ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

UTILIZACIÓN DE UN ALGORITMO DE ENCRIPTACIÓN APLICADO A LA COMUNICACIÓN HUMANO-ROBOT

WILIAN XAVIER SÁNCHEZ LABRE

Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo, presentado ante el Instituto de Posgrado y Educación Continua de la ESPOCH, como requisito parcial para la obtención del grado de:

MAGÍSTER EN SEGURIDAD TELEMÁTICA.

RIOBAMBA - ECUADOR

Octubre 2017

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

CERTIFICACIÓN:

EL TRIBUNAL DE TRABAJO DE TITULACIÓN CERTIFICA QUE:

El **Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo**, titulado "**UTILIZACIÓN DE UN ALGORITMO DE ENCRIPTACIÓN APLICADO A LA COMUNICACIÓN HUMANO-ROBOT**", de responsabilidad del Sr. Wilian Xavier Sánchez Labre ha sido prolijamente revisado y se autoriza su presentación.

Tribunal:

Ing. Wilson Armando Zuñiga Vinueza; M.Sc.	
PRESIDENTE	**FIRMA**
Ing. José Luis Morales Gordón; M.Sc.	
DIRECTOR	**FIRMA**
Ing. Cristhy Nataly Jiménez Granizo; M.Sc.	
MIEMBRO	**FIRMA**
Ing. Veloz Remache Germania del Rocío; M.Sc.	
MIEMBRO	**FIRMA**

Riobamba, Octubre 2017

DERECHOS INTELECTUALES

Yo, Wilian Xavier Sánchez Labre, declaro que soy responsable de las ideas, doctrinas y resultados expuestos en el **Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo**, y que el patrimonio intelectual generado por la misma pertenece exclusivamente a la Escuela Superior Politécnica de Chimborazo.

_____

Wilian Xavier Sánchez Labre

DEDICATORIA

Este trabajo va dedicado a mi esposa Ximena por todo el amor y el apoyo brindado, logrando así cumplir todas mis metas y sueños. A mis hijos Emily, Eliana y Eliel quienes son los pilares fundamentales en cumplir una nueva meta profesional.

Wilian

AGRADECIMIENTO

A Dios por permitirme seguir junto a mi familia. Al Ing. MsC. José Luis Morales por compartir sus conocimientos para el desarrollo de este trabajo de investigación, a la Ing. MsC. Cristhy Jiménez e Ing. MsC. Germanía Veloz por el apoyo dedicado para cumplir el objetivo de esta nueva meta alcanzada.

Wilian

ÍNDICE

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

ÍNDICE DE ANEXOS

RESUMEN

Se implementó un algoritmo de encriptación aplicado a la comunicación humano–robot para garantizar la seguridad de información transmitida, para ayudar a proteger la transferencia de información por medio de una aplicación de software, utilizando el algoritmo criptográfico seleccionado como método de encriptación de datos fue el algoritmo asimétrico HASH y dentro de éste se optó por el método SHA-256; logrando mitigar ataques a la transmisión de datos. Se ha desarrollado una aplicación de software con el lenguaje de programación PHP y arduino para la simulación de los movimientos izquierda y derecha por medio de un dispositivo electrónico arduino, ethernetshield y microservo induciéndole un ataque básico a la transferencia de la información para su respectiva comunicación. La elaboración del prototipo demuestra un aumento significativo de los niveles de seguridad en un 1.8 veces mejor con respecto a la implementación del software en la transferencia de datos mediante la encriptación de la información con el algoritmo SHA-256. Con la utilización de algoritmos criptográficos que existen en la actualidad se concluyó que se puede mejorar el nivel de seguridad de la información que será transmitida por cualquier medio de comunicación y asegurar la integridad de la transferencia de datos con la creación de certificados de seguridad SSL. Se recomienda realizar sistemas de interacción humano-robot mediante la utilización de voz como medio de comunicación y así proporcionar la posibilidad de una comunicación más natural con los robots como trabajo futuro.

Palabras clave: <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>, <TELECOMUNICACIONES> <ENCRIPTACIÓN> <ALGORITMO CRIPTOGRÁFICO> <ALGORITMO CRIPTOGRÁFICO SHA-256/HASH> <ALGORITMO DE RESUMEN O HASH><CLAVE PÚBLICA O ASIMÉTRICO> <CLAVE SECRETA O SIMÉTRICA>

SUMMARY

Implemented an encryption algorithm applied to human-robot communication to ensure the security of transmitted information, to help protect the transfer of information through a software application, using the algorithm Cryptographic selected as a data encryption method was the asymmetric hash algorithm and within it was opted for the SHA-256 method; Managing to mitigate attacks on data transmission. It has developed a software application with the programming language PHP and Arduino for the simulation of the movements left and right by means of an electronic device Arduino, Ethernetshield and Micro servo inducing a basic attack on the Transfer of information for their respective communication. Prototype development demonstrates a significant increase in security levels by 1.8 times better with respect to software implementation in data transfer by encrypting information with the SHA – 256 algorithms. With the use of cryptographic algorithms that exist today it was concluded that one can improve the level of security of the information that will be transmitted by any means of communication and ensure the integrity of the data transfer with the Creating SSL security certificates. It is recommended to perform systems of human-robot interaction through the use of voice as a means of communication and thus provide the possibility of a more natural communication with robots as future work.

Keywords: <TECHNOLOGY AND ENGINEERING SCIENCES>, <TELECOMMUNICATIONS>, <ENCRYPTION>, <CRYPTOGRAPHIC ALGORITHM>, <SHA – 256/HASH CRYPTOGRAPHIC ALGORITHM>, <DIGEST OR HASH ALGORITHM>, <PUBLIC OR ASYMMETRIC KEY>, <SECRET OR SYMMETRIC KEY>.

CAPÍTULO I

1. INTRODUCCIÓN

Mediante el avance de las tecnologías informáticas y con ello el uso del internet a un nivel mundial para el acceso a la información ésta debe ser primordial conocer cada uno de los recursos de las empresas que necesitan protección para así poder controlar el acceso por niveles a cada uno de sus sistemas informáticos.

Un sistema informático puede ser altamente seguro, si se logra identificar las respectivas amenazas hacia la información y así prever el curso de acción de un atacante donde la seguridad significa proteger recursos valiosos, que pertenecen a un cierto propietario, de los posibles peligros y ataques efectuados por agentes no autorizados, por ende, la seguridad informática se ocupa de proteger los recursos de un sistema informático tales como: información, servicios y arquitecturas.

Los algoritmos modernos se basan en su seguridad por medio de la utilización de llaves; donde el mensaje solo puede ser desencriptados si se tiene la llave que coincide para encriptar.

Existen dos clases de algoritmos de encriptación basados en llaves, algoritmos simétricos o de llave privada y algoritmos asimétricos o de llave pública. La diferencia es que los algoritmos simétricos utilizan la misma clave para encriptar y desencriptar y los algoritmos asimétricos utilizan llaves distintas para la encriptación y desencriptación. Generalmente, los algoritmos simétricos son más rápidos de ejecutar en las computadoras que los asimétricos.

1.1. Problema de investigación

1.1.1. Planteamiento del problema

Con el avance tecnológico en las telecomunicaciones y la transferencia de información para la comunicación digital, existe un problema en la seguridad para la comunicación de datos entre las redes existentes en el planeta las mismas que pueden ser interceptadas pudiendo ser alterada la información en cualquier instante, por ende, la seguridad de la transmisión de datos entre emisor y receptor se debe garantizar en todo momento.

Para garantizar la seguridad de la información se debe crear un mecanismo para transferir los datos de un sitio a otro por medio de la criptografía mediante sistemas informáticos se pueda proteger la información en cualquier instante dando seguridad a las comunicaciones.

Es necesario transmitir información utilizando técnicas que consigan datos ilegibles utilizando métodos de encriptación que pueden ser simétricos y asimétricos, considerándose la velocidad de procesamiento en el caso de transmisión de datos digitales y texto. (Michael Brady, s. f.).

Existen varias investigaciones realizadas acerca de creación de un algoritmo de encriptación aplicado a la comunicación humano–robot:

- La investigación “Desarrollo de robot móvil de exploración dirigido mediante transferencia de video”, el objetivo es dirigir a un robot con movimientos por intermedio de la transmisión de video, utilizando el algoritmo de encriptación AES. («HCA066SN.pdf», s. f.).
- La investigación “Control Remoto por Voz Robot Móvil Pioneer P3-DX”. Este proyecto consistió en un control remoto por voz para el robot móvil Pioneer P3-DX que posee el Departamento de Eléctrica y Electrónica de la ESPE. Para realizar el control de la plataforma robótica se han integrado dos entornos de programación distintos, MATLAB escrito en lenguaje propietario y ARIA desarrollada en lenguaje C++, se utiliza la técnica de encriptación AES. («Tesis_Completa - T-ESPE-047250.pdf», s. f.).
- La investigación "ZigBee profile specification for mobile robotics" presenta la especificación de un perfil de ZigBee para la robótica móvil, destacando los principales sensores y actuadores necesarios para el desarrollo de las actividades de dichos sistemas, y las relaciones con los grupos definidos en la biblioteca clúster ZigBee existente, la encriptación de información se la realiza con el algoritmo RSA. (Villegas, Madrigal, & Serna, 2012).
- La investigación “Diseño e Implementación de una plataforma hardware y software para el control de un robot móvil utilizando el Ipad”, se desarrolló en una plataforma hardware y software que permite el control y supervisión de un robot móvil a escala utilizando una Tablet iPad, este trabajo está enmarcado en una investigación macro para acercar a las personas de tercera edad con las tecnologías robóticas, gracias a la facilidad de manejo que presentan los dispositivos de computación móvil. El sistema cuenta con una plataforma de comunicación en base a un servidor LAMP, construido en una Raspberry pi que permite la transferencia de información entre el robot móvil el Ipad y un computador, para la encriptación se utiliza el algoritmo DSA. («digital_24689.pdf», s. f.).

La presente investigación se centrará en la selección de un algoritmo de encriptación que será aplicado a la comunicación humano-robot, con el objetivo de mejorar la seguridad de la información al momento de transmitirla.

1.2. Formulación del problema

¿Cómo mejorará la seguridad con el uso de un algoritmo criptográfico para el intercambio de información entre emisor y receptor aplicado humano-robot, para evitar posibles alteraciones y/o pérdidas de datos, que afectarían la ejecución de órdenes por parte del robot o la toma de decisiones por parte del humano?

1.3. Sistematización del problema

¿Qué algoritmos criptográficos son utilizados en la actualidad?
¿Qué ventajas y desventajas brindan los algoritmos criptográficos actuales?
¿Qué método se podría utilizar para prevenir ataques básicos en comunicaciones de información emisor y receptor?
¿Cuáles son las técnicas de comunicación entre humano-robot?
¿Cómo aplicar el algoritmo de encriptación en la técnica de comunicación humano–robot para la mejorar la seguridad?

1.4. Justificación

1.4.1. Justificación teórica

El presente trabajo de investigación plantea la integración de dos campos relacionados con la seguridad de la información:

1. La criptografía.
2. La comunicación.

La encriptación se da como una medida de seguridad y es utilizada para almacenar y transmitir información segura, que no pueda ser adquirida o interceptada por terceras personas. Existe un proceso inverso a la encriptación que es la desencriptación de la información que se transmitida entre emisor y receptor, obteniendo el estado original de los datos con métodos de encriptación que no pueden ser revertidos.

La encriptación de los datos en una comunicación para la transferencia de información entre humano-robot permiten que estén seguros hacia su destino a través de un medio, como son las redes informáticas y así proteger los datos entre ambas partes de la comunicación.

1.4.2. Justificación metodológica

La ventaja de la criptografía es la encriptación de la información, por tal motivo se seleccionará y se utilizará un algoritmo especializado, donde se compararán cada uno de los beneficios existentes de los métodos criptográficos y posteriormente establecer el algoritmo óptimo para la transmisión de datos entre redes informáticas teniendo así una mayor seguridad.

Una vez realizado el nuevo algoritmo de encriptación éste protegerá la información a ser transmitida por algún medio de comunicación como las redes informáticas, el cuál será utilizado en la comunicación humano-robot, garantizando así la seguridad de la información entre emisor y receptor. Cuando la transferencia de datos sea interceptada por terceros, ésta información no podrá ser interpretada ni modificada, garantizando así la encriptación efectiva de los datos.

1.4.3. Justificación práctica

Para realizar los experimentos de ésta investigación se utilizarán tres escenarios de prueba, como se detallan a continuación:

1. En el primer experimento de prueba y sin ninguna encriptación de datos, consistirá en alterar la transmisión de información, verificando así que el receptor no obtiene la misma información que fue enviada por el emisor
2. Por medio del segundo escenario como experimento de prueba se utilizará un algoritmo criptográfico, el mismo que será agregado a la transmisión de datos mejorando la seguridad.

1.5. Objetivos

1.5.1. Objetivo general

- Implementar un algoritmo de encriptación aplicado a la comunicación humano–robot para garantizar la seguridad de información transmitida.

1.5.2. Objetivos específicos

- Comparar los algoritmos criptográficos existentes en la actualidad y seleccionar el óptimo para la encriptación de datos.
- Asegurar la integridad de la información, órdenes de acciones, que el dispositivo recibe y envía.
- Implementar la encriptación de datos mediante el algoritmo seleccionado e incorporarlo en la comunicación humano-robot.
- Comprobar el nivel de seguridad en los escenarios de prueba implementados con el nuevo algoritmo en la incorporación en la comunicación humano-robot.

