INSTITUTO TECNOLÓGICO SUPERIOR DE CHICONTEPEC
Ingeniería en sistemas computacionales

MATERIA:
Programación Web.

ALUMNO:

José Luis De La Cruz Cruz.

Docente:

Ing. Efrén Flores Cruz.

Tema:

Reporte de práctica.

Fecha entrega:

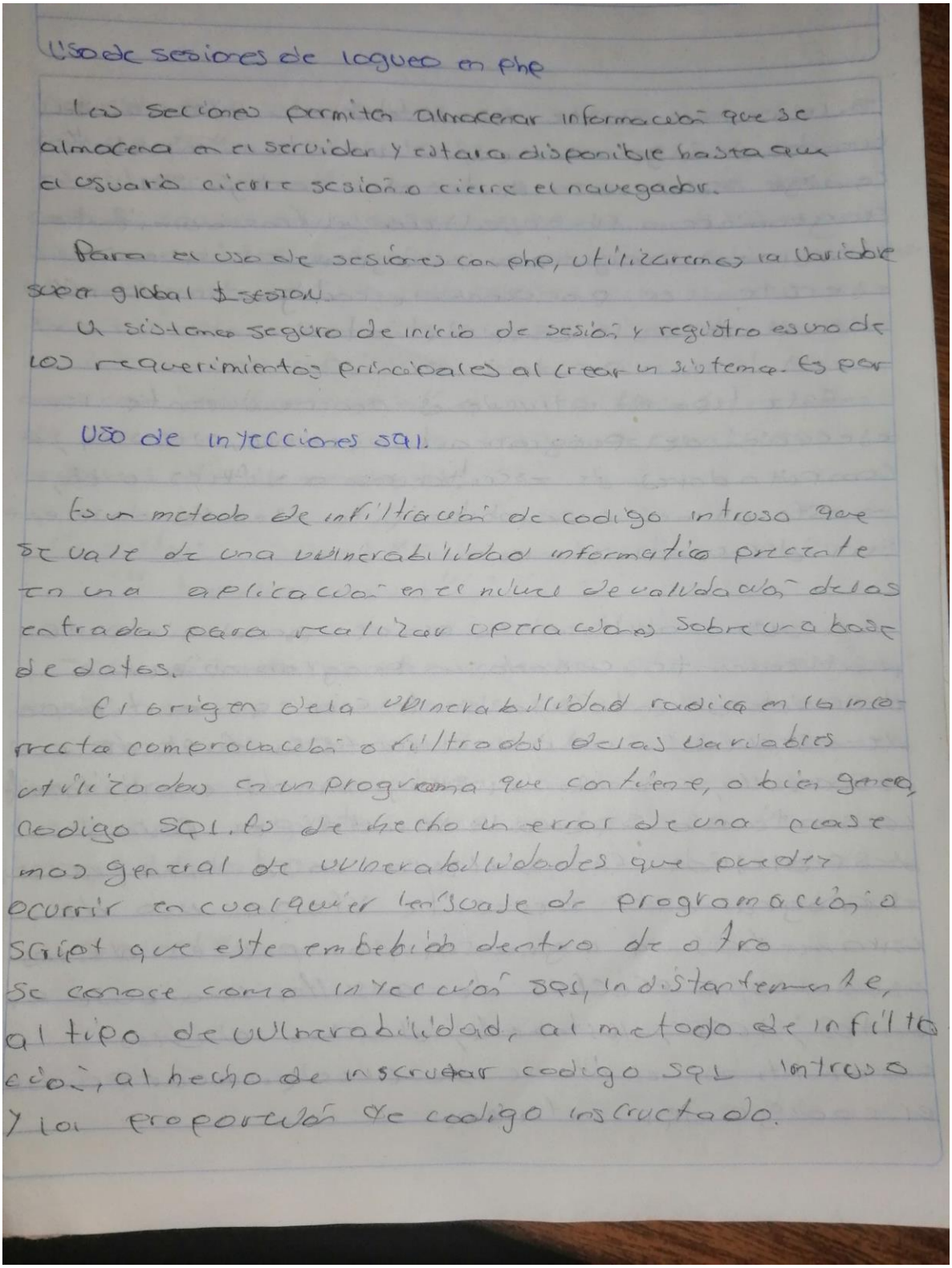
29/04/2020

Índice

Introducción

En el presente documento se encuentra la evidencia de los temas desarrollados por el alumno, de la materia de programación web de la cual consiste de los siguientes temas, uso de logueo, es la parte donde se desarrolla las interfaces delas cuales se les muestra al usuario donde es la interface gráfica la cual se desarrolla, así como las inyecciones que se le puede hacer a las bases de datos cuando se encuentran en ejecución al mismo tiempo explica dónde y cómo podemos ver que se está realizando una inyección, una inyección en una base de datos consiste en un código que es insertado por otro usuario, este se puede realizar desde el servidor en el que se encuentra alojado los archivos de ejecución, al mismo tiempo podremos observar que se pueden realizar encriptaciones en las bases de datos las cuales ayudan a mantener un tipo de seguridad en las bases de datos y así no cualquiera pueda