Informe Laboratorio 3

Sección 1
Ignacio Santiago Medina Díaz
e-mail: ignacio.medina1@mail.udp.cl

Octubre de 2024

Índice

1. Descripción de actividades

Su objetivo será auditar la implementación de algoritmos hash aplicados a contraseñas en páginas web desde el lado del cliente, así como evaluar la efectividad de estas medidas contra ataques de tipo Pass the Hash (PtH). Para llevar a cabo esta auditoría, deberá registrarse en un sitio web y crear una cuenta, ingresando una contraseña específica para realizar las pruebas.

Al concluir la tarea, es importante que modifique su contraseña por una diferente para garantizar su seguridad.

Dado que la cantidad de sitios chilenos que utilizan hash es limitada, se permite realizar esta tarea en cualquier sitio web a nivel mundial. En este sentido, realice las siguientes actividades:

- Identificación del algoritmo de hash utilizado para las contraseñas al momento del registro en el sitio.
- Identificación del algoritmo de hash utilizado para las contraseñas al momento de iniciar sesión.
- Generación del hash de la contraseña desde la consola del navegador, partiendo de la contraseña en texto plano.
- Interceptación del tráfico de login utilizando BurpSuite desde su equipo.
- Realización de un intento de login, modificando una contraseña incorrecta por el hash obtenido en el punto anterior.
- Descripción de las políticas de privacidad o seguridad relacionadas con las contraseñas, incluyendo un enlace a las mismas.
- Cuatro conclusiones sobre la seguridad o vulnerabilidad de la implementación observada.

2. Desarrollo de actividades según criterio de rúbrica

2.1. Identifica el algoritmo de hash utilizado al momento de registrarse en el sitio

Para comenzar con el laboratorio 3 de Criptografía y Seguridad en Redes, se inició la búsqueda de una página web que utilice un algoritmo de hash en sus contraseñas, como MD5, SHA1 o incluso SHA256. Para ello, se utilizó la página PublicWWW buscando a través de 'md5'. Finalmente, se obtuvo la siguiente página web que se utilizará a lo largo del laboratorio.

La pagina obtenida es https://www.osmosis.org/ que posee un inicio de sesión y registro de sesión, fundamentales para el laboratorio.

Se partio registrandose en el sitio web para identifica el algoritmo hash utlizado en la pagina y corroborar la busqueda realizada por PublicWWW. Para ello se utilizo la pagina Tempmail para crear un correo electronico temporal y no ocupar datos sensibles y personales.

Figura 1: Creación de correo electronico temporal.

Una vez generado el correo electronico se inicio la creación de la cuenta en la pagina de Osmosis, utilizando la contraseña: ***Pepito1212@***.

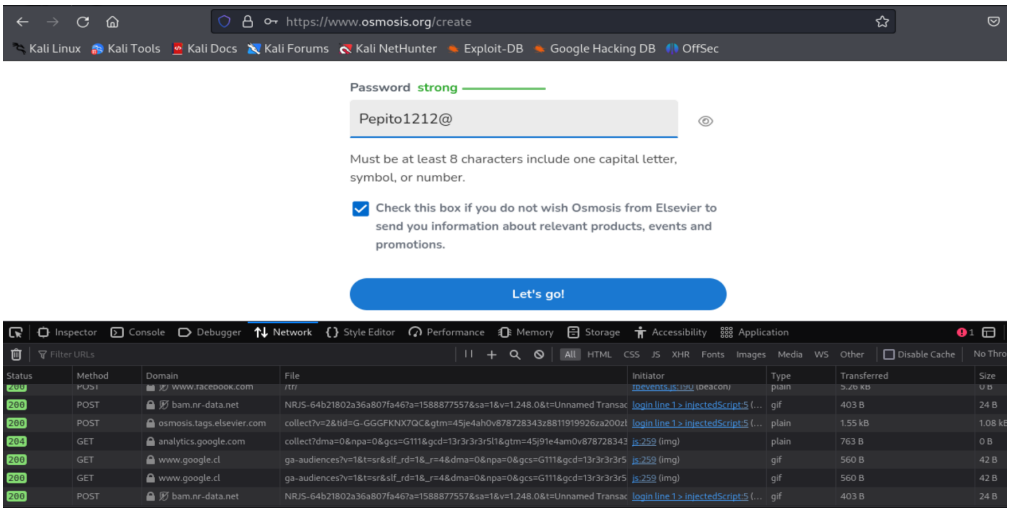

Figura 2: Creación de la cuenta en Osmosis.

Para identificar el algoritmo, se utilizó la opción 'Inspeccionar Elemento' en la página web, haciendo clic derecho sobre ella. Luego, se dirigió al apartado de "Network". Una vez allí, se hizo clic en el botón celeste 'Let's go!'. Tras realizar esta acción, apareció lo siguiente.

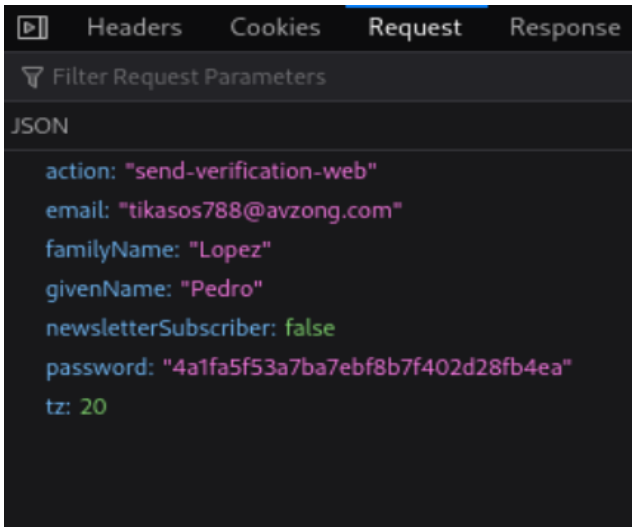

Figura 3: Request Register.