1.6. Hipótesis

La utilización de un algoritmo de encriptación aplicado a la comunicación humano-robot permitirá mejorar la integridad de los datos que controlan tanto los movimientos mecánicos como la información que proporciona el robot.

CAPÍTULO II

2. MARCO REFERENCIAL

2.1. Comunicación e información

La comunicación es percepción, crea expectativas y plantea exigencias, mientras que la información aumenta el conocimiento, comunica novedades es decir la información mejora la comunicación. Los mecanismos de transferencia de la comunicación están dados por: participantes, prototipo, canales de transmisión segura o insegura, interconexión entre canales de comunicación, personas y grupos, redes de transmisión empleadas para la comunicación.

La información tiene como función principal transmitir datos necesarios para tomar decisiones en una comunicación y la comunicación tiene como objetivo interactuar con individuos, grupos mediante talleres, servicios.

2.2. Redes de transmisión de datos

Las redes informáticas permiten compartir cualquier tipo de información y transferir datos entre computadoras con gran difusión geográfica, sumamente rápido y en grandes volúmenes. (Eugenio Duarte, 2013).

2.3. Seguridad de la información

La transferencia de información por intermedio de las redes informáticas permite el intercambio de datos importantes para los usuarios. Los datos son enviados muchas veces sin protección alguna, donde la información es de suma importancia y de gran trascendencia, esta información debe ser protegida de alguna manera por medio de una codificación conocida solo por los interesados entre un emisor y receptor (encriptación). (Marlon Jiménez Bazán, s. f.).

El interés actual de los usuarios con respecto a la seguridad de las transmisiones de datos y el cifrado de la información siempre ha habido la necesidad de ocultar información, mucho antes de que existieran los primeros equipos informáticos y calculadoras. Actualmente el internet ha evolucionado hasta convertirse en una herramienta esencial de la comunicación a nivel global,

sin embargo, esta comunicación implica un número creciente de problemas estratégicos relacionados con las actividades de las empresas en la web por la seguridad de los datos. La seguridad de ésta información al ser enviada por cualquier medio de comunicación debe garantizarse dando lugar a la criptografía. (Arturo Ribagorda Garnacho, s. f.).

Para un nivel de seguridad alto en la información a ser transmitida se puede utilizar cualquier algoritmo de encriptación teniendo un conocimiento previo del algoritmo tanto en la parte transmisora como receptora, sin este conocimiento previo no tendría sentido el proceso, ya que una pequeña desincronización entre las partes puede producir la pérdida total o parcial de la información considerando la aparente no relación entre los datos originales y encriptados, que es característica de un buen sistema de seguridad. (Marlon Jiménez Bazán, s. f.).

2.4. Estudio de la criptología

2.4.1. Definiciones generales

Criptología: Se ocupa del estudio de los algoritmos, protocolos y sistemas que se utilizan para proteger la información y dotar de seguridad a las comunicaciones y a las entidades que se comunican, es decir estudia la criptografía y criptoanálisis.

Criptografía: Codifica un mensaje haciéndolo ininteligible o cifrado y recuperando el mensaje original a partir de esa versión ininteligible a esto se le conoce como descifrado.

Algoritmo criptográfico: Es un método matemático utilizado para cifrar y descifrar un mensaje. Un mensaje antes de ser cifrado se denomina texto en claro y una vez cifrado se denomina texto cifrado.

Sistema criptográfico: Es utilizado para cifrar y descifrar información compuesto por un conjunto de algoritmos criptográficos, claves y, posiblemente, varios textos en claro con sus correspondientes versiones en texto cifrado.

Algoritmos de resumen de mensajes: Es la variación del mensaje de un tamaño variable a texto cifrado de un tamaño fijo sin utilizar claves.

Algoritmos de clave pública o asimétrica: Usa una clave privada para cifrar el mensaje y una clave pública para descifrar.

Algoritmos de clave secreta o simétrica: Transforma un mensaje en texto cifrado con el mismo tamaño del original manteniendo la una única clave para cifrar y descifrar, como es el algoritmo criptográfico SHA-256.

Criptoanálisis: Conjunto de instrucciones, procesos y métodos empleados para romper un algoritmo criptográfico, descifrar un texto cifrado o descubrir las claves empleadas para generarlo.

2.4.2. *Criptografía*

Para transmitir información secreta desde cualquier medio de comunicación es haciendo uso de la criptografía. El término criptografía procede del griego kryptos, que significa oculto, y graphein, que representa escribir; el significado de esta etimología sería "escritura oculta". La técnica de la criptografía es proteger archivos, documentos y datos cuya función es utilizar codificación secreta en archivos, documentos y datos personales que transitan a través de las redes locales o en internet. (Ribagorda Arturo Garnacho, s. f.).

Con la criptografía se puede diseñar, implementar, implantar, y hacer uso de varios sistemas criptográficos para transmitir información de forma segura. La criptografía está basada en operaciones matemáticas como es el caso de un texto que consiste en transformar las letras del texto en una serie de números (en forma de bits ya que los equipos informáticos usan el sistema binario) y luego realizar cálculos con estos números para: transformar en incomprensibles el mensaje cifrado llamado **texto cifrado y** con el mensaje inicial llamado **texto simple** y el receptor pueda descifrarlos.

La codificación de un mensaje para que sea secreto se llama cifrado y el proceso inverso consiste en recuperar el mensaje original se le llama descifrado.

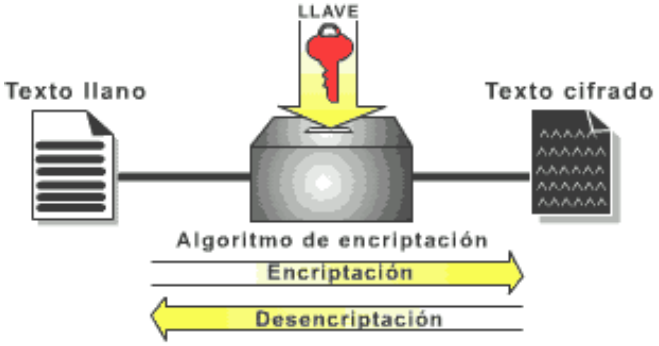


Figura 1-2: Transmisión de un mensaje
Fuente: (Gutiérrez Pedro, 2013)

La eficiencia de un sistema criptográfico está basada en la seguridad que aporta dicho sistema, midiendo la seguridad en función de la forma incondicional o si cumple ciertas condiciones. El cifrado normalmente se realiza mediante una clave de cifrado y el descifrado requiere una clave de descifrado. Las claves generalmente se dividen en dos tipos:

- Las claves simétricas: son las claves que se usan tanto para el cifrado como para el descifrado. En este caso se habla de cifrado simétrico o cifrado con clave secreta. (Laboratorio de Redes y Seguridad, s. f.).

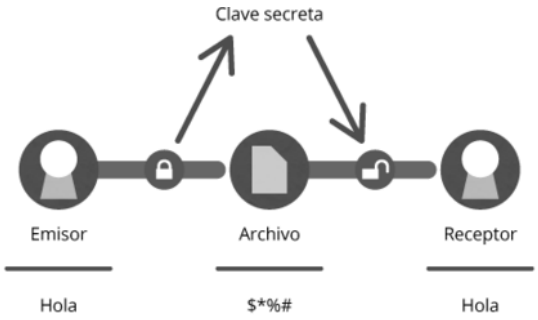


Figura 2-2: Clave simétrica
Fuente: (Gutiérrez Pedro, 2013)

- Las claves asimétricas: son las claves que se usan en el caso del cifrado asimétrico llamado cifrado con clave pública. En este caso, se usa una clave para el cifrado y otra para el descifrado. (Laboratorio de Redes y Seguridad, s. f.).

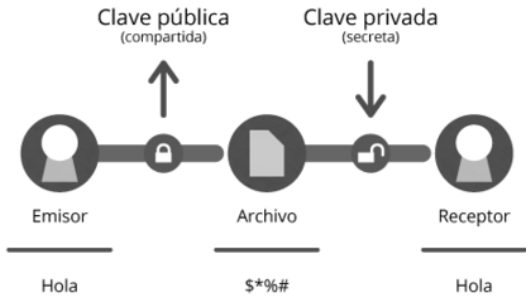


Figura 3-2: Clave asimétrica
Fuente: (Gutiérrez Pedro, 2013)

La criptografía proporciona algunas propiedades, como se describe a continuación:

Confidencialidad: Garantiza el acceso a la información por medio del personal autorizado.
Integridad: Garantiza la corrección y completitud de la información.
Autenticación: Proporciona mecanismos que permiten verificar la identidad del comunicante.
No repudio: Protege en caso de que alguna de las entidades implicadas en la comunicación pueda negar haber participado en toda o parte de la comunicación.

2.4.2.1. Ventajas y desventajas de algoritmos criptográficos

Tabla 1-2: Ventajas y desventajas de algoritmos criptográficos.

CLAVES	VENTAJAS	DESVENTAJAS	ALGORITMOS
SIMÉTRICA	Sistema eficiente en grupos muy reducidos, ya que solo es necesaria una única clave. No es necesario de disponer de una tercera parte confiable Infraestructura sencilla. Confidencialidad, Integridad.	Es necesario compartir la clave entre emisor y receptor por medios inseguros. Si se compromete la clave se compromete toda la comunicación. No permite autenticar al emisor ya que una misma clave la utilizan dos personas. Se necesita un elevado número de claves n*(n-1)/2, siendo n el número de personas implicadas en una comunicación cifrada.	DES con tamaño de clave de 56 bits. Triple-DES con tamaño de clave de 128 bits a 256 bits. Blowfish con tamaño de clave de 128 bits a 256 bits. AES con tamaño de clave de 128, 192 o 256 bits
ASIMÉTRICA	Número de claves reducido, ya que cada individuo necesita únicamente un par de claves. Computacionalmente es complicado encontrar la clave privada a partir de la pública. No es necesario transmitir la clave privada a partir de la pública entre emisor y receptor. Permite autenticar a quien utilice la clave pública. Confidencialidad, Integridad, Autenticación de origen, No repudio.	Alto costo computacional en el proceso de generación de claves. La necesidad de un tercero (autoridad de certificación) en el proceso. Necesidad de una gran infraestructura independientemente del número de individuos. Se precisa mayor tiempo de proceso y claves más grandes.	RSA con tamaño de clave igual a 1024 bits. DSA con tamaño de clave de 512 bits a 1024 bits. ElGamal con tamaño de clave comprendida entre 1024 bits y los 2048 bits

Fuente: («Guimi», 2012).
Realizado por: Sánchez Wilian

2.4.2.2. Técnicas criptográficas

Acuerdo de claves: Permite a dos partes que no tienen un conocimiento previo el uno del otro, acordar una clave secreta y establecer una comunicación cifrada usando un canal inseguro. (Guimi, 2009).

Autenticación de las partes: Permiten verificar que los extremos de una comunicación son quienes dicen ser. (Guimi, 2009).

Firma electrónica: Permiten firmar un fichero, un mensaje o un resumen mediante su cifrado con una clave, de forma que se garantice la procedencia mediante un correcto descifrado. (Guimi, 2009).

2.5. Encriptación

El encriptado transforma la información en datos con signos ilegibles para los usuarios que no posean la clave secreta para desencriptarlos. El propósito de la encriptación es asegurar la privacidad de la información e inclusive a aquellos usuarios con acceso a la información encriptada. (Paul Fahn, s. f.).

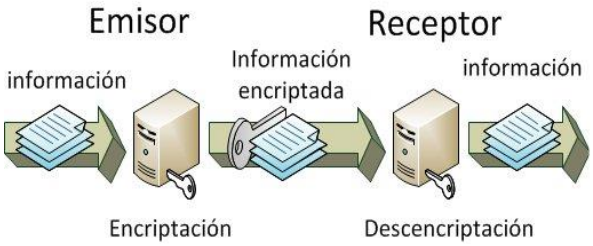


Figura 4-2: Sistema estándar de encriptación
Fuente: (Jean Triquet, 2016).

2.6. Algoritmos criptográficos

La función principal de un algoritmo criptográfico es codificar los datos logrando así alcanzar autenticación, integridad y confidencialidad.

2.6.1. Procesamiento de algoritmos criptográficos

Son usados con funciones matemáticas en los procesos de encriptación y desencriptación; como resultado al utilizar las funciones matemáticas se obtiene datos encriptados y para el procesamiento inverso de ese cálculo se obtiene la desencriptación. Si la llave o los datos son modificados el algoritmo produce un resultado diferente. (Laboratorio de Redes y Seguridad, s. f.).

Un algoritmo criptográfico debe estar diseñado para hacer difícil la desencriptación los datos sin utilizar la llave. Los algoritmos criptográficos se pueden clasificar en tres grupos:

1. Algoritmo Criptografía simétrica o de clave secreta.
2. Algoritmo Criptografía asimétrica o de clave pública: Función Hash o de resumen.

2.6.1.1. *Algoritmo simétrico*

Los algoritmos simétricos o claves secretas encriptan y desencriptan con la misma llave. Las principales ventajas de los algoritmos simétricos son su seguridad y su velocidad. Son aproximadamente 1.000 veces más rápidos que los asimétricos. (Saravia, s. f.-a).