observar por ejemplo el usuario y contraseña del usuario, ya que este utiliza barios bits para poder encriptar una usuario y contraseña y solo se encripta y no se puede realizar la desincriptacion.

Desarrollo



Uso de sesiones de logueo en php

Las secciones permiten almacenar información que se almacena en el servidor y estara disponible hasta que el usuario cierre sesión o cierre el navegador.

Para el uso de sesiones con php, utilizaremos la variable super global $_SESION.

Un sistema seguro de inicio de sesión y registro es uno de los requerimientos principales al crear un sistema. Es por

Uso de inyecciones sql.

Es un metodo de infiltración de codigo intruso que se vale de una vulnerabilidad informatica presente en una aplicación en el nivel de validación de las entradas para realizar operaciones sobre una base de datos.

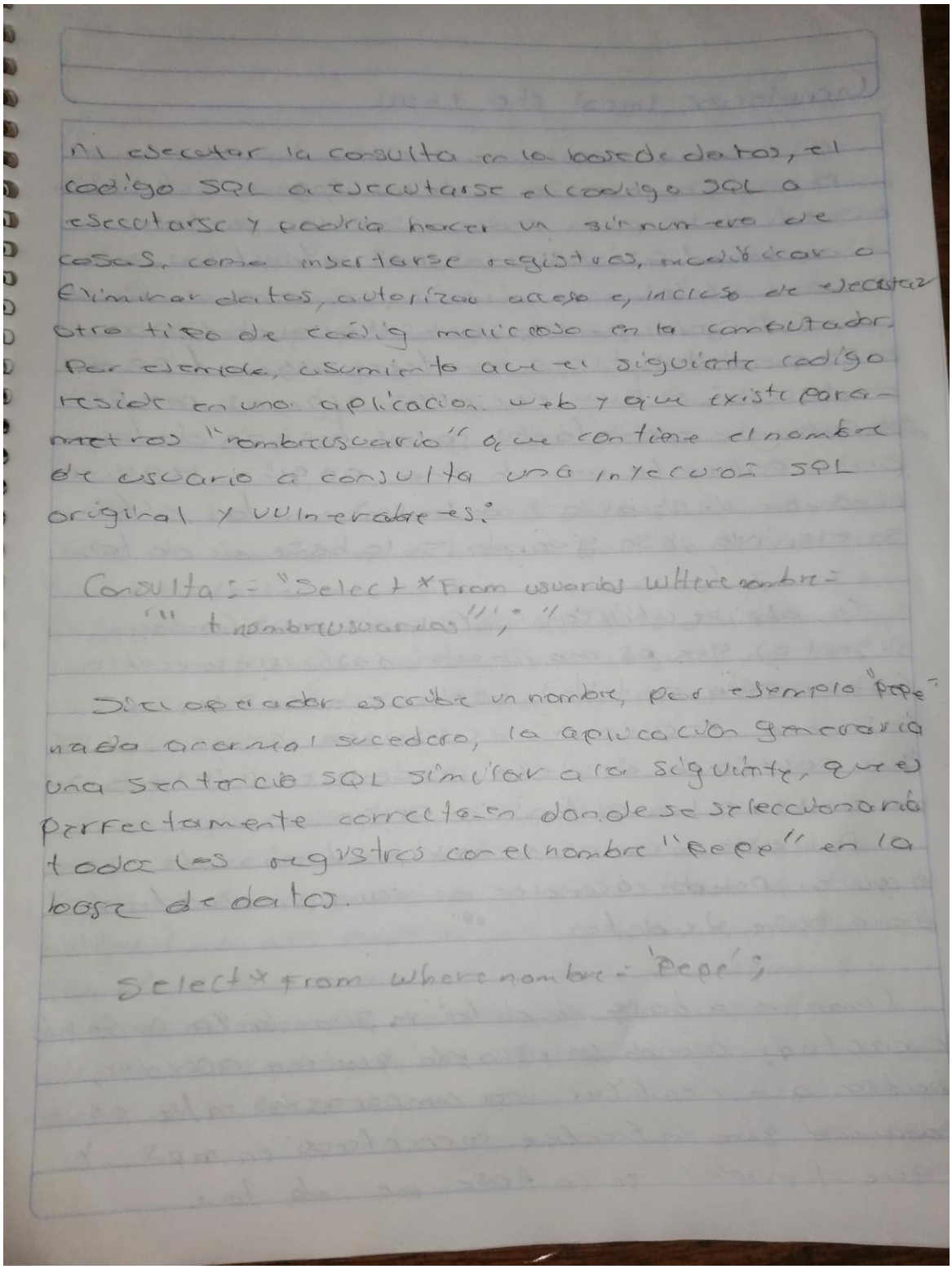
El origen de la vulnerabilidad radica en la incorrecta comprobación o filtrado de las variables utilizadas en un programa que contiene, o bien genera, codigo SQL. Es de hecho un error de una clase mas general de vulnerabilidades que puede ocurrir en cualquier lenguaje de programación o Script que este embebido dentro de otro