Como se aprecia en la imagen de la figura 3, **la contraseña no se encuentra en texto plano, sino que está hasheada**. Para poder identificar el tipo de hash, se recurrió a una página especializada en identificar mensajes codificados. Para ello, se utilizó la página 'https://www.dcode.fr'.

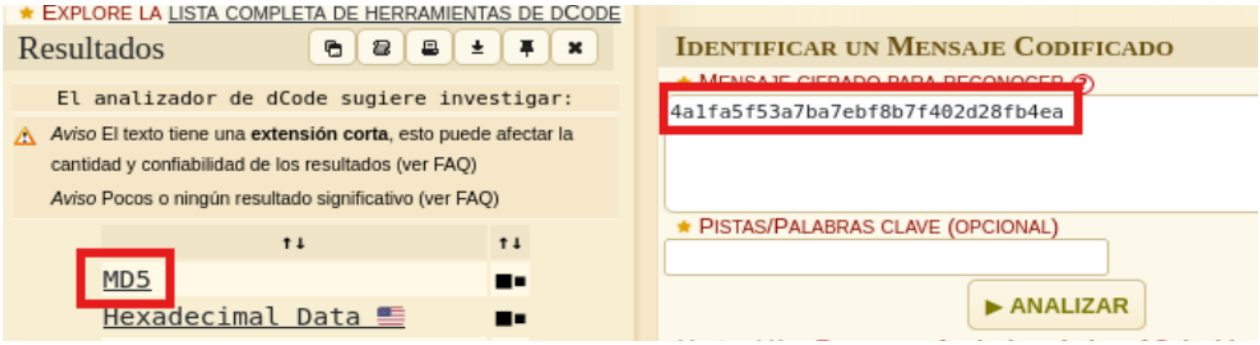

Figura 4: Algoritmo MD5.

Tal como se muestra en la figura anterior, la pagina de Osmosis utiliza del ***algoritmo MD5*** en la seguridad de las contraseñas de su sitio web.

2.2. Identifica el algoritmo de hash utilizado al momento de iniciar sesión

Una vez realizado el registro en la página Osmosis, se procedió a iniciar sesión para poder identificar el algoritmo utilizado durante el proceso de inicio de sesión.

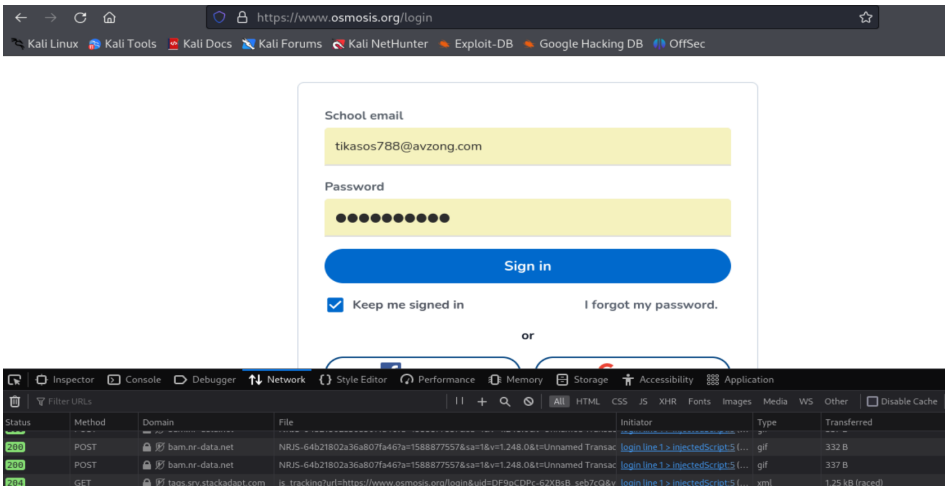

Figura 5: Inicio sesión Osmosis.

Cabe destacar que, al momento de iniciar sesión, se realizaron los mismos pasos que durante el registro, incluyendo la acción de hacer clic en el botón, tal como se hizo al inspeccionar el elemento. Por último, el request generado fue el siguiente.

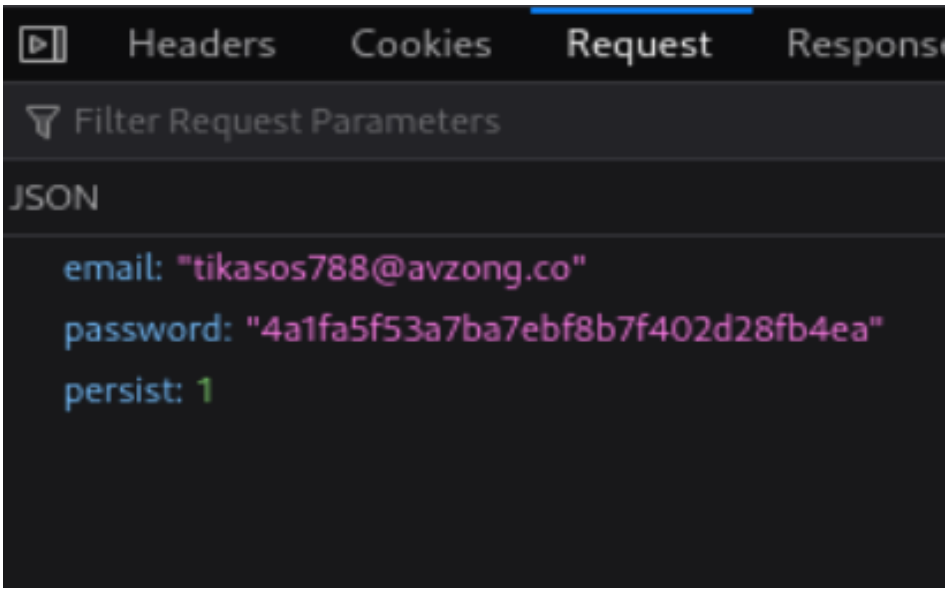

Figura 6: Request Login.