Tabla 2-2: Algoritmos criptográficos simétricos

ALGORITMO	DESCRIPCIÓN	LONGITUD CLAVE	TAMAÑO BLOQUE
DES	La cadena de 56 bits es muy corta, por ende puede ser comprometida en menos de 1 día.	52 bits	No
TDES	También conocido como 3DES o TripleDES. En realidad, se dobla la longitud efectiva, siendo de 112 bits.	192 bits (eficaces 112 bits).	No
RC2	Es seguro contra ataques de fuerza bruta eligiendo el tamaño de clave apropiadamente.	Variable entre 64 y 128 bits.	No
ICE	Es seguro contra el criptoanálisis diferencial y lineal.	Cualquier múltiplo de 64 bits.	No
IDEA	Se usa para cifrar como para descifrar y es considerado uno de los más seguros.	128 bits	64 bits
GOST	Es utiliza como función hash.	256 bits	64 bits
AES	Está basado en sustituciones, permutaciones y transformaciones lineales.	128, 192 y 256 bits.	128 bits
RINDJAEL	Se basa en una red de sustitución-permutación.	Variable múltiplo de 4 bytes: 128, 192 y 256 bits.	Variable múltiplo de 4 bytes: 128, 192 y 256 bits.
SERPENT	Las operaciones se realizasen en paralelo, usando 32 desplazamientos de 1 bit.	Variable, 128, 192 y 256 bits.	128 bits

Fuente: (Mariiss, 2012).
Realizado por: Sánchez Wilian

2.6.1.2. *Algoritmo asimétrico*

También conocidos como clave pública, utilizan claves diferentes es decir una clave encripta y otra clave desencripta por ende cada clave es diferente de la otra. Este tipo de algoritmos poseen longitud mayor en sus claves comparado con el algoritmo simétrico. (Villegas et al., 2012).

Tabla 3-2: Algoritmos criptográficos asimétricos

ALGORITMO	DESCRIPCIÓN	LONGITUD CLAVE	TAMAÑO BLOQUE
ElGamal	Se utiliza para generar firmas digitales como para cifrar/descifrar, cuyo funcionamiento se basa en realizar cálculos sobre "logaritmos discretos"	Sin límite. El límite lo marca el protocolo usado	No
RSA	Trabaja con dos claves diferentes: clave "pública", y "clave privada" donde las dos claves son complementarias entre sí.	Longitud de clave: Variable: 128, 256, 1024, 2048 y 4096 bits.	No

DSA	Es utilizado para las firmas digitales y está en constante desarrollo por tener el apoyo gubernamental.	Entre 1024 y 3072 bits.	No

Fuente: (Mariiss, 2012).
Realizado por: Sánchez Wilian

2.6.1.3. Algoritmo HASH

La confidencialidad es la función de lograr que nadie sepa el contenido de la información enviada a través de una canal de comunicación considerado inseguro entre un emisor y receptor. Con la propiedad de integridad, lo que se busca es asegurar que cualquier mensaje enviado no haya sido alterado mientras este estuvo en tránsito. (Villegas et al., 2012).

El algoritmo de encriptación HASH es usado en múltiples aplicaciones, como los arrays asociativos, criptografía, procesamiento de datos y firmas digitales, entre otros. Mientras más seguro es este tipo de cifrado más lento es su procesamiento y su uso.

El método HASH permite resumir identificando de manera íntegra la información existente en un mensaje, texto, archivo, logrando evitar que la información pueda ser modifica. (D. Bauer, s. f.).

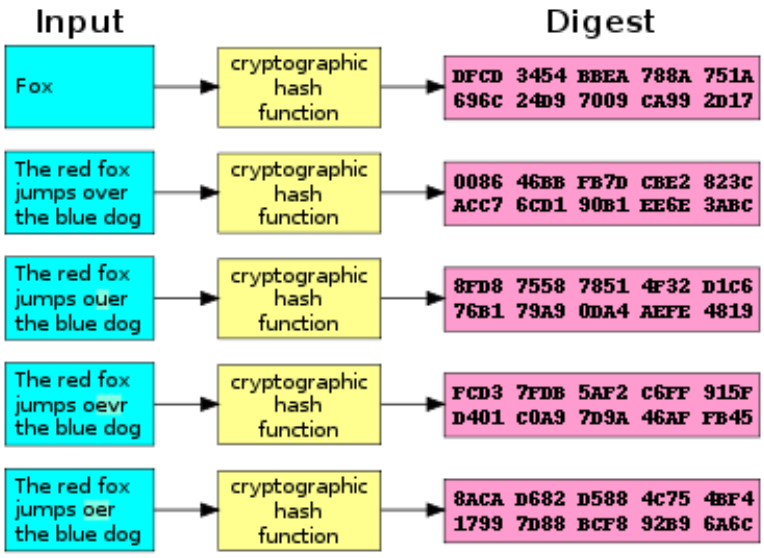


Figura 5-2: Algoritmo criptográfico hash
Fuente: (Gutiérrez, 2013).

Características principales HASH

- Funciona en una solo dirección es decir es unidireccional.
- Si cambian cualquier carácter en la transmisión varía el resumen final.
- Este tipo de método es de tamaño estándar.

Tabla 4-2: Algoritmos criptográficos HASH

ALGORITMO	DESCRIPCIÓN	LONGITUD CLAVE	TAMAÑO BLOQUE
MD5	Comprueba que algún archivo no haya sido modificado, es inseguro.	128 bits	No
SHA-1	Es un algoritmo más robusto y seguro.	160 bits	160 bits
SHA-2	Este algoritmo es más seguro que SHA-1.	512 bits	320 bits

Fuente: (Mariiss, 2012).
Realizado por: Sánchez Wilian

Tabla 5-2: Comparación entre criptografía simétrica y asimétrica

ATRIBUTO	CLAVE SIMÉTRICA	CLAVE ASIMÉTRICA
Años de utilizar	Miles de años	Menor a los 50 años
Objetivo principal	Cifrar grandes volúmenes de datos	Intercambio de claves, firma digital
Estándares actuales	Triple DES , DES, AES	Diffie-Herllman, DSA, SHA, RSA
Velocidad de encriptación	Velocidad rápida	Velocidad lenta
Uso de laves	Comparte entre usuarios	Clave privada y pública
Intercambio de uso de claves	El intercambio por un canal inseguro es muy difícil	Clave pública es compartida por cualquier canal y la clave privada no se comparte.
Tamaño de claves	Tamaño 56 bit es vulnerable y tamaño a 256 es seguro.	Tamaños de 1024-2048 bits.
Seguridad	Confidencialidad Integridad Autenticación	Confidencialidad Integridad Autenticación, No repudio

Fuente: (Mariiss, «Criptografía simétrica y asimétrica», 2014)
Realizado por: Sánchez Wilian

2.7. Análisis de selección del algoritmo criptográfico

Para el análisis de selección del algoritmo criptográfico son el AES y las funciones resumen SHA-1 y SHA-256. Estos algoritmos son implementados como coprocesadores del procesador MicroBlaze utilizando interfaces FSL para el intercambio de datos entre ellos. Estos coprocesadores son integrados dentro de la biblioteca OpenSSL considerando la naturaleza multitarea del sistema operativo Linux, por lo que se selecciona un mecanismo de sincronización para controlar el acceso a estos dispositivos. (Cabrera Aldaya, Sarmiento, & José, 2013a). El proyecto OpenSSL es un esfuerzo de colaboración para desarrollar un conjunto de herramientas robusto, de grado comercial, completo y de código abierto que implementa los protocolos Secure Sockets Layer (SSL v2 / v3) y Transport Layer Security (TLS v1) Biblioteca de criptografía de propósito general de fuerza. (Cabrera Aldaya, Sarmiento, & José, 2013a)

El protocolo TLS (TransportLayer Security) permite establecer comunicaciones seguras entre dos nodos a través de redes IP. El mismo se divide en dos etapas fundamentales: inicio de sesión e intercambio de datos. El propósito del inicio de sesión es lograr que los nodos que desean establecer una comunicación segura acuerden las claves secretas necesarias para enviar y recibir datos durante la segunda etapa del protocolo. Para llevar a cabo esta tarea se utilizan algoritmos criptográficos de intercambio de claves y de firma digital, ambos basados en criptografía asimétrica (clave de cifrado difiere de la clave de descifrado). La segunda etapa utiliza las claves generadas durante el inicio de sesión para proteger los datos intercambiados entre los dos nodos. Durante esta etapa se utilizan algoritmos de cifrado simétrico y de chequeo de integridad. (Cabrera Aldaya, Sarmiento, & José, 2013a).

Tabla 6-2: Consumo de tiempo de distintos algoritmos

RSA-4096	ECDSA-571	AES-256	SHA-512
20 s (firma)	8,3 s (verificación)	6 ms (1500 bytes)	18 ms (1500 bytes)

Fuente: (Cabrera Aldaya, Sarmiento, & José, 2013b).
Realizado por: Sánchez Wilian

El algoritmo AES-256 y las funciones resumen (HASH) SHA-1 y SHA-256 fueron las seleccionadas durante esta investigación por ser algoritmos seguros y de tamaño de claves de 256 bits para su encriptación. A continuación, se presentan algunas comparaciones de la velocidad de procesamiento de los algoritmos implementados en software en la propia biblioteca OpenSSL, así como utilizando los coprocesadores diseñados durante esta investigación. (Cabrera Aldaya et al., 2013a).

En la TABLA 7-2, se muestran las velocidades de ejecución de las descripciones software (SW) y hardware (HW) de los algoritmos AES-256 y HMAC utilizando las funciones SHA-1 y SHA-256. Como se puede apreciar en esta TABLA 2.7, la aceleración obtenida con el coprocesador hardware del algoritmo AES-256-CBC varía entre 4,7 y 25,8 veces, siendo mayor a medida que se incrementa el tamaño de bloque. (Cabrera Aldaya et al., 2013a).

Tabla 7-2: Velocidades obtenidas al integrar los algoritmos

IMPLEMENTACIÓN	VELOCIDAD EN KBYTES/S PARA DIFERENTES TAMAÑOS DE BLOQUES (EN BYTES)		
	16	**256**	**8192**
AES-256-CBC-SW	249	266	277
AES-256-CBC-HW	1187	4836	7169
HMAC-SHA1-SW	87	791	1804
HMAC-SHA1-HW	113	1890	16871
HMAC-SHA256-SW	32	195	310
HMAC-SHA256-HW	132	1365	17440

Fuente: (Cabrera Aldaya et al., 2013b).
Realizado por: Sánchez Wilian

La Tabla 7-2, muestra la velocidad de ejecución de las implementaciones hardware de los algoritmos utilizando o no la función flock, permitiendo apreciar el impacto (en la velocidad) de la utilización de esta función para sincronizar el acceso a los coprocesadores por parte de diferentes aplicaciones. (Cabrera Aldaya et al., 2013a).

Tabla 8-2: Sincronización del acceso a coprocesadores

IMPLEMENTACIÓN	VELOCIDAD EN KBYTES/S PARA DIFERENTES TAMAÑOS DE BLOQUES (EN BYTES)		
	16	**256**	**8192**
AES-256-CBC	1374	5658	7017
AES-256-CBC-BLOCK	1187	4836	7169
HMAC-SHA1-	156	2364	19501
HMAC-SHA1-BLOCK	113	1890	16871
HMAC-SHA256	164	2557	21041
HMAC-SHA256-BLOCK	132	1365	17440

Fuente: (Cabrera Aldaya et al., 2013b).
Realizado por: Sánchez Wilian

Por otra parte, la herramienta OpenVPN permite implementar redes privadas virtuales (VPN, Virtual Private Network) a nivel de usuario utilizando los protocolos SSL/TLS. Esta herramienta utiliza la biblioteca OpenSSL para la implementación de estos protocolos y de los algoritmos criptográficos involucrados.

En la Tabla 9-2 se muestran las velocidades alcanzadas con diferentes configuraciones de la herramienta OpenVPN. La primera línea muestra la velocidad alcanzada sin la utilización de criptografía para proteger el túnel de comunicaciones entre los dos extremos de la VPN, mientras que en las restantes se puede apreciar la velocidad utilizando la implementación software de los algoritmos y los coprocesadores diseñados. (Cabrera Aldaya et al., 2013a).

Tabla 9-2: Impacto coprocesadores en la aceleración de redes VPN

CONFIGURACIÓN DE OPENVPN	VELOCIDAD ALCANZADA (MBPS)
Sin criptografía	6,72
SW (AES-SHA1)	2,35
HW (AES-SHA1)	3,51
SW (AES-SHA256)	1,93
HW (AES-SHA256)	3,49

Fuente: (Cabrera Aldaya et al., 2013b).
Realizado por: Sánchez Wilian

La velocidad (Tabla 2.9) se multiplica 1,8 veces cuando se utilizan los algoritmos AES y SHA-256 con respecto a la implementación software, además la velocidad alcanzada utilizando la función SHA-1 y SHA-256 es prácticamente la misma, por lo que es posible aumentar la seguridad del sistema sin comprometer el rendimiento del sistema si se utilizan los coprocesadores criptográficos.

Los algoritmos de encriptación AES y SHA, alcanzan velocidades similares para la seguridad de la información mediante la encriptación, por tal motivo se puede utilizar el algoritmo criptográfico AES o SHA para el respectivo trabajo de investigación, optando por el algoritmo de encriptación SHA, donde el algoritmo de encriptación no comparte la clave privada.

