar y localizar las debilidades en los sistemas y algoritmos de seguridad criptográfica.

este
es
un
mensaje
de
prueba

```
La llave es:  SEGUROSEGUROSEGUROSEGURO                          Decrypt 3DES Content.user.js:811:9
Los mensajes cifrados son:  6                                   Decrypt 3DES Content.user.js:819:9
xEopI5pGBCQ= este                                              Decrypt 3DES Content.user.js:834:13
vGECWVrA2f4= es                                                Decrypt 3DES Content.user.js:834:13
MBDQE5ZiDsQ= un                                                Decrypt 3DES Content.user.js:834:13
c8zxLt/4Iuk= mensaje                                           Decrypt 3DES Content.user.js:834:13
tPciMWg0pmg= de                                                Decrypt 3DES Content.user.js:834:13
/XZw4C/lGEk= prueba                                            Decrypt 3DES Content.user.js:834:13
» |
```

2. Desarrollo (Parte 1)

2.1. Detecta el cifrado utilizado por el informante

A través del análisis del código HTML y la estructura de los mensajes cifrados, se identificó que el algoritmo utilizado por el informante era TripleDES.

2.2. Logra que el script solo se gatille en el sitio usado por el informante

Mediante el uso de la directiva @match en el script de Tampermonkey, nos aseguramos de que el script sólo se active en el sitio web del informante, "https://cripto.tiiny.site".

2.3. Define función que obtiene automáticamente el password del documento

En el script, definimos una función que automáticamente extrae la llave para el descifrado de los mensajes ocultos en la página web. Esto se hace dividiendo un párrafo específico en el sitio web en oraciones y recogiendo el primer carácter de cada oración.

```
// ==UserScript==
// @name         Cripto Laboratorio
// @namespace    http://tampermonkey.net/
// @version      0.1
// @description Trabajo con cifrado
// @author       You
// @match        https://cripto.tiiny.site
// @grant        none
// ==/UserScript==

(function () {
    "use strict";

    // Agregar la librería CryptoJS con SRI al documento HTML
    var script = document.createElement("script");
    script.src =
        "https://cdnjs.cloudflare.com/ajax/libs/crypto-js/4.1.1/crypto-js.min.js";
    script.integrity =
        "sha512-E8QSvWZ0eCLGk4km3hxSsNmGWbLtSCSUcewDQPQWZF6pEU8GlT8a5fF32wOl1i8ftdMhssTrF/OhyGWwonTcXA==";
    script.crossOrigin = "anonymous";
    document.head.appendChild(script);

    script.onload = function () {
        // Parte 1: Obtener la llave
        let parrafo = document.querySelector(".Parrafo p").innerText;
        let oraciones = parrafo.split(". ");
        let llave = oraciones.map((o) => o[0]).join("");
        console.log("La llave es: " + llave);

        // Parte 2: Identificar la cantidad de mensajes cifrados
        let mensajesCifrados = document.querySelectorAll('div[class^="M"]');
        console.log("Los mensajes cifrados son: " + mensajesCifrados.length);

        // Parte 3: Obtener cada mensaje cifrado y descifrarlo
        mensajesCifrados.forEach((mensajeCifrado) => {
            let mensajeCifradoBase64 = mensajeCifrado.id;
            let mensajeCifradoBytes =
                CryptoJS.enc.Base64.parse(mensajeCifradoBase64);
            let mensajeDescifradoBytes = CryptoJS.TripleDES.decrypt(
                { ciphertext: mensajeCifradoBytes },
                CryptoJS.enc.Utf8.parse(llave),
                { mode: CryptoJS.mode.ECB }
            );
            let mensajeDescifrado = mensajeDescifradoBytes.toString(
                CryptoJS.enc.Utf8
            );
            console.log(mensajeCifradoBase64 + " " + mensajeDescifrado);
            mensajeCifrado.innerText = mensajeDescifrado;
        });
    };
})();
```

Figura 1: Función en JavaScript para obtener la llave

2.4. Muestra la llave por consola

La llave obtenida se imprime en la consola del navegador. Esto permite verificar que la extracción de la llave se realizó correctamente.

3. Desarrollo (Parte 2)

3.1. Reconoce automáticamente la cantidad de mensajes cifrados

El script también identifica automáticamente la cantidad de mensajes cifrados en la página web. Esto se logra buscando todos los elementos div que contienen un mensaje cifrado.

3.2. Muestra la cantidad de mensajes por consola

La cantidad de mensajes cifrados encontrados se imprime en la consola del navegador, permitiendo una verificación rápida y fácil de que se están detectando todos los mensajes cifrados en la página.

4. Desarrollo (Parte 3)

4.1. Importa la librería CryptoJS

El script importa la librería CryptoJS, que proporciona las funciones de cifrado y descifrado necesarias para el análisis de los mensajes.