En la figura 6, se pueden apreciar dos elementos importantes. El primero es que al correo le falta una 'm'; esto se debe a que, si se colocaba el correo completo, la página saltaba directamente al dashboard del usuario y se borraba el POST del network del Login, perdiendo así el Request. En segundo lugar, se confirma que efectivamente utiliza MD5 en sus contraseñas. Sin embargo, surgen las siguientes interrogantes: ¿Es un único algoritmo o son varios? ¿Estará mezclado con algún token o será único?

Para poder contestar las preguntas anteriores, se dirigió a una página web que genera MD5 para así poder colocar la contraseña y comparar los resultados.

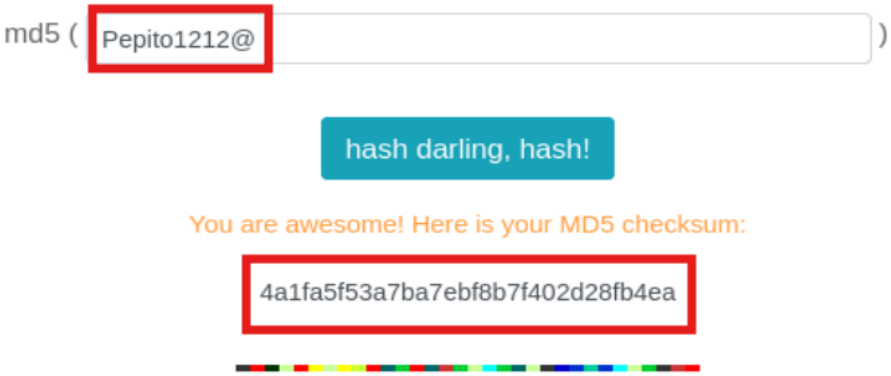

Figura 7: Generador Algoritmo MD5.

La figura 7 responde las dudas anteriores. El hash de la contraseña es idéntico al mostrado en la figura 3 y la figura 6; por lo tanto, se concluye que la página Osmosis, en sus contraseñas, utiliza un **único** algoritmo de hash llamado **MD5**.

2.3. Genera el hash de la contraseña desde la consola del navegador

Como ya se conoce que la página utiliza MD5, se procede a generar el hash utilizando la función md5 dentro de la consola del navegador.

A la hora de querer utilizar la función md5, lanzaba el error de que no estaba definida, esto se debe a que no está definido en la consola (es decir, la página no lo incluye como parte de su código), para ello se agrego la biblioteca manualmente. Para esto, se cargo **CryptoJS** escribiendo el siguiente código en la consola del navegador.

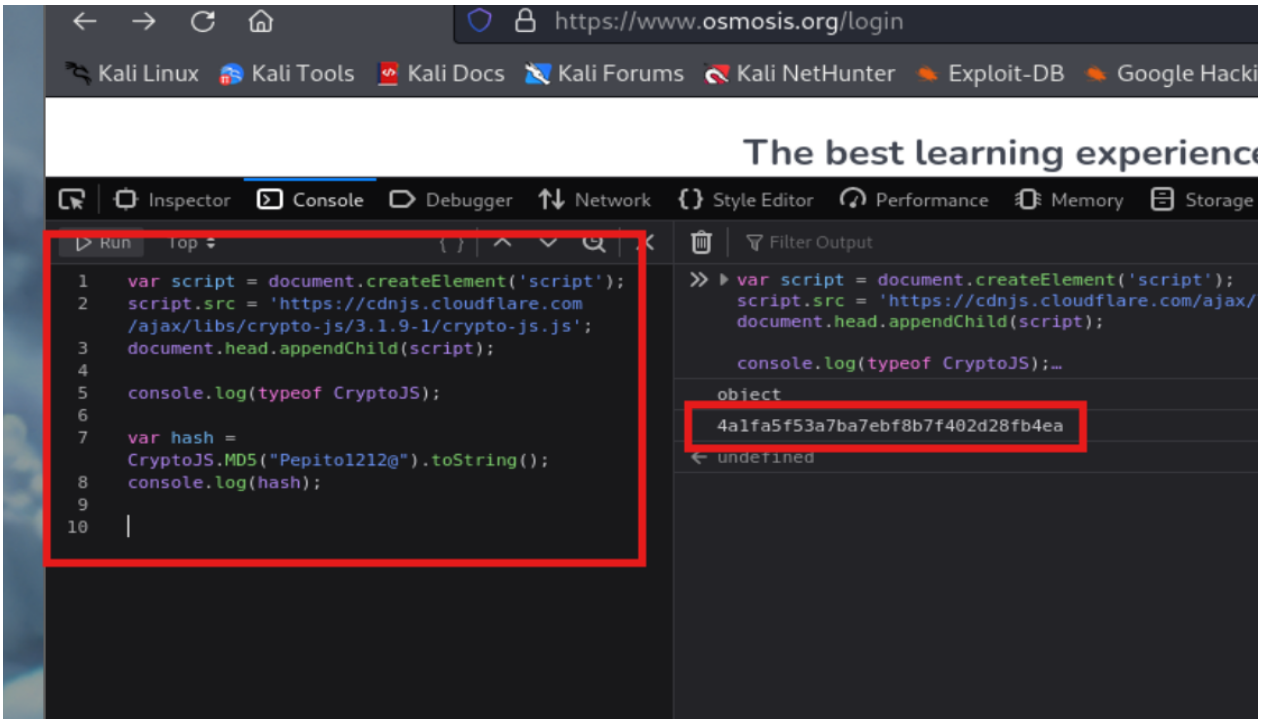

Figura 8: Biblioteca manual CryptoJS.