2.8. Ataques a la información de datos

En la actualidad saber acerca de las diferentes etapas que conforman un ataque informático brinda la ventaja de aprender a pensar como los atacantes. Desde la perspectiva del profesional de seguridad, se debe aprovechar esas habilidades para comprender y analizar la forma en que los atacantes llevan a cabo un ataque, como se muestra en la figura N° 2.6. y se describe a continuación:

Figura 6-2: Fases o etapas de un ataque informático
Fuente: (Diliana Fernández, 2014).

FASE 1:

Reconnaissance (Reconocimiento): El atacante realiza tácticas para obtener toda la información necesaria. (Dilianafernandez, 2014).

FASE 2:

Scanning (Exploración): Utiliza toda la información que fue obtenida en el proceso anterior de reconocimiento para identificar vulnerabilidades específicas. (Dilianafernandez, 2014).

FASE 3:

Gaining Access (Obtener acceso): El atacante realiza el acceso al sistema informático donde utiliza las vulnerabilidades que encontró en la Fase de Exploración, por ende, el acceso puede ocurrir localmente, offline, o sobre el Internet. (Dilianafernandez, 2014).

FASE 4:

Maintaining Access (Mantener el acceso): El atacante utiliza recursos propios y los recursos del sistema informático accedido, pudiendo así realizar nuevos ataques por medio del sistema informático atacado. Una vez que el atacante mantiene el acceso éste puede alterar todo el funcionamiento de aplicaciones de software. (Dilianafernandez, 2014).

FASE 5:

Covering Tracks (Borrar huellas): El atacante intenta desaparecer la evidencia de algún posible rastreo de las actividades ilegales realizadas para seguir teniendo el acceso parcial o total al sistema informático comprometido. (Dilianafernandez, 2014).

2.8.1. Ataques destinados a páginas y portales web

Los principales tipos de ataques realizadas a las páginas y portales web se detallan a continuación:

XSS - cross site scripting: La función de este tipo de ataque es inserta código o script en el sitio web de la víctima para obtener hurto de sesiones o datos vulnerables. (Urrego Jonatan, 2013).

Fuerza bruta: Realizan procesos automatizados mediante pruebas de usuario y contraseña al azar. (Urrego Jonatan, 2013).

Inyección de código: El objetivo principal de este tipo de ataques es realizar la inyecta de código fuente en SQL, HTML en el sitio web que está siendo atacado. (Urrego Jonatan, 2013).

DoS- Denegación del servicio: Aprovecha los errores en la programación realizada de un sitio web, siendo atacado a los recursos del servidor como es el procesador, memoria. (Urrego Jonatan, 2013).

Fuga de información: Se basa donde el administrador del sitio deja abierto el registro de errores del sistema. (Urrego Jonatan, 2013).

2.8.2. Ataques destinados a personas y usuarios de internet

Cualquier usuario conectado a Internet siempre estará expuesto a riesgos de seguridad de su información parcial o total.

Pesca de datos – Phishing: Está basado en suplantar o realizar copias exactas de la página web, donde el usuario ingresará datos personales tales como claves, números de tarjeta, etc. (Urrego Jonatan, 2013).

SCAM: La función de este tipo de ataque es en realizar mensajes falsos de regalos dinero, recompensas, herencias, premios de lotería. (Urrego Jonatan, 2013).

Ingeniería social: Es utilizado mediante la suplantación de personas y entidades logrando obtener datos personales, por intermedio de llamadas telefónicas, mensajes de texto o funcionarios falsos. (Urrego Jonatan, 2013).

Spoofing: Consiste en suplantar la identidad del computador. (Urrego Jonatan, 2013).

Troyano: Se instala programas espías dentro del computador afectado, logrando así tener el manejo remoto, cambio de archivos, robo de información, captura de datos personales. (Urrego Jonatan, 2013).

2.8.3. Hacking ético

Es la forma donde una persona usa sus altos niveles de conocimientos de informática y seguridad para realizar pruebas en redes y encontrar vulnerabilidades, luego reportarlas para que se tomen medidas de seguridad y así no se pueda hacer hurto de la información.

2.8.3.1. *Herramientas hacking ético*

Nmap: Controla de forma remota la seguridad de un sistema informático por medio de puertos abiertos, servicios de red, registro del host, actividad de red informática y configuración del host. (Sistema operativo, firewall, etc). (Velasco, 2015).

Metasploit: Utiliza pequeñas software llamados exploits diseñadas para aprovechar vulnerabilidades en otras aplicaciones o sistemas operativos. (Velasco, 2015).

Angry ip scanner: Es también conocida como IPScan la misma que es utilizada para realizar escaneos de red. (Velasco, 2015).

Thc Hydra: Prueba la seguridad de las contraseñas en las páginas web. (Velasco, 2015).

Cain and abel, el todo en uno de las contraseñas: Realiza ataques a los datos de la red examinando las debilidades en las contraseñas de los usuarios por medio de técnicas de explotación, fuerza bruta, diccionario y criptoanálisis. (Velasco, 2015).

John the ripper: Realiza ataques de diccionario, fuerza bruta para poder obtener contraseñas. (Velasco, 2015).

Burp suite: Es aplicado específicamente para aplicaciones web mediante el "Burp Suite Spider", donde se enumeran los parámetros vulnerables de una red. (Velasco, 2015).

Ettercap: Se utiliza esta herramienta en diferentes ataques de la red como ARP poisoning para identificar sistemas dentro de una red. (Velasco, 2015).

Nessus remote security scanner: Por medio de esta herramienta se analiza y se comprueba vulnerabilidades en sistemas remotos. (Velasco, 2015).

Wapiti: Sirve para el escaneo total de un sistema o red informática detectando varias vulnerabilidades existentes también realiza un análisis de seguridad de aplicaciones web. (Velasco, 2015).

2.8.4. Ataques contra los sistemas criptográficos

Atacar a los sistemas criptográficos se puede descubrir las claves utilizadas para cifrar determinados mensajes o documentos almacenados en un sistema, o bien obtener determinada información sobre el algoritmo criptográfico utilizado. (Urrego Jonatan, 2013). Podemos distinguir varios tipos de ataques contra los sistemas criptográficos: (Emanuel García, 2012).

- Ataque a partir del cifrado.
- Ataque a partir del texto en claro.
- Ataque a partir del texto en claro elegido.
- Ataque a partir de la clave.

- Suplantación de identidad.
- Ataque mediante intromisión.
- Ataques adaptativos basados en texto claro conocido.
- Búsqueda exhaustiva.

2.9. Comunicaciones inalámbricas

La comunicación inalámbrica o sin cables es aquella en la que extremos de la comunicación (emisor/receptor) no se encuentran unidos por un medio de propagación físico, sino que se utiliza la modulación de ondas electromagnéticas a través del espacio. En este sentido, los dispositivos físicos sólo están presentes en los emisores y receptores de la señal. (Saravia, s. f.-b).

WI-FI: Es un estándar para la conexión de dispositivos electrónicos mediante tecnología inalámbrica. En la actualidad la mayor parte de los dispositivos que utilizamos incorporan esta tecnología (teléfonos, consolas, portátiles, etc.) con la que podemos conectarnos a una red (o Internet) sin la necesidad de utilizar cables. (Saravia, s. f.-b).

BLUETOOTH: Bluetooth es una especificación industrial para Redes Inalámbricas de Área Personal (WPANs) que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2,4 GHz. (Saravia, s. f.-b).

2.10. Robots

Actualmente, el concepto de robótica ha evolucionado hacia los sistemas móviles autónomos, que son aquellos que son capaces de desenvolverse por sí mismos en entornos desconocidos y parcialmente cambiantes sin necesidad de supervisión. Uno de los grandes retos de la robótica se basa en la colaboración y autodeterminación de sus movimientos, buscando que los robots puedan trabajar de una forma lo más autónoma e independiente del ser humano posible. Para ello, deberán hacer uso de todas sus posibilidades, siendo las más importantes la sensorización, el control y la comunicación (Adrián Cervera, 2011).

2.10.1. Tipos de robots

Los diferentes tipos de robots de se pueden realizar en distintas clasificaciones (por grado de autonomía, por tipo de propósito, por función, por medio, por tamaño y peso, anatomía e

inteligencia), a continuación, se describe los principales tipos de robots que se utilizan en la actualidad (Adrián Cervera, 2011).

2.10.1.1. Robots industriales

Se entiende por Robot Industrial a un dispositivo de maniobra destinado a ser utilizado en la industria y dotado de uno o varios brazos, fácilmente programable para cumplir operaciones diversas con varios grados de libertad y destinado a sustituir la actividad física del hombre en las tareas repetitivas, monótonas, desagradables o peligrosas (Adrián Cervera, 2011). Los robots industriales son: Robots industriales de primera, segunda, tercera, cuarta y quinta generación.

2.10.1.2. Robots móviles

Este tipo de robots además de las características de los robots industriales poseen nuevas características la cual es el movimiento en el espacio físico, es decir, la posibilidad de desplazarse por el entorno para observarlo e interactuar con él, y de esta forma emular con mayor fidelidad las funciones y capacidades de los seres vivos (Adrián Cervera, 2011). Los robots móviles se clasifican en: robots andantes, reptadores, nadadores, voladores y Rodantes.

Los robots rodantes son aquellos que desplazan haciendo uso de ruedas. Podemos encontrar varias configuraciones para la posición y el número de ruedas. Es la configuración que llevan los coches: dos ruedas con tracción traseras, y dos ruedas de dirección delanteras. (Adrián Cervera, 2011)**.**

2.10.2. Sistemas de control de robots

Un sistema de control es la combinación de componentes que actúan juntos para realizar el control de un proceso. Este control se puede hacer de forma continua, es decir en todo momento o de forma discreta, es decir cada cierto tiempo (Adrián Cervera, 2011).

- Si el sistema es continuo, el control se realiza con elementos continuos.
- Cuando el sistema es discreto el control se realiza con elementos digitales, como el ordenador, por lo que hay que digitalizar los valores antes de su procesamiento y volver a convertirlos tras el procesamiento.
- Existen dos tipos de sistemas, sistemas en lazo abierto y sistemas en lazo cerrado.

2.10.2.1. Sistemas de control en bucle abierto

Estos toman la información acerca de las condiciones de operación que reciben de varios sensores y entonces usan esa información para determinar (sea por medios mecánicos o usando medios electrónicos programados) exactamente qué acción debe aplicarse para alcanzar la situación deseada (Adrián Cervera, 2011).

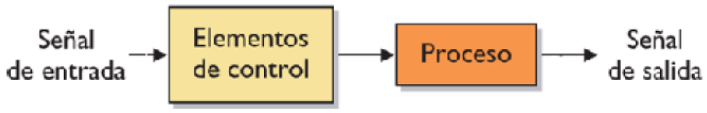


Figura 7-2: Esquema bucle abierto
Fuente: (Adrián Cervera, 2011).

2.10.2.2. Sistemas de control en bucle cerrado

En un sistema de lazo cerrado o de retroalimentación, la información acerca de cualquier cosa que esté siendo controlada es continuamente retro-alimentada al sistema como un dato de entrada (Adrián Cervera Andés, 2011).

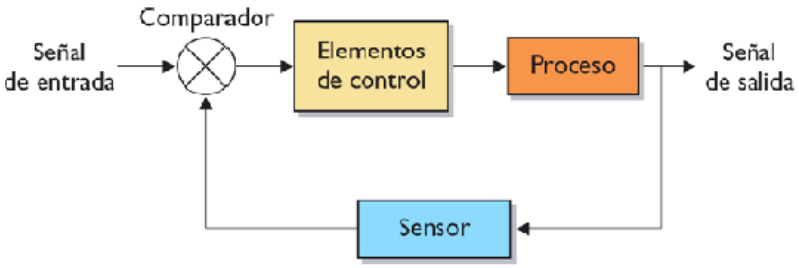


Figura 8-2: Esquema en bucle cerrado
Fuente: (Adrián Cervera, 2011).

CAPÍTULO III

3. DISEÑO DE INVESTIGACIÓN

3.1. Diseño de estudio

El diseño de ésta investigación es de tipo cuasiexpiremental, porque se selecciona un algoritmo criptográfico de entre varios algoritmos existentes en la actualidad, el mismo que será utilizado como base para su implementación en la comunicación entre humano-robot mejorando la seguridad de la información, generando datos de prueba por el autor de este trabajo de investigación.

3.2. Tipo de estudio

Para la realización de esta investigación se realizará por medio de dos tipos: tipo de estudio aplicativo y tipo de estudio experimental.

3.2.1. Tipo de estudio aplicativo

El trabajo de investigación está apoyado en investigaciones similares respecto a la utilización de un algoritmo criptográfico para establecer nuevos procesos que permitan mejorar la seguridad de transmisión de la información existente.

3.2.2. Tipo de estudio experimental

Se basa en pruebas experimentales como son escenarios de laboratorio, donde se podrá observar los elementos más significativos del ente de estudio que se investiga logrando obtener una captación de los fenómenos a primera vista, mejorando el nivel de seguridad de transmisión de información.

3.3. Población de estudio

La población de estudio para validar la muestra está constituida por pruebas de laboratorio (algoritmo criptográfico) mediante ataques inducidos a la seguridad de datos transmitidos entre el emisor y receptor.