Se conoce como inyección SQL, indistintamente, al tipo de vulnerabilidad, al metodo de infiltración, al hecho de inscrutar codigo SQL intruso y la porción de codigo instructado.

Se dice que existe cuando, de alguna manera, se inserta o "inyecta" código SQL invasor dentro del código SQL programado a fin de alterar el funcionamiento normal del programa y lograr así que se ejecute la porción de código "invasor" inscrustado, en la base de datos.

Este tipo de instrucciones ocurre durante la ejecución del programa vulnerable, ya sea, en computadoras de escritorio o sitios web, en este último caso obviamente ejecutándose en el servidor que los aloja.

La vulnerabilidad se puede producir automáticamente cuando un programa "arma descuidadamente" una sentencia SQL en tiempo de ejecución, o bien durante la fase de desarrollo, cuando el programador explícitamente la sentencia SQL a ejecutar en forma desprotegida. En cualquier caso siempre que el programador necesite y haga uso de parametros a ingresar por partes de los usuarios, a efectos de consultar una base de datos; ya que, justamente, dentro de estos parametros es donde se puede incorporar el codigo SQL intruso.



Al ejecutar la consulta en la base de datos, el código SQL a ejecutarse el código SQL a ejecutarse y podría hacer un sin número de cosas, como insertarse registros, modificar o eliminar datos, autorizar acceso e, incluso de ejecutar otro tipo de código malicioso en la computador. Por ejemplo, asumiendo que el siguiente código reside en una aplicación web y que existe parámetros "nombreusuario" que contiene el nombre de usuario a consulta una inyección SQL original y vulnerable es:

Consulta := "Select * From usuarios Where nombre = " " + nombreusuarios + " ' " ";

Si el operador escribe un nombre, por ejemplo "Pepe" nada anormal sucedería, la aplicación generaría una sentencia SQL similar a la siguiente, que es perfectamente correcta en donde se seleccionará todos los registros con el nombre "Pepe" en la base de datos.

Select * From Where nombre = 'Pepe';