Al añadir la bibliteca manual obtenemos el mismo codigo cifrado por **MD5** de la contraseña '**Pepito1212@**'.

2.4. Intercepta el tráfico login con BurpSuite

Para poder interceptar el tráfico de Mozilla Firefox en BurpSuite, se ingresó un certificado (instalado en el laboratorio anterior) y se modificó el parámetro del proxy a 127.0.0.1 con el puerto 8080. Además, se activó la comunicación proxy tal como se muestra en la siguiente figura.

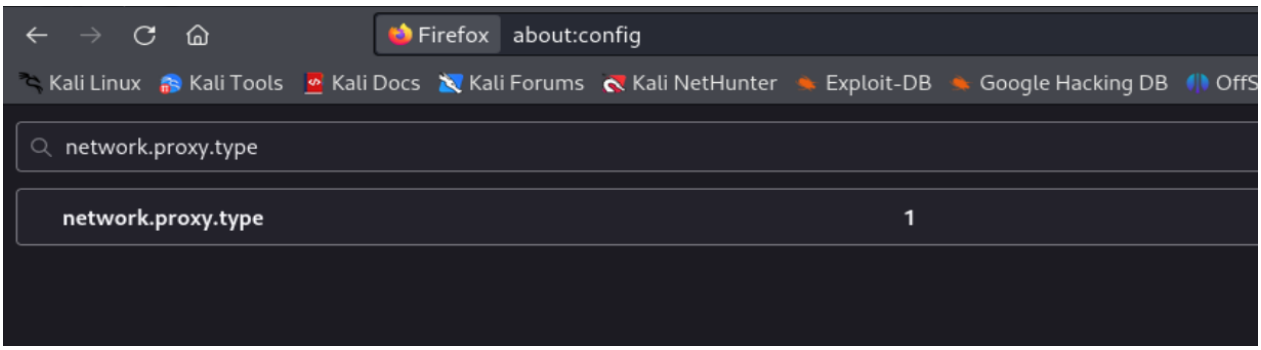

Figura 9: Configuración comunicación proxy.

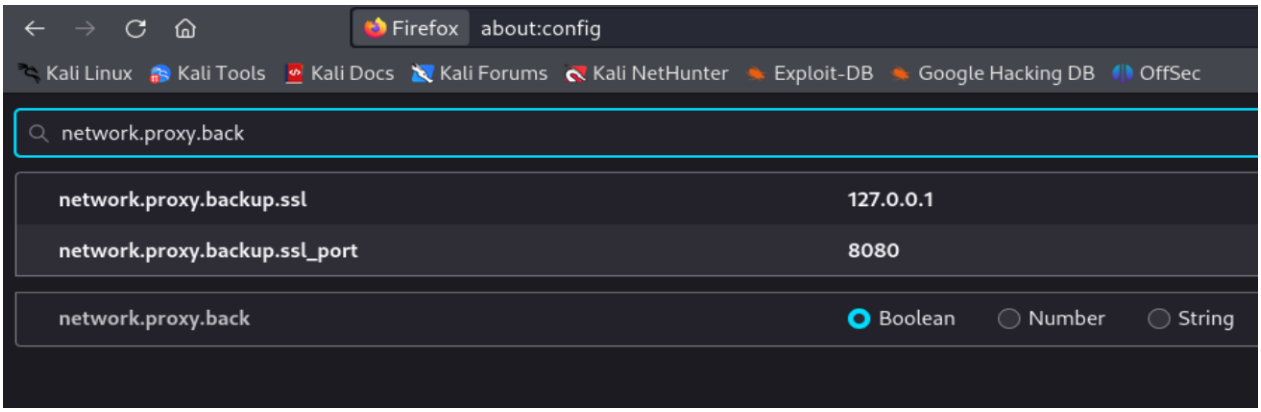

Figura 10: Configuración del proxy.

Una vez configurado lo anterior, se inició el proceso de Login en la página Osmosis, interceptando el tráfico en BurpSuite tal como se muestrá en la siguiente imagen.