3.4. Métodos de estudio

Para la realización de ésta investigación se utilizará el método científico, donde dicho método se refiere a una serie de etapas que hay que recorrer para obtener como resultado un conocimiento válido desde el punto de vista científico, utilizando para éste método instrumentos que resulten fiables, los mismos que constan de las siguientes etapas: planteamiento del problema, formulación de la hipótesis, levantamiento de la información, análisis e interpretación de resultados, comprobación de la hipótesis y difusión de resultados.

3.5. Técnicas de estudio

En la presente investigación se utilizarán las siguientes técnicas:

3.5.1. Técnica de estudio de búsqueda de información

Esta técnica permite obtener la información necesaria y suficiente acerca del objeto de estudio de la investigación para su respectivo desarrollo, utilizando fuentes secundarias disponibles actualmente.

3.5.2. Técnica de estudio de pruebas

Por medio de esta técnica se logrará realizar experimentos en los escenarios de laboratorio.

- En el primer experimento de prueba se realizará sin ninguna encriptación de datos.
- Por medio del segundo escenario como experimento de prueba se utilizará un algoritmo criptográfico para la seguridad de la transmisión de la información.

3.5.3. Técnica de estudio de observación

Con ésta técnica se podrá determinar los resultados de las pruebas realizadas en los escenarios de laboratorio de las técnicas de estudio de prueba.

- Al realizar el primer experimento de prueba y sin ninguna encriptación de datos, se observó que el receptor no obtiene la misma información que fue enviada por el emisor.
- Mediante el segundo escenario como experimento de prueba utilizando un algoritmo criptográfico se verificó la seguridad de la información al ser transmitida.

3.6. Instrumentos de estudio

Las principales fuentes que serán utilizadas en el estudio de investigación serán:

3.6.1. Instrumentos de estudio primaria

Para las fuentes primarias tenemos el hardware utilizado

- Dispositivo electrónico arduino
- Dispositivo electrónico ethernetshield
- Dispositivo electrónico microservo

Dispositivo electrónico arduino: El arduino es una herramienta para hacer que los ordenadores puedan controlar el mundo físico a través de un ordenador personal. Es una plataforma de desarrollo de computación física, de código abierto, basada en una placa con un sencillo microcontrolador y un entorno de desarrollo para crear software (programas) para la placa.

Figura 1-3: Dispositivo electrónico arduino uno genérico
Realizado por: (Sánchez Wilian 2017)

Dispositivo electrónico ethernet shield: La ethernet shield permite a una placa Arduino conectarse a internet a través de un conector ethernet estándar RJ45 con el shield. La ethernet shield soporta hasta cuatro conexiones de socket simultáneas.

Figura 2-3: Dispositivo electrónico ethernetshield
Realizado por: (Sánchez Wilian 2017)

Dispositivo electrónico microservo: Un microservo es un motor de corriente continua con un potenciómetro que le permite saber la posición en la que se encuentra y así poder controlarla.

Figura 3-3: Dispositivo electrónico microservo
Realizado por: (Sánchez Wilian 2017)

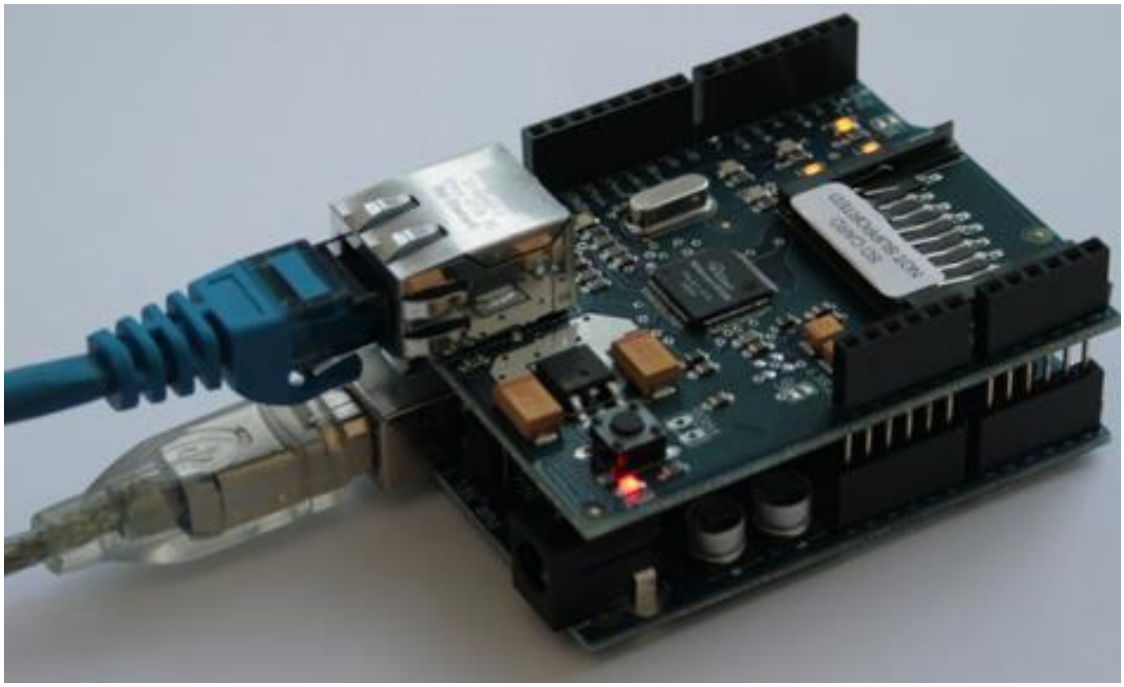


Figura 4-3: Ensamblado del dispositivo electrónico ethernet shield
Realizado por: (Sánchez Wilian 2017)

Para las fuentes primarias tenemos el software utilizado

- Sistema Operativo Virtual Kali Linux
- XAMPP
- Motor de Base de Datos MYSQL
- IDE MYSQLWORKBENCH
- Servidor Web APACHE
- Lenguaje de Programación PHP y Arduino
- Algoritmo criptográfico seleccionado HASH (SHA-2)
- Pruebas, Observación de resultados.

Sistema operativo virtual kali Linux: Éste sistema operativo, nos ayuda a realizar la penetración del ataque para la manipulación de la información dentro de una base de datos.

Figura 5-3: Sistema operativo kali linux
Realizado por: (Sánchez Wilian 2017)

XAMPP: Por medio de esta herramienta de software podemos ejecutar los servicios mySql, servidor apache.

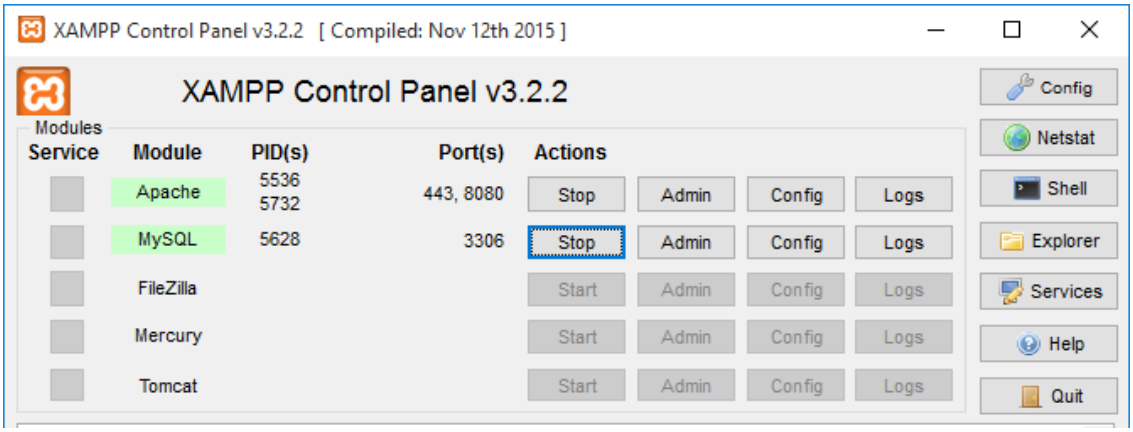


Figura 6-3: Herramienta xampp
Realizado por: (Sánchez Wilian, 2017)

Motor de base de datos mysql: Mysql es un sistema de gestión de base de datos de código abierto, confiabilidad y facilidad de uso comprobados, es muy utilizado para aplicaciones basadas en la Web

Figura 7-3: Motor base de datos mysql
Realizado por: (Sánchez Wilian, 2017)

IDE MYSQL WORKBENCH

Es una herramienta de entorno de desarrollo gráfico para la creación de base de datos, creación de tablas, funciones y/o procedimientos almacenados.

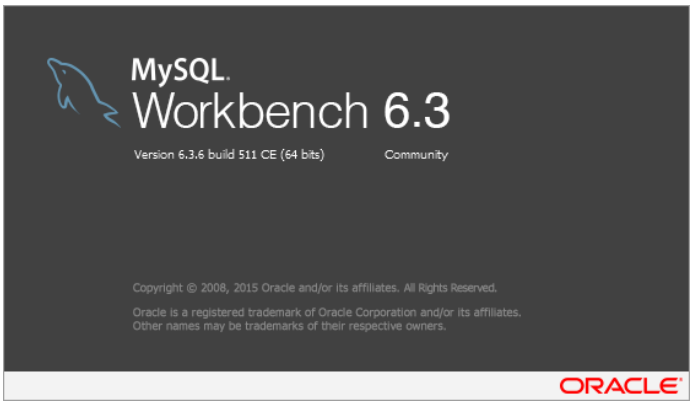


Figura 8-3: Ide mysql workbench
Realizado por: (Sánchez Wilian, 2017)

Servidor web apache: El servidor apache, es un servidor web HTTP de código abierto utilizado crear páginas y servicios web. Es un servidor multiplataforma, gratuito, muy robusto y que destaca por su seguridad y rendimiento.

Figura 9-3: Servidor web apache
Realizado por: (Sánchez Wilian, 2017)

Lenguaje de programación php: Hypertext Preprocessor o PHP es un lenguaje de código abierto, adecuado para el desarrollo web y que puede ser incrustado en HTML.

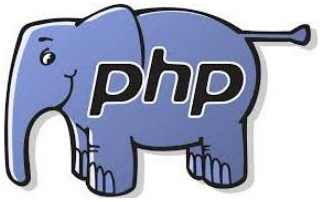


Figura 10-3: Lenguaje de programación php
Realizado por: (Sánchez Wilian, 2017)

Lenguaje de programación arduino: Este software permite programar en la placa del Arduino para realizar movimientos derecha e izquierda de un dispositivo electrónico microservo.

Figura 11-3: Lenguaje de programación arduino
Realizado por: (Sánchez Wilian, 2017)

3.6.2. Instrumentos de estudio secundaria

En las fuentes secundarias tenemos:

- Páginas de internet que brinden información confiable.
- Diccionarios especializados.
- Conferencias académicas, congresos, seminarios.
- Revistas indexadas y no indexadas publicadas de prestigio.
- Revistas electrónicas.
- Tesis realizadas internacionales y nacionales de cuarto nivel.
- Trabajos de investigaciones internacionales y nacionales.
- Artículos científicos en base de datos de bibliotecas virtuales.
- Libros especializados en la biblioteca y electrónicos.

3.7. Aplicación del método

Mediante la ejecución de transferencia de datos existe un nivel alto de incertidumbre por las diferentes modificaciones que éstas pueden sufrir, llegando así a la inseguridad de los sistemas informáticos.

En la presente propuesta de ésta investigación se utilizará un algoritmo de encriptación para mitigar la inseguridad en la transmisión de información, mediante ataques inducidos a la aplicación del software.

Se utilizará el algoritmo de encriptación SHA-2, el mismo que fue seleccionado en el apartado Capítulo II. Sección 2.7. (análisis de selección del algoritmo criptográfico), el cual permitirá mitigar los ataques inducidos hacia la aplicación de software.

3.8. Definición de los escenarios de prueba

Ambiente de pruebas: Se utiliza un ambiente de pruebas común donde se comparten los escenarios para el análisis en la utilización de un algoritmo criptográfico para la comunicación humano-robot. Las condiciones del ambiente de pruebas para los 2 escenarios son: aplicación cifrada, velocidad de encriptación de información, vulnerabilidad de aplicaciones web.

Escenarios: En el ambiente de pruebas se definen dos escenarios:

Escenario 1: En el primer escenario se utilizará el Prototipo I - Algoritmo Criptográfico AES-256, que utilizará la con Procesador Intel y ADM para obtener las velocidades de cifrado y la utilización de la Herramienta OpenSSL para la integridad de la información.

Escenario 2: En el segundo escenario se utilizará el Prototipo II - Algoritmo Criptográfico SHA-256, que utilizará la con Procesador Intel y ADM para obtener las velocidades de cifrado y la utilización de la Herramienta OpenSSL para la integridad de la información.

Resultados: Los resultados se obtendrán mediante las pruebas en los dos escenarios planteados con la finalidad de demostrar la mejora de la seguridad en la comunicación de información; donde el Prototipo I y Prototipo II incluye la ejecución de la aplicación con el programa desarrollado, además se considera directrices brindadas en la guía de mejores prácticas para programación en PHP y Arduino.