4.2. Utiliza SRI en la librería CryptoJS

El script importa la librería CryptoJS, que proporciona las funciones de cifrado y descifrado necesarias para el análisis de los mensajes. Ahora, en lugar de utilizar la directiva @require de Tampermonkey, el script crea una etiqueta de script y la añade al documento HTML. Esta etiqueta de script incluye el atributo de integridad con el hash correspondiente a la versión de la biblioteca CryptoJS que estamos utilizando, lo que garantiza la integridad de la biblioteca mediante el uso de Subresource Integrity (SRI). Además, el script ahora espera a que la librería CryptoJS se cargue antes de ejecutar el resto del código, lo que se logra utilizando el evento onload de la etiqueta de script.

4.3. Logra decifrar uno de los mensajes

El script logra descifrar los mensajes cifrados utilizando la llave y la función de descifrado proporcionada por CryptoJS.

4.4. Imprime todos los mensajes por consola

Todos los mensajes, tanto en su forma cifrada como descifrada, se imprimen en la consola del navegador para su verificación.

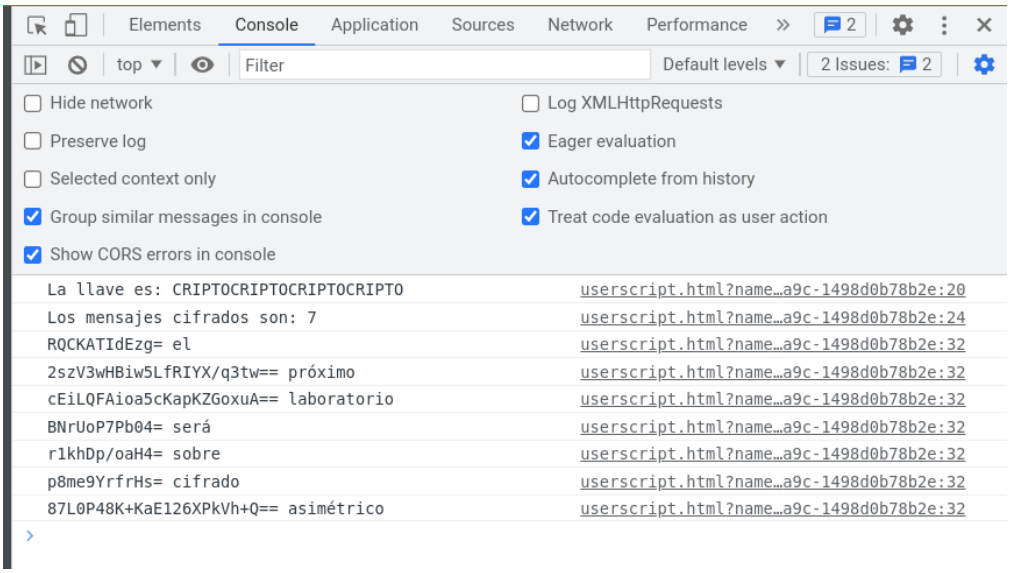
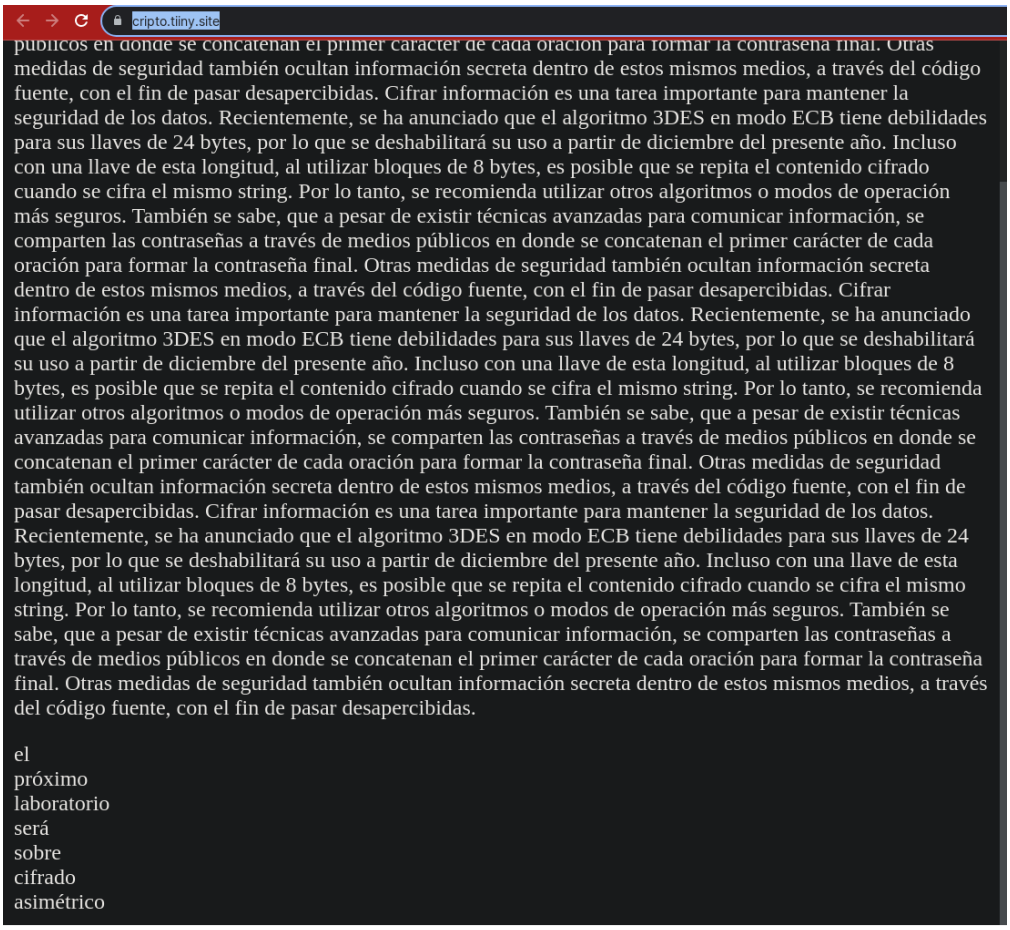