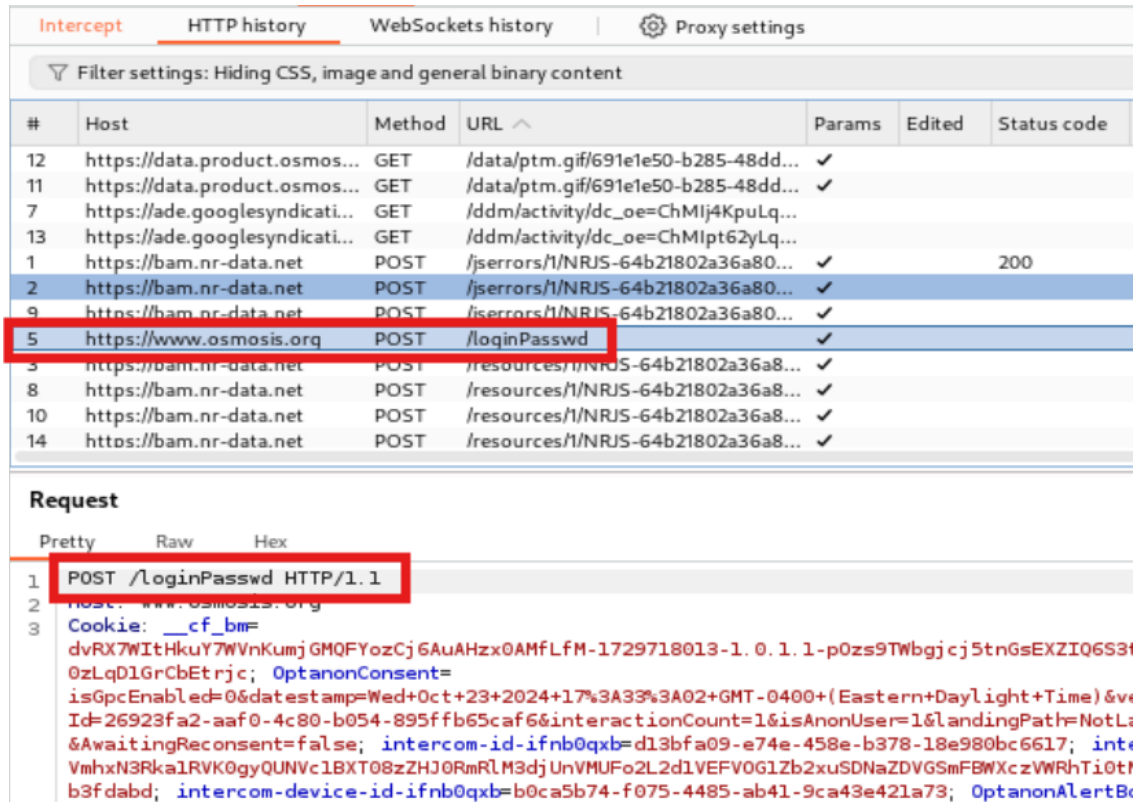

Figura 11: LoginPasswd interceptado con exitó.

Al interceptar el login se obtiene lo siguiente, identico a lo que se obtenia en el primer apartado.

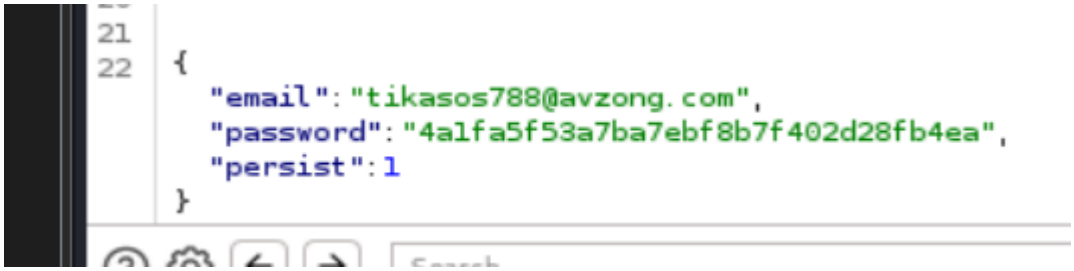

Figura 12: Mensaje interceptado del login.

2.5. Realiza el intento de login

Una vez interceptado el tráfico, se implementó una técnica llamada PassTheHash. Como su nombre lo indica, consiste en pasar el hash directamente para burlar la autenticación y poder iniciar sesión en la página exitosamente. En términos más técnicos, es una técnica de piratería que permite a un atacante autenticarse en un servidor o servicio remoto utilizando el valor hash de la contraseña en lugar de la contraseña en texto plano.

Para ello se comenzo iniciando sesión con una contraseña incorrecta, utilizando '**Pepito1213@**' e interceptandola en BurpSuite obteniendo un hash diferente al anterior.

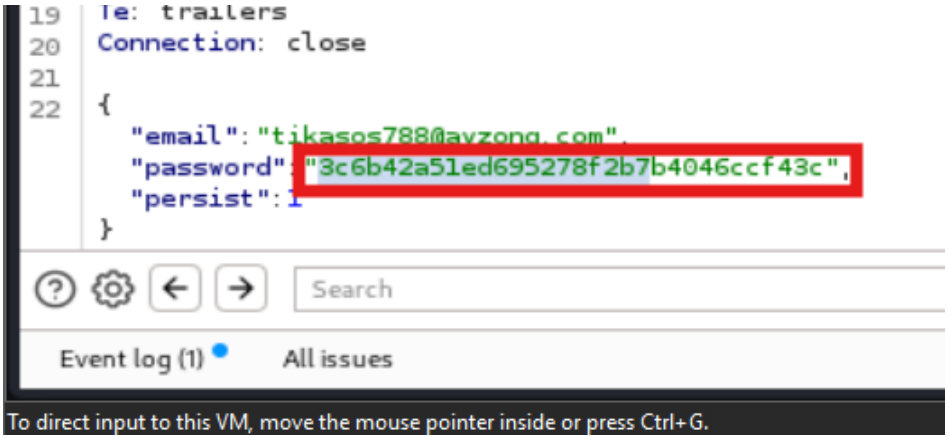

Figura 13: Intercepción Login incorrecto.

Como se aprecia en la imagen 13, el hash de la contraseña es distinto al inicial, ya que son contraseñas diferentes.