3.9. Variables e indicadores

Mediante la hipótesis para la investigación planteada “La utilización de un algoritmo de encriptación aplicado a la comunicación humano-robot permitirá mejorar la integridad de los datos que controlan tanto los movimientos mecánicos como la información que proporciona el robot”, se determinan las siguientes variables:

Variable independiente: Algoritmo de encriptación

Variable dependiente: Seguridad en la transmisión de datos

Tabla 1-3: Operacionalización conceptual

VARIABLE	TIPO	CONCEPTO
Algoritmo de encriptación	Independiente	Algoritmo criptográfico SHA-256 utilizado para cifrar información. Algoritmo de encriptación que será incorporado a la transmisión de información entre humano/robot.
Seguridad en la transmisión de datos	Dependiente	Garantizar confidencialidad, integridad y disponibilidad de la información en la transmisión de datos por cualquier medio.

Realizado por: Sánchez Wilian, 2017

Tabla 2-3: Operacionalización metodológica

VARIABLE	CATEGORÍA	INDICADOR	TÉCNICA	FUENTE
Algoritmo de encriptación	Criptografía	Complejidad	Búsqueda de información y fuentes	Navegadores, Internet
		Líneas de código	Pruebas	IDE
		Recursos utilizados	Observación	IDE
Seguridad en la transmisión de datos	Medio de Transmisión	Integridad	Pruebas	IDE
		Disponibilidad	Simulación	IDE
		Confidencialidad	Observación	IDE

Realizado por: Sánchez Wilian, 2017

3.10. Análisis de variables

Para el análisis de las variables en esta investigación se propone un algoritmo criptográfico SHA-256 para mejorar el nivel de seguridad en la transmisión de datos entre el emisor y receptor.

3.10.1. Indicadores de la variable independiente

Al realizar la comprobación de la hipótesis planteada se aplicará un algoritmo de encriptación donde la información será transmitida entre emisor y receptor la misma que deberá proporcionar integridad en la transmisión de los datos mediante los respectivos movimientos izquierda y derecha por medio del dispositivo electrónico microservo.

3.10.2. Indicadores de la variable dependiente

Para establecer si se logra mejorar la seguridad en la transmisión de datos con el algoritmo de encriptación seleccionado se analizará los resultados obtenidos con las respectivas pruebas de laboratorio realizadas, pudiendo así obtener una transmisión de datos confiables entre el emisor y receptor.

CAPÍTULO IV

4. RESULTADOS Y DISCUSIÓN

4.1. Procedimiento general

El algoritmo de encriptación SHA-256 es utilizado para la comunicación entre el emisor y receptor logrando niveles de seguridad en la trasmisión de información. Se analizaron los algoritmos criptográficos seguros y difíciles de ser vulnerados con el fin de utilizar el más apropiado; mediante sus respectivas ventajas, tiempo de procesamientos.

El procedimiento se realiza a partir de una aplicación de software impresa en la placa del arduino, que es controlado por medio de ingreso de datos proporcionados por el usuario hacia la aplicación para obtener los movimientos derecha e izquierda de un dispositivo electrónico microservo donde se aplica el algoritmo criptográfico SHA-256 y se realizaron pruebas de ataque con el algoritmo funcionando y sin el algoritmo.

Para todas las pruebas se empleó el sistema operativo virtual kali linux y dentro de él, se utilizó la herramienta de sqlmap para realizar la penetración al sistema de la víctima logrando obtener información del motor de base de datos mysql; la explotación de la información se ejecutará mediante la aplicación del arduino por intermedio del puerto serial monitor.

4.2. Presentación de resultados

Los resultados obtenidos demuestran de una manera directa que el uso de la propuesta ayudaría a mejorar la seguridad, confiabilidad, disponibilidad e integridad de la información transmitida entre el emisor y receptor.

4.3. Demostración de la hipótesis

4.3.1. *Planteamiento*

Mediante un algoritmo de encriptación aplicado a la comunicación humano-robot permitirá mejorar la seguridad de los datos que controlen los movimientos izquierda y derecha de un

dispositivo electrónico microservo, así como la información que proporciona la aplicación informática.

4.3.2. *Población*

La población de estudio para validar la muestra está constituida por diez pruebas de laboratorio (algoritmo de encriptación) mediante ataques inducidos a la seguridad de datos transmitidos entre el emisor y receptor.

4.3.3. Selección del nivel de significación

El nivel de significación en esta investigación es de $\infty = 0.05$

4.3.4. Especificación del estadístico

Para la distribución de la población se realizó con Chi Cuadrado y así lograr obtener mayor seguridad en la transmisión de información de datos con un nivel de significancia de 5%.

4.4. Comprobación de hipótesis

Para la prueba de hipótesis planteada se utilizó la prueba chi cuadrado o X^2, que es una prueba no paramétrica a través de la cual se mide la relación entre la variable dependiente e independiente. La distribución chi-cuadrado depende de desviaciones independientes, grados de libertad y ésta no puede ser negativa.

Hipótesis nula H_0:

El algoritmo criptográfico seleccionado y utilizado en el software informático para movimientos izquierda y derecha de un dispositivo electrónico microservo; no permite mitigar ataques a la información de datos.

Hipótesis alternativa H_1:

El algoritmo criptográfico seleccionado y utilizado en el software informático para movimientos izquierda y derecha de un dispositivo electrónico microservo; permite mitigar ataques a la información de datos.

Tabla 1-4: Velocidad de cifrado del algoritmo criptográfico

VELOCIDAD DE CIFRADO Kb/s – BLOQUE 256 bits				
N°	AES-256		SHA-256	
	VELOCIDAD PROCESADOR INTEL	VELOCIDAD PROCESADOR AMD	VELOCIDAD PROCESADOR INTEL	VELOCIDAD PROCESADOR AMD
1	3576	3845	3589	3859
2	3879	3746	3946	3811
3	3998	3846	5772	5552
4	3897	3759	3904	3765
5	3901	4015	3947	4062
6	4097	4111	4161	4175
7	4120	4356	4090	4324
8	3987	3968	4353	4332
9	3756	3869	3943	4061
10	3658	3900	3671	3914

Realizado Por: Sánchez Wilian, 2017

La tabla de contingencia de frecuencias esperadas son los valores que se esperaría encontrar si las variables no estuvieran relacionadas. Chi cuadrado parte del supuesto de "no relación entre las variables" y se evaluará si es cierto o no, analizando si sus frecuencias observadas son diferentes de lo que pudiera esperarse en caso de ausencia de correlación. La frecuencia esperada de cada celda, se calcula mediante la siguiente fórmula aplicada a la tabla de frecuencias observadas.

$$fe = \frac{(TotalColumna) * (Total\ Fila)}{N}$$

Dónde:

N: Número total de frecuencias observadas.

Aplicando la fórmula a los valores de la Tabla 4.1 se obtiene la tabla de contingencia de valores esperados, como se muestra en la Tabla 4.2.

Tabla 2-4: Tabla de contingencia de frecuencias esperadas de velocidad de encriptación

VELOCIDAD DE CIFRADO Kb/s – BLOQUE 256 bits					
N°	AES-256		SHA-256		TOTAL
	PROCESADOR INTEL	PROCESADOR AMD	PROCESADOR INTEL	PROCESADOR AMD	
1	353,092509	362,103724	384,263957	384,53981	**1484**
2	365,465023	374,791995	397,728732	398,014251	**1536**
3	456,117479	467,757978	496,384101	496,740442	**1917**
4	364,037425	373,327963	396,175104	396,459508	**1530**
5	378,551335	388,212281	411,97032	412,266064	**1591**
6	404,010162	414,320838	439,676684	439,992317	**1698**
7	401,630832	411,880786	437,087304	437,401078	**1688**
8	385,451391	395,288432	419,479522	419,780656	**1620**
9	371,65128	381,13613	404,461119	404,751472	**1562**
10	359,992565	369,179875	391,773158	392,054402	**1513**
TOTAL	**3840**	**3938**	**4179**	**4182**	**16139**

Realizado Por: Sánchez Wilian, 2017

Una vez obtenida la tabla de frecuencias esperadas, se aplica la siguiente fórmula de chi cuadrado.

$$X^2 = \sum_{j=1}^{k} \frac{(f_0 - f_e)^2}{f_e}$$

Dónde:

f_o: Frecuencia observada en cada celda

f_e: Frecuencia esperada en cada celda

Para realizar la comparación de los resultados obtenidos se utilizará la escala de Likert para la integridad de información aplicándole al certificado digital SSL.

Para la escala definida de las velocidades de encriptación con la integridad de la información mediante el certificado digital SSL está basada en que la relación de la seguridad es inversamente proporcional a las velocidades de encriptación de los datos a ser transmitidos.

Tabla 3-4: Tabla de escala para la integridad de la información con certificado digital SSL

VELOCIDAD DE CIFRADO Kb/s	VALOR	DESCRIPCIÓN INTEGRIDAD
0.01 kb/s … 1.00 kb/s	4	ALTA
1.01 kb/s … 2.00 kb/s	3	MEDIA
2.01 kb/s … 3.00 kb/s	2	BAJA
> 3.00 kb/s	1	INSUFICIENTE

Realizado Por: Sánchez Wilian, 2017

Mediante la escala de valores para la integridad de la información con certificado digital SSL, descrita en la Tabla N° 3-4., se obtiene los respectivos valores de integridad para cada velocidad de cifrado en la transmisión de la información dando como resultado que entre más rápido la velocidad de cifrado se ejecute mayor es la integridad de la información en la transmisión de datos.

Tabla 4-4: Calculo de chi-cuadrado – SHA-256

VELOCIDAD DE CIFRADO Kb/s – BLOQUE 256 bits – SEGURIDAD E INTEGRIDAD								
N°	**AES-256**				**SHA-256**			
	PROCESADOR INTEL		**PROCESADOR AMD**		**PROCESADOR INTEL**		**PROCESADOR AMD**	
	VELOCIDAD	**INTEGRIDAD**	**VELOCIDAD**	**INTEGRIDAD**	**VELOCIDAD**	**INTEGRIDAD**	**VELOCIDAD**	**INTEGRIDAD**
1	0,0432	4	1,3241	3	1,7951	2	0,0006	1
2	1,2689	2	0,0017	1	0,0350	1	0,7273	1
3	7,1526	1	14,2902	4	13,0925	4	6,8329	4
4	1,7117	2	0,0075	1	0,0963	1	1,0558	2
5	0,3462	4	0,4212	1	0,7839	1	0,0952	1
6	0,0616	4	0,0266	1	1,0341	2	1,2015	2
7	0,2677	4	1,2977	2	1,8049	2	0,0667	1
8	2,2503	3	0,0013	1	0,5742	1	0,4163	1
9	0,0509	1	0,0621	1	0,2706	1	0,0039	1
10	0,0697	1	1,1742	2	1,5665	1	0,0028	1
X^2							**137,2855**	

Realizado Por: Sánchez Wilian, 2017

Interpretación

Para determinar si el valor de X^2 es o no significativo, se debe determinar los grados de libertad mediante la siguiente fórmula.

$$GL = (f - 1)(c - 1)$$

Dónde:

f: Número de filas de la tabla de continencia

c: Número de columnas de la tabla de contingencia

Por lo tanto:

$$GL = (10 - 1)(8 - 1) = (9)(7) = 63$$

4.5. Conclusión de la hipótesis

Mediante la tabla de distribución X^2 y eligiendo como nivel de significancia de α=5% equivalente a α=0.05 para obtener un nivel de confianza del 95%, se obtiene como punto crítico de X^2 para 63 grados de libertad $X^2_{Crìtico}$ = 79.0820.

El valor $X^2_{calculado}$ en esta investigación es de 137,2855 que es superior al valor de la tabla de distribución de 79.0820, por lo que el valor calculado de $X^2_{calculado}$ se encuentra en el sector de rechazo de la hipótesis nula H_0, y se acepta la hipótesis H_1 de investigación que es significativa, con un nivel de significancia de α=5% equivalente a α=0.05 para obtener un nivel de confianza del 95%.

CAPÍTULO V

5. PROPUESTA

5.1. Introducción

En el presente capítulo se presenta la utilización de un algoritmo de encriptación aplicado a la comunicación humano-robot.

5.2. Objetivos

- Obtener mayor seguridad cuando se envía y se recibe información aplicados al software para los movimientos izquierda y derecha de un dispositivo electrónico microservo.
- Enviar y recibir datos de manera confiable, el o los usuarios correspondientes.

5.3. Descripción del método propuesto

5.3.1. Descripción general del método HASH

Mediante la utilización del algoritmo criptográfico SHA-256 aplicado al dispositivo electrónico microservo, obtendremos un nivel de seguridad aceptable para la transferencia de datos.

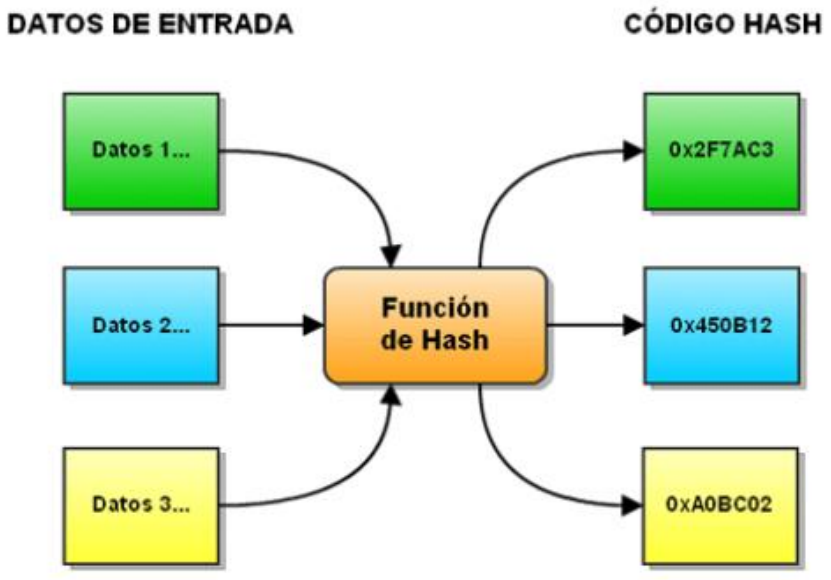


Figura 1-5: Funcionamiento del método hash
Realizado por: (Michael D. Bauer, 2001).