Encriptación (md5) php y mysql

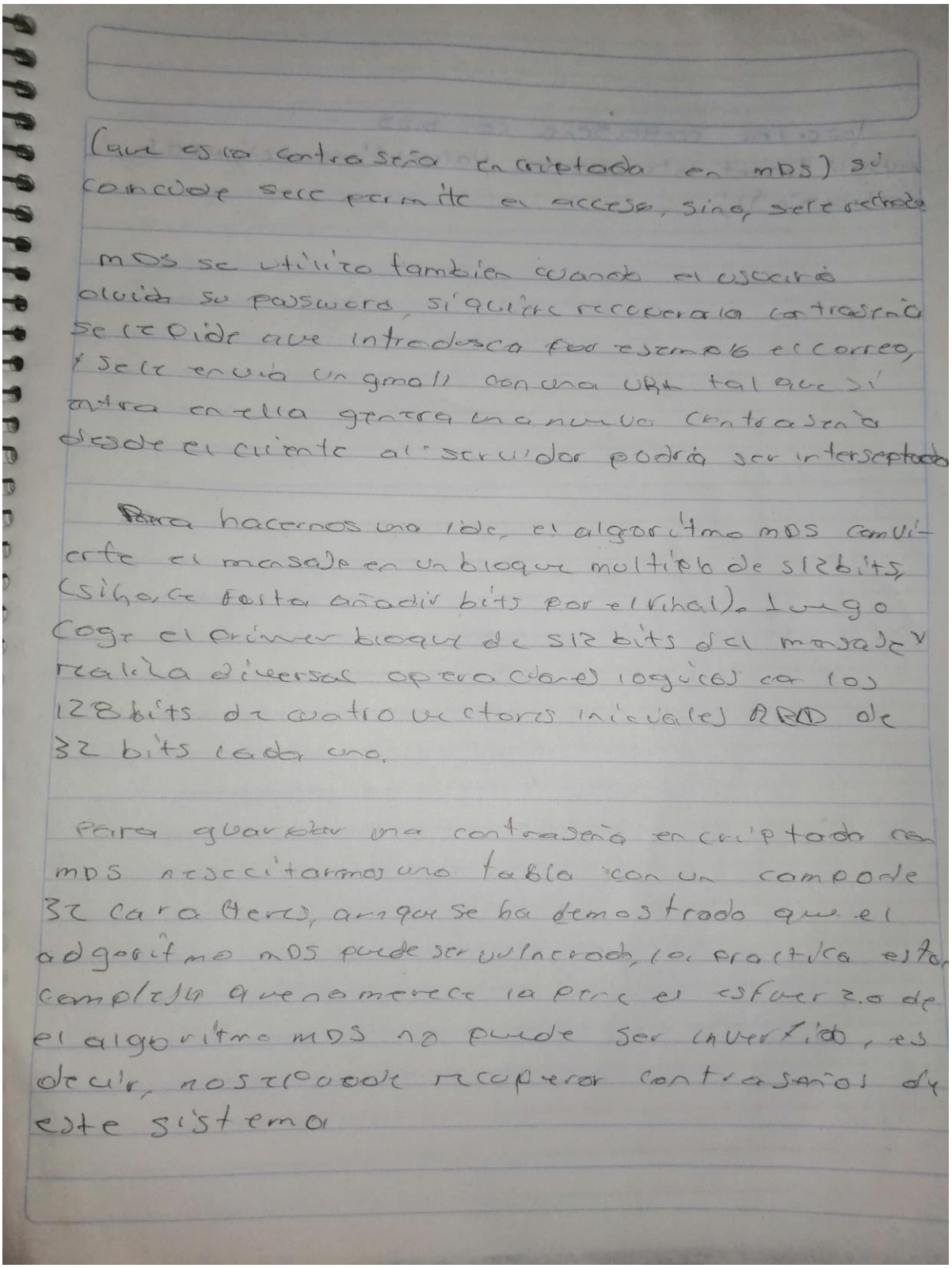
Hoy en día la mayoría de las páginas web utilizan bases de datos para poder desarrollar portales dinámicos y así hacerlos más atractivos a la vez que útiles.

Esta información que se guarda en la base de datos tiene que tener algún tipo de protección. Es que por ellos algunos campos se guardan encriptados en las base de datos, principalmente cuando una página requiere el nombre de usuario y contraseña; esta última se encripta y se guarda en la base de datos.

En php se utiliza la función MD5 (message digest 5) que es una función hash irreversible (de un solo sentido), es decir, encripta el password ingresado por el usuario y es posible que partiendo desde la cadena encriptada se devuelve a la contraseña origen.

Por esto mismo no hay problema de que alguien pueda acceder al campo encriptado de la base de datos.

Como en la base de datos se guarda la contraseña encriptada cuando un usuario quiera acceder, habrá que realizar una comparación entre el password que introduce encriptado en MD5 y lo que tenemos en la base de datos.