Para realizar el PassTheHash, cambiaremos el hash de la password de la figura 13 con el hash de la contraseña correcta.

En la pestaña Inspector se pueden observar los detalles de la solicitud, como los **parámetros de formulario**, **headers**, y datos relevantes como el **username** y **password**, siendo posible modificar cualquier parte de la solicitud. Para modificar el campo de la contraseña, se debe localizar el parámetro correspondiente y colocar el hash de la contraseña correcta, tal como se muestra en la figura 14.

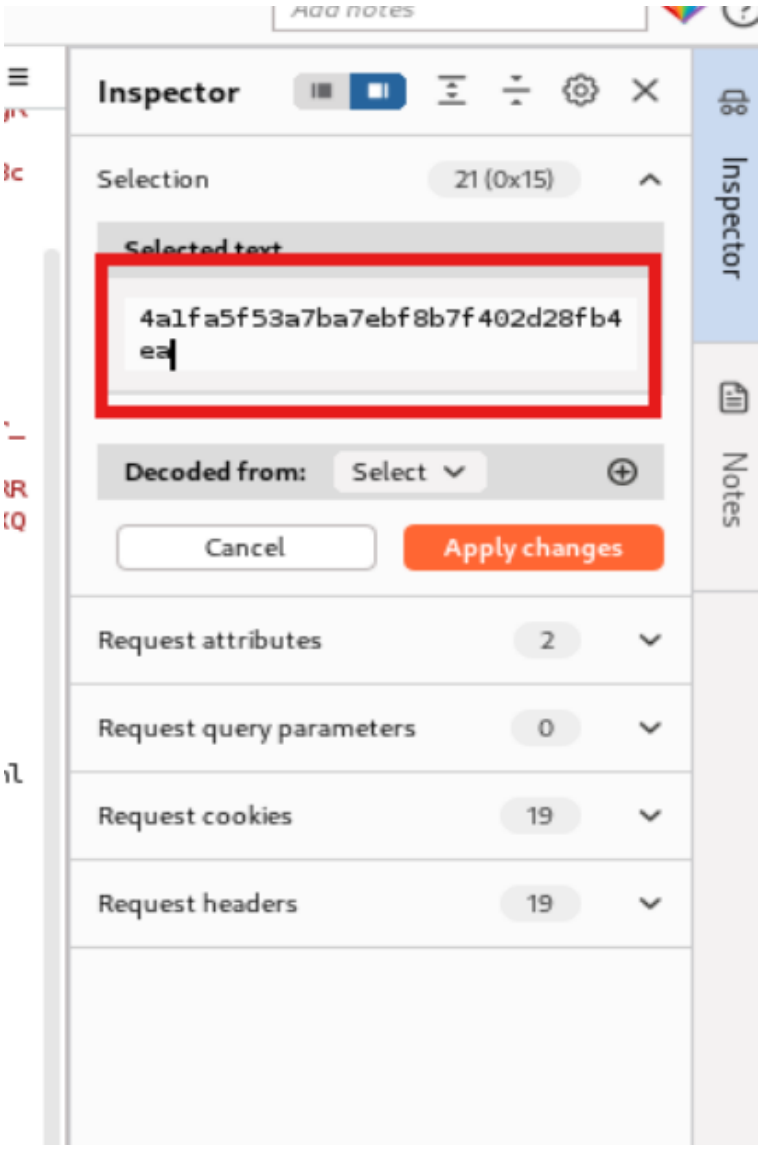

Figura 14: Modificar la solicitud de inicio sesión con el hash correcto.

Una vez modificada el hash de la contraseña por el hash correcto, se aplican los cambios, y luego se corrobora en el mensaje interceptado.

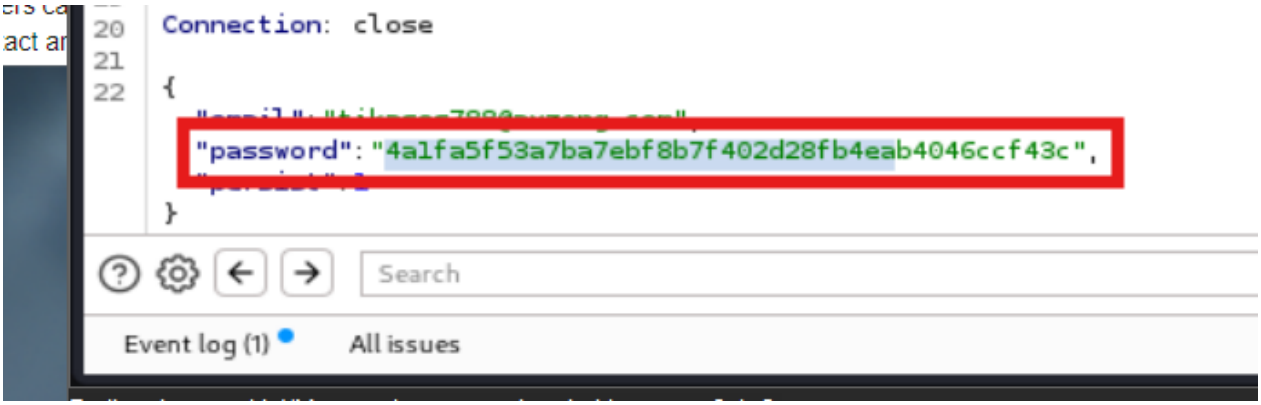

Figura 15: Mensaje cambiado por el hash correcto.

Una vez corroborada la modificación, se presiona el botón Forward, tal como lo muestra la imagen a continuación, para reenviar la solicitud modificada al servidor.