5.3.2. Descripción del escenario con el método propuesto

En síntesis, la propuesta de la utilización del método criptográfico aplicado a comunicación humano-robot consiste en seleccionar un algoritmo criptográfico a utilizar en un software para los movimientos izquierda y derecha de un dispositivo electrónico microservo ejecutando un ataque inducido. El usuario puede interactuar con el software para realizar los movimientos izquierda y derecha de un dispositivo electrónico microservo aplicando el algoritmo criptográfico al sistema informático para la transmisión de la información, el mismo que fue implementado con el lenguaje de programación arduino, motor de base de datos mysql, lenguaje de programación PHP, y además se creó un certificado de seguridad digital SSL para la integridad de la información.

5.3.2.1. *Encriptación de los movimientos del microservo*

Creamos el código fuente que se imprimirá en la placa del arduino, llamamos a las librerías y creamos las variables básicas

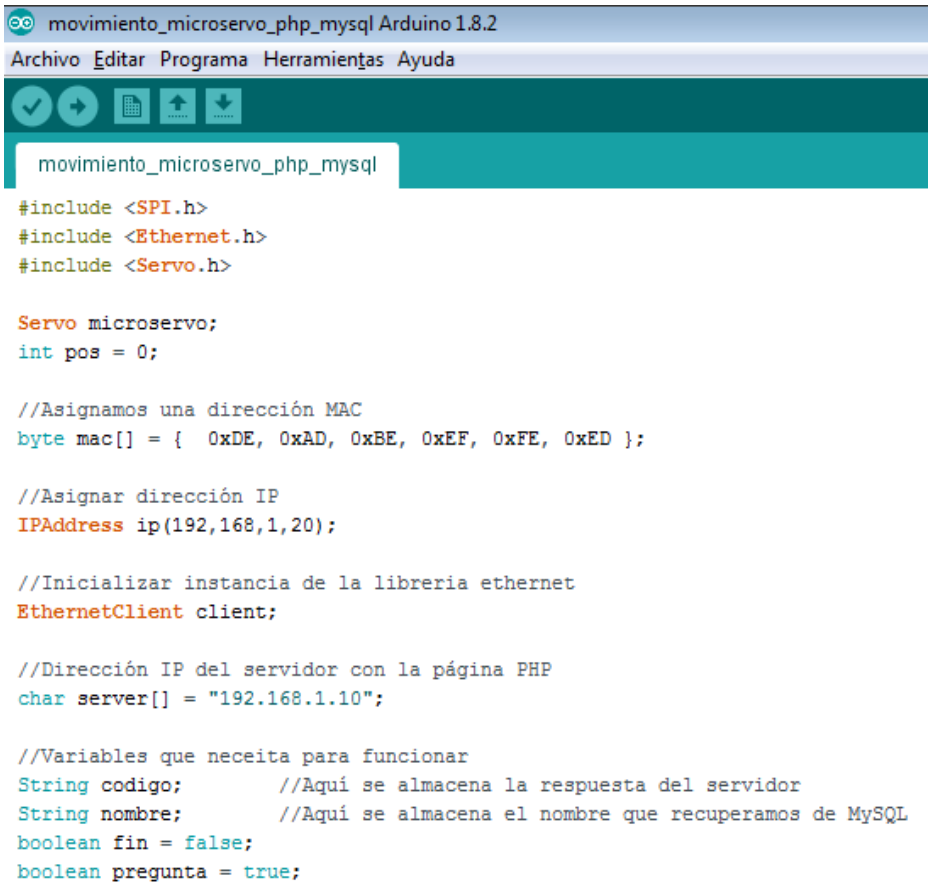


Figura 2-5: Inicialización de variables y liberarías
Realizado por: Sánchez Wilian, 2017

Inicializamos las variables creadas mediante la función setup()

Figura 3-5: Inicialización del puerto serie
Realizado por: Sánchez Wilian, 2017

Creamos un bucle infinito mediante la función loop(), para pedir al usuario el ingreso del movimiento izquierda o derecha que desea realizar

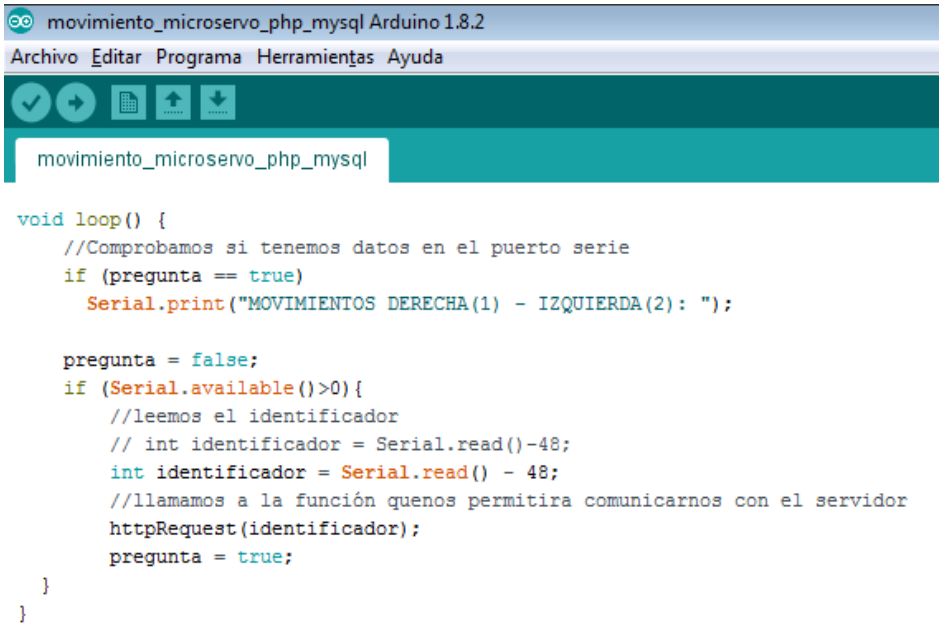
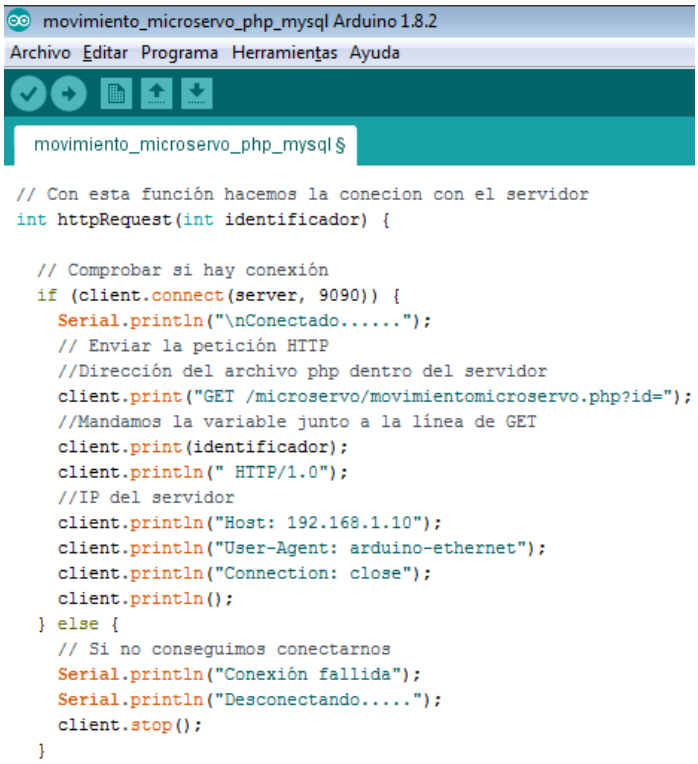


Figura 4-5: Ingreso de movimiento
Realizado por: Sánchez Wilian, 2017

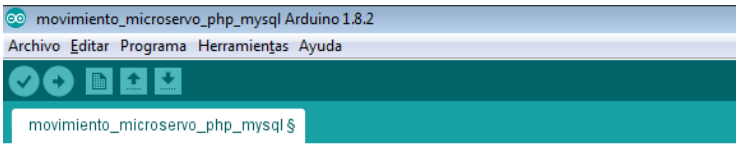
Creamos una función para verificar si tenemos conexión con el servidor de php.




```
// Con esta función hacemos la conecion con el servidor
int httpRequest(int identificador) {

  // Comprobar si hay conexión
  if (client.connect(server, 9090)) {
    Serial.println("\nConectado.....");
    // Enviar la petición HTTP
    //Dirección del archivo php dentro del servidor
    client.print("GET /microservo/movimientomicroservo.php?id=");
    //Mandamos la variable junto a la línea de GET
    client.print(identificador);
    client.println(" HTTP/1.0");
    //IP del servidor
    client.println("Host: 192.168.1.10");
    client.println("User-Agent: arduino-ethernet");
    client.println("Connection: close");
    client.println();
  } else {
    // Si no conseguimos conectarnos
    Serial.println("Conexión fallida");
    Serial.println("Desconectando.....");
    client.stop();
  }
```

Figura 5-5: Conexión con el servidor
Realizado por: Sánchez Wilian, 2017

Extraemos información del servidor conectado




```
delay(500);
//Comprobamos si tenemos respuesta del servidor y la
//almacenamos en el string ----> codigo.
while (client.available()) {
  char c = client.read();
  codigo += c;
  //Habilitamos la comprobación del código recibido
  fin = true;

}
//Si está habilitada la comprobación del código entramos en el IF
if (fin)  {
 // Serial.println(codigo);
 //Analizamos la longitud del código recibido
  int longitud = codigo.length();
  //Buscamos en que posición del string se encuentra nuestra variable
  int posicion = codigo.indexOf("valor=");
  //Borramos lo que haya almacenado en el string nombre
  nombre = "";

  //Analizamos el código obtenido y almacenamos el nombre en el string nombre
  for (int i = posicion + 6; i < longitud; i ++){
    if (codigo[i] == ';') i = longitud;
    else nombre += codigo[i];
  }
```

Figura 6-5: Obtención de información
Realizado por: Sánchez Wilian, 2017

Comprobamos y verificamos el algoritmo criptográfico

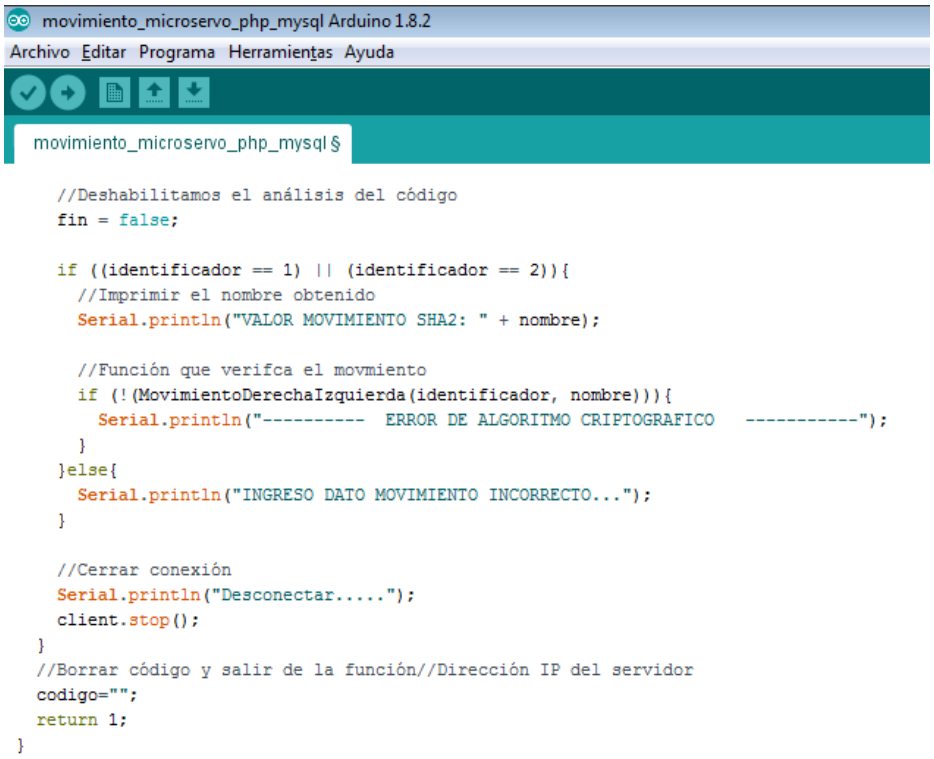


Figura 7-5: Comprobación de información obtenida
Realizado por: Sánchez Wilian, 2017

Ejecutamos el código fuente por medio del puerto serial monitor y verificamos la encriptación y manipulación de datos.