Figura 2: Consola del navegador mostrando los mensajes cifrados y descifrados

4.5. Muestra los mensajes en texto plano en el sitio web

El script reemplaza los mensajes cifrados en la página web con sus versiones descifradas, permitiendo una fácil lectura de los mismos.


cripto.tiiny.site

públicos en donde se concatenan el primer carácter de cada oración para formar la contraseña final. Otras medidas de seguridad también ocultan información secreta dentro de estos mismos medios, a través del código fuente, con el fin de pasar desapercibidas. Cifrar información es una tarea importante para mantener la seguridad de los datos. Recientemente, se ha anunciado que el algoritmo 3DES en modo ECB tiene debilidades para sus llaves de 24 bytes, por lo que se deshabilitará su uso a partir de diciembre del presente año. Incluso con una llave de esta longitud, al utilizar bloques de 8 bytes, es posible que se repita el contenido cifrado cuando se cifra el mismo string. Por lo tanto, se recomienda utilizar otros algoritmos o modos de operación más seguros. También se sabe, que a pesar de existir técnicas avanzadas para comunicar información, se comparten las contraseñas a través de medios públicos en donde se concatenan el primer carácter de cada oración para formar la contraseña final. Otras medidas de seguridad también ocultan información secreta dentro de estos mismos medios, a través del código fuente, con el fin de pasar desapercibidas. Cifrar información es una tarea importante para mantener la seguridad de los datos. Recientemente, se ha anunciado que el algoritmo 3DES en modo ECB tiene debilidades para sus llaves de 24 bytes, por lo que se deshabilitará su uso a partir de diciembre del presente año. Incluso con una llave de esta longitud, al utilizar bloques de 8 bytes, es posible que se repita el contenido cifrado cuando se cifra el mismo string. Por lo tanto, se recomienda utilizar otros algoritmos o modos de operación más seguros. También se sabe, que a pesar de existir técnicas avanzadas para comunicar información, se comparten las contraseñas a través de medios públicos en donde se concatenan el primer carácter de cada oración para formar la contraseña final. Otras medidas de seguridad también ocultan información secreta dentro de estos mismos medios, a través del código fuente, con el fin de pasar desapercibidas. Cifrar información es una tarea importante para mantener la seguridad de los datos. Recientemente, se ha anunciado que el algoritmo 3DES en modo ECB tiene debilidades para sus llaves de 24 bytes, por lo que se deshabilitará su uso a partir de diciembre del presente año. Incluso con una llave de esta longitud, al utilizar bloques de 8 bytes, es posible que se repita el contenido cifrado cuando se cifra el mismo string. Por lo tanto, se recomienda utilizar otros algoritmos o modos de operación más seguros. También se sabe, que a pesar de existir técnicas avanzadas para comunicar información, se comparten las contraseñas a través de medios públicos en donde se concatenan el primer carácter de cada oración para formar la contraseña final. Otras medidas de seguridad también ocultan información secreta dentro de estos mismos medios, a través del código fuente, con el fin de pasar desapercibidas.

el
próximo
laboratorio
será
sobre
cifrado
asimétrico

Figura 3: Se muestran mensaje descifrados en la página web.

4.6. El script logra funcionar con otro texto y otra cantidad de mensajes

El código está estrucutrado de manera tal, que debería funcionar para un HTML que tenga los mensajes en el HTML de la misma forma, es decir, con una etiqueta de class ”M.ªcompañada de un numero para cada mensaje. Cabe destacar que el script solo hace match con la URL del informante, por lo que si se quiere para otra URL, habría que editar este campo.

4.7. Indica url al código .js implementado para su validación

La url para descargar y validar el script implementado es https://greasyfork.org/es/scripts/467202-cripto-laboratorio.

Conclusiones y comentarios

Este laboratorio proporcionó una experiencia práctica invaluable en el uso de Tampermonkey y CryptoJS para interactuar con una página web y descifrar mensajes ocultos. Fue un desafío interesante entender cómo se cifraron los mensajes y encontrar una manera de descifrarlos automáticamente. El script resultante es robusto y capaz de manejar diferentes textos y cantidades de mensajes cifrados, lo cual es una prueba de su eficacia.