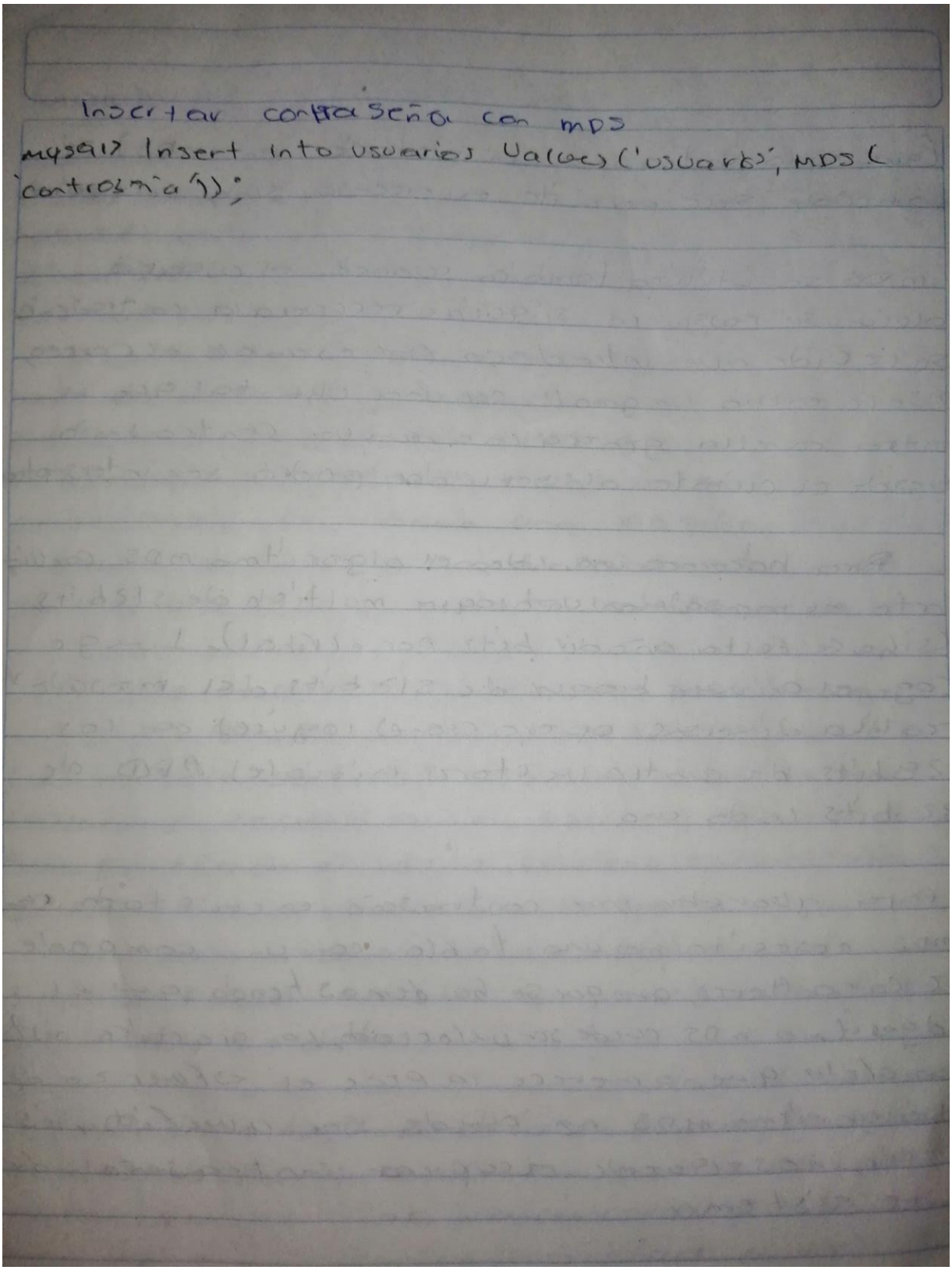


(que es la contraseña encriptada en MD5) se concede o se permite el acceso, sino, se rechaza.

MD5 se utiliza también cuando el usuario olvida su password, si quiere recuperarla contraseña se le pide que introduzca por ejemplo el correo, y se le envía un gmail con una URL tal que si entra en ella genera una nueva contraseña desde el cliente al servidor podría ser interceptado

Para hacernos una idea, el algoritmo MD5 convierte el mensaje en un bloque multiplo de 512 bits, (si hace falta añadir bits por el final). Luego coge el primer bloque de 512 bits del mensaje y realiza diversas operaciones lógicas con los 128 bits de cuatro vectores iniciales MD de 32 bits cada uno.

Para guardar una contraseña encriptada con MD5 necesitaríamos una tabla con un campo de 32 caracteres, aunque se ha demostrado que el algoritmo MD5 puede ser vulnerado, lo cual en la práctica es complejo y no merece la pena el esfuerzo ya que el algoritmo MD5 no puede ser invertido, es decir, nos resultaría imposible recuperar contraseñas de este sistema



Insertar contraseña con MD5

```
mysql> Insert into usuarios Values('usuario', MD5('contraseña'));
```

Conclusión

Los temas desarrolla dos ayudan a comprender al alumno de qué manera se pueden realizar y que tipos de seguridad se utilizan para mantener un código y las bases de dato que aloja una página web ya que es muy importante mantener los usuarios y contraseñas así como datos personales de los usuarios en un modo que los demás no puedan acceder a ellos para así no ser vulnerables a implantaciones de identidad.