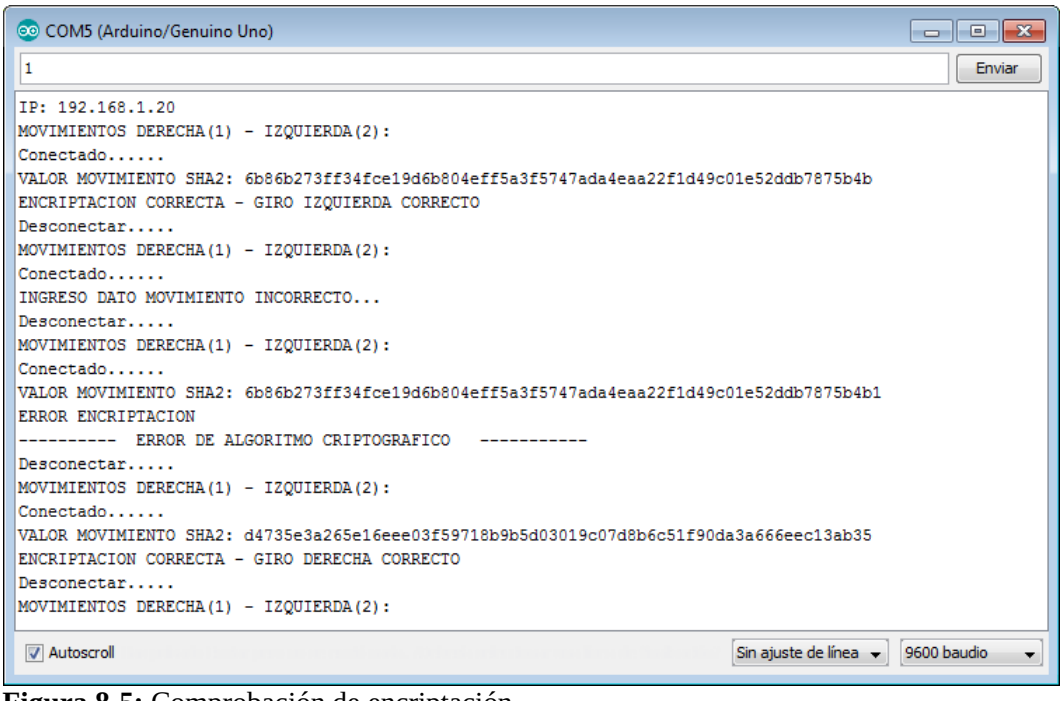


Figura 8-5: Comprobación de encriptación
Realizado por: Sánchez Wilian, 2017

5.3.2.2. Creación de un certificado digital ssl

Instalar openssl: Para instalar OpenSSL debes ejecutar el siguiente comando desde nuestro terminal: sudo apt-get install openssl

Crear una llave privada: La llave privada (1024 bits) nos servirá para generar el certificado, por ende, una vez creado, nuestro certificado SSL dependerá de esta llave para la implementación del mismo en cualquier servicio que requiera una conexión segura, para eso ejecutamos el siguiente comando en nuestro terminal: openssl genrsa -out server.key 1024.

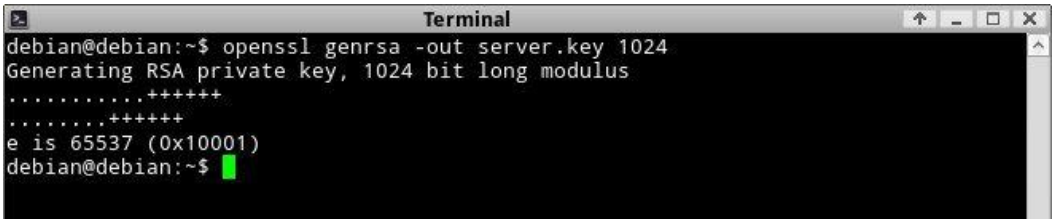


Figura 9-5: Creación de una llave privada
Realizado por: Sánchez Wilian, 2017

Crear un csr (certificate signing request): Un CSR es la base para un certificado SSL, en él se definen datos como el dominio, organización, ubicación, información de contacto, entre otros. Para generar el CSR debes ejecutar el siguiente comando desde nuestro terminal: openssl req -new -key server.key -out server.csr

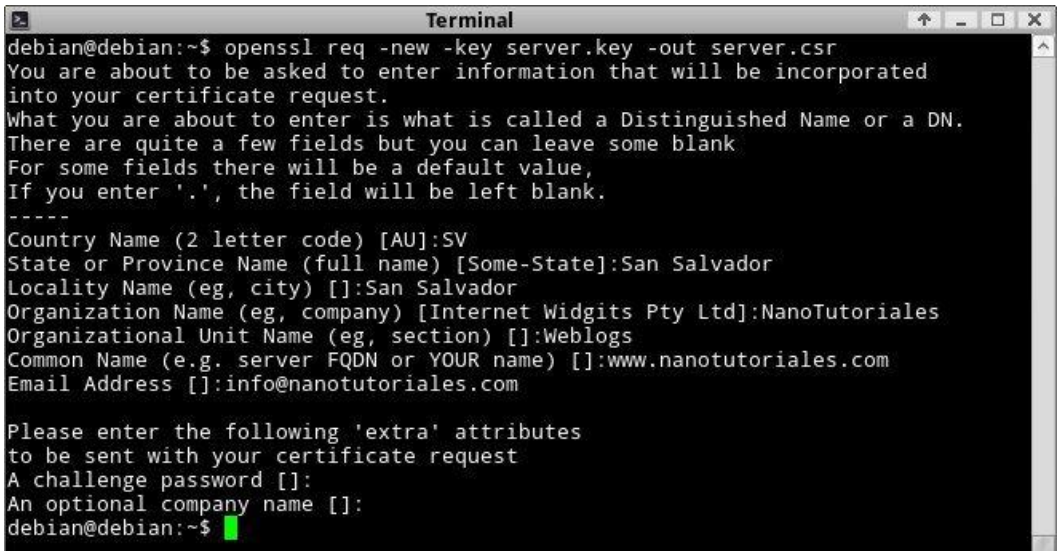


Figura 10-5: Creación de un csr
Realizado por: Sánchez Wilian, 2017

Al ejecutar este comando tendremos varias preguntas las mismas que debemos llenar para poder crear nuestro CSR:

- **Country Name (2 letter code):** Código de país en formato ISO de dos letras.
- **State or Province Name (full name):** Estado o provincia.
- **Locality Name:** Localidad o ciudad.
- **Organization Name:** Nombre de la organización.
- **Organizational Unit Name:** Sector de la organización.
- **Common Name:** Nombre del dominio ó FQDN.
- **Email Address:** Dirección de correo de contacto. Se puede dejar en blanco los campos A challenge password y An optional company name, los mismos que no se van a utilizar.

Generando el certificado SSL: Para generar el certificado SSL vamos a necesitar tanto la llave privada como el CSR que acabamos de crear. Para generar el certificado SSL debemos ejecutar el siguiente comando desde el terminal: El parámetro days sirve para definir la fecha de expiración del certificado.

openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt

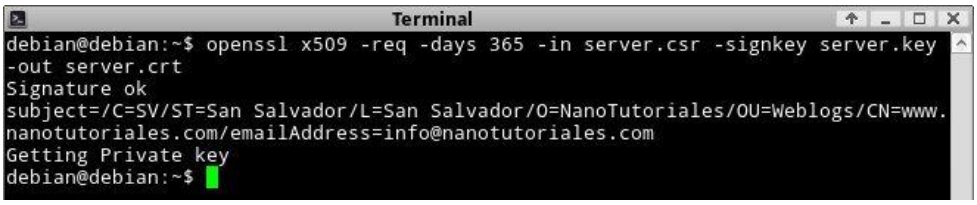


Figura 11-5: Generación del certificado ssl
Realizado por: Sánchez Wilian, 2017

Configurar el certificado ssl en apache: Primero vamos a copiar los archivos a la carpeta **/etc/ssl/certs**.

sudo cp server.crt /etc/ssl/certs/ssl.crt sudo cp server.key /etc/ssl/certs/ssl.key

Luego vamos a habilitar el módulo de SSL en Apache: sudo a2enmod ssl

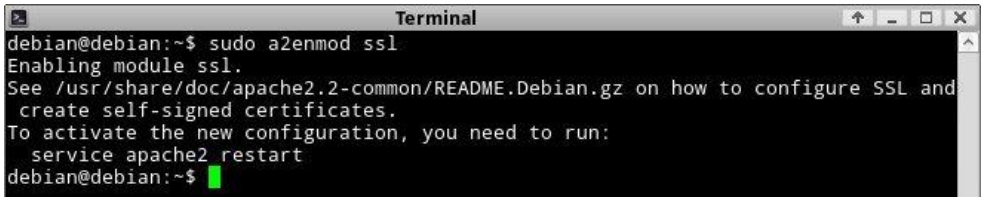


Figura 12-5: Configuración del certificado ssl en apache
Realizado por: Sánchez Wilian, 2017

Ahora vamos a editar el archivo **vhosts** con el editor de texto nano.

sudo nano /etc/apache2/sites-available/default-ssl

Borramos todo el contenido del archivo, luego copia y pega lo siguiente información:

```
<IfModule mod_ssl.c>
  <VirtualHost _default_:443>
    ServerName www.microservo.com
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www
    <Directory />
       Options FollowSymLinks
       AllowOverride None
    </Directory>
    <Directory /var/www/>
      Options -Indexes FollowSymLinks MultiViews
      AllowOverride None
      Order allow,deny
      allow from all
    </Directory>
    ErrorLog ${APACHE_LOG_DIR}/error.log
    LogLevel warn
    CustomLog ${APACHE_LOG_DIR}/ssl_access.log combined
    SSLEngine on
    SSLCertificateKeyFile /etc/ssl/certs/ssl.key
    SSLCertificateFile /etc/ssl/certs/ssl.crt
    #SSLCACertificateFile /etc/ssl/certs/bundle.crt
    BrowserMatch "MSIE [2-6]" \
      nokeepalive ssl-unclean-shutdown \
      downgrade-1.0 force-response-1.0
    # MSIE 7 and newer should be able to use keepalive
    BrowserMatch "MSIE [17-9]" ssl-unclean-shutdown
  </VirtualHost>
</IfModule>
```

Una vez configurado el certificado en el archivo **vhost**, debemos habilitarlo con el siguiente comando:

```
sudo a2ensite default-ssl
```

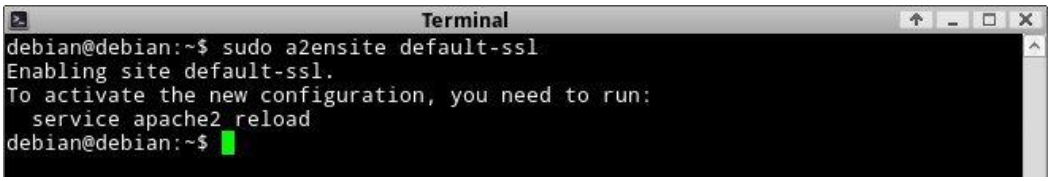


Figura 13-5: Habilitación del certificado ssl
Realizado por: Sánchez Wilian, 2017

Para que la configuración tenga efecto, vamos a cargarla nuevamente el servicio de apache. Para esto utilizaremos el siguiente comando desde nuestro terminal:

```
sudo service apache2 reload
```

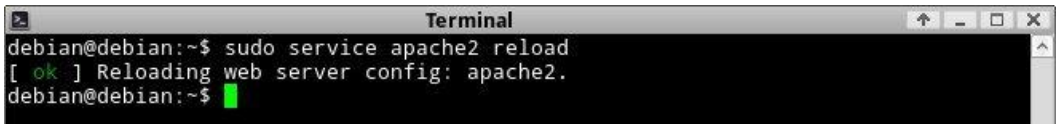


Figura 14-5: Reinicio del servicio de apache
Realizado por: Sánchez Wilian, 2017

Verificamos el certificado ssl desde un navegador: Para esto vamos a escribir **https://** seguido del dominio que hemos configurado en el certificado (https://www.microservo.com).

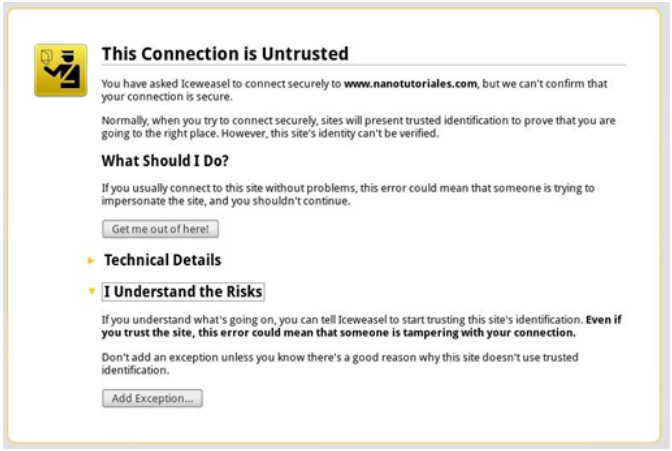


Figura 15-5: Verificación del certificado ssl
Realizado por: Sánchez Wilian, 2017

La primera vez que ingresemos nos va a dar una alerta de que el certificado no es de confianza, esto es por no encontrar una identificación de un proveedor autorizado, damos clic en entiendo el riesgo y luego en añadir una excepción.