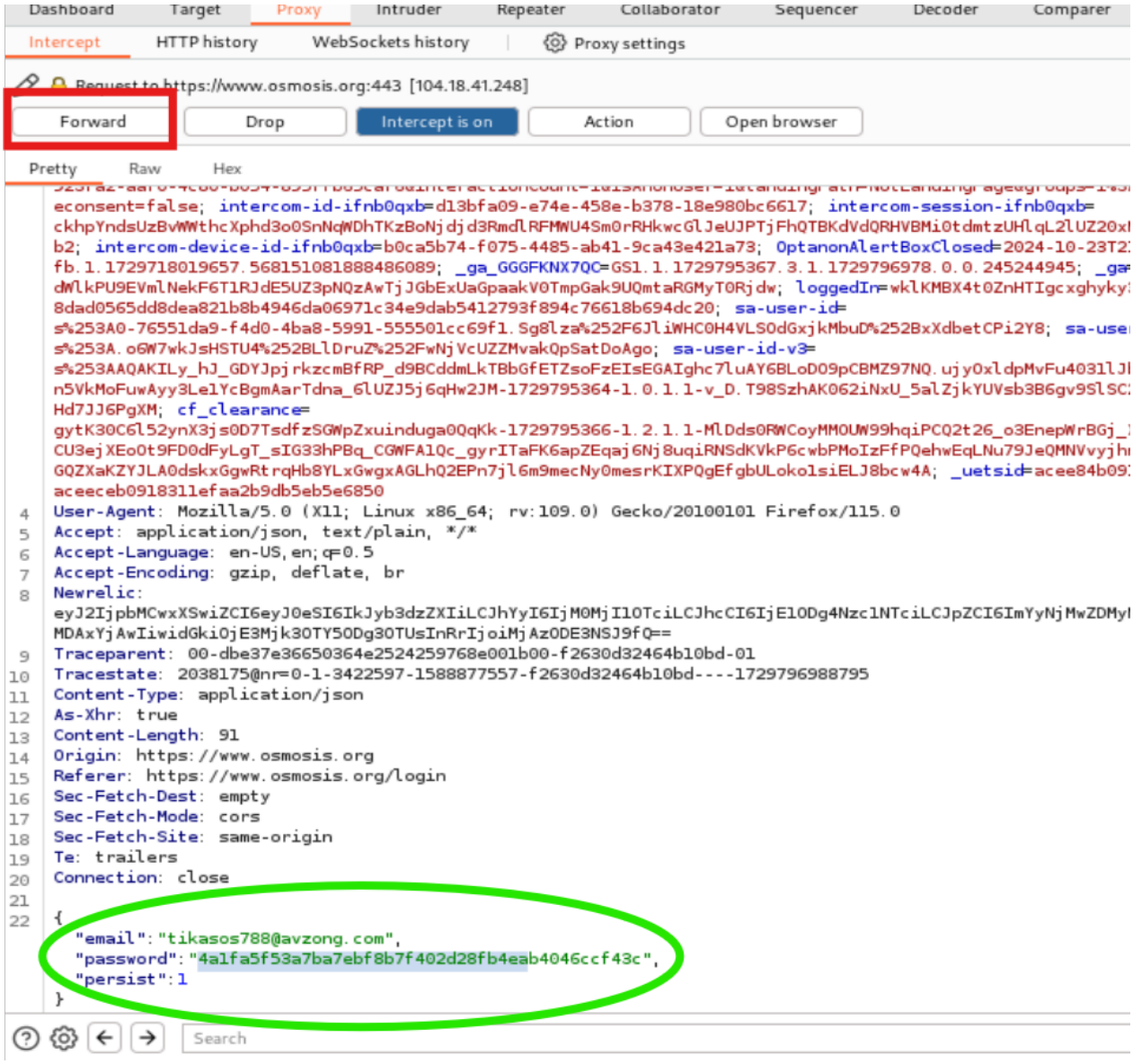

Figura 16: Reenviar la solicitud modificada al servidor.

Una vez realizado el clic en el botón Forward, se redirige a la página web de Osmosis y se obtiene lo siguiente.

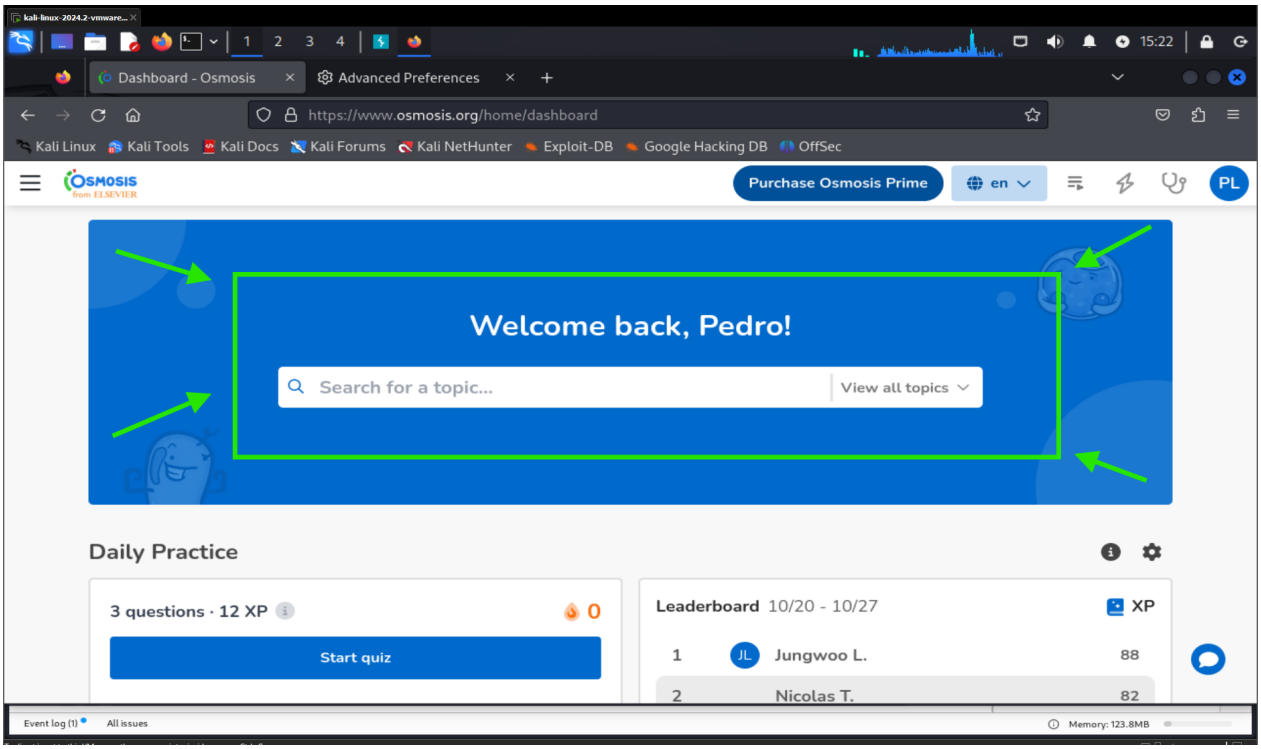

Figura 17: Inicio sesión exitoso.

Tal como se muestra en la imagen 17, se logró realizar exitosamente el inicio de sesión mediante la técnica PassTheHash.

2.6. Identifica las políticas de privacidad o seguridad

Para comenzar, la página Osmosis es administrada por la empresa Elsevier, por lo tanto, es esta entidad la que regula las políticas de privacidad que maneja la empresa. Dichas políticas se encuentran al final de la página web de Osmosis (https://www.elsevier.com/legal/privacy-policy).

La empresa explica que recoge información personal, como datos de contacto, de registro y de uso, para fines operativos, de análisis y marketing. Además, se pueden recopilar datos sobre la navegación y preferencias para mejorar la experiencia del usuario.

Elsevier asegura que se toman medidas para proteger la información, cumpliendo con leyes de protección de datos como el GDPR, y ofrece opciones para gestionar la privacidad y uso de los datos.

Por otro lado en termino de la seguridad, la contraseña debe tener al menos 8 caracteres e incluir una letra mayúscula, un símbolo o un número. Este tipo de política de seguridad asegura que las contraseñas tengan un nivel de complejidad básico, lo que dificulta que sean adivinadas o vulnerables a ataques de fuerza bruta.

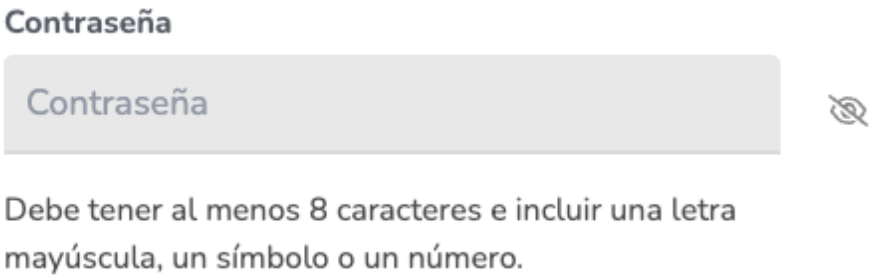

Figura 18: Complejidad mínima de la contraseña.

Esta es una medida común para proteger cuentas de usuario, asegurando que las contraseñas sean más resistentes a ataques automatizados o intentos de hackeo.

2.7. Demuestra 4 conclusiones sobre la seguridad

Respecto al desarrollo del laboratorio, se llegaron a cuatro conclusiones sobre la seguridad de la página web.

1. **MD5 es un algoritmo obsoleto** debido a sus vulnerabilidades criptográficas y su susceptibilidad a ataques, lo que ha llevado a su reemplazo en casi todas las aplicaciones de seguridad moderna. Hoy en día, se recomiendan algoritmos más seguros como SHA-256 para la mayoría de los usos criptográficos, por lo tanto es recomendable migrar a algoritmos más seguros.

2. La página web Osmosis es **vulnerable mediante un ataque de Pass-the-Hash**, en el cual se logró autenticarse utilizando un hash de contraseña 'robada', sin necesidad de conocer la contraseña original. Este incidente pone en evidencia fallas en la gestión y protección de los hashes de contraseñas, así como en los mecanismos de autenticación empleados en la plataforma, comprometiendo la integridad de la página.

3. El sitio web **no implementa ninguna forma de autenticación multifactor (MFA)**. Esta es una capa de seguridad esencial para mitigar el riesgo de ataques basados en contraseñas o hashes comprometidos. Al añadir un segundo factor de autenticación, se reduce considerablemente la posibilidad de que un atacante acceda a una cuenta, incluso si logra obtener las credenciales.

4. Finalmente, se determinó que el hash de las contraseñas se está generando en el **navegador del cliente antes de ser enviado al servidor** (*según lo observado mediante BurpSuite*). Aunque esta técnica podría ofrecer una protección básica al evitar la transmisión directa de contraseñas en texto plano, también incrementa la vulnerabilidad del sitio, facilitando los ataques en la misma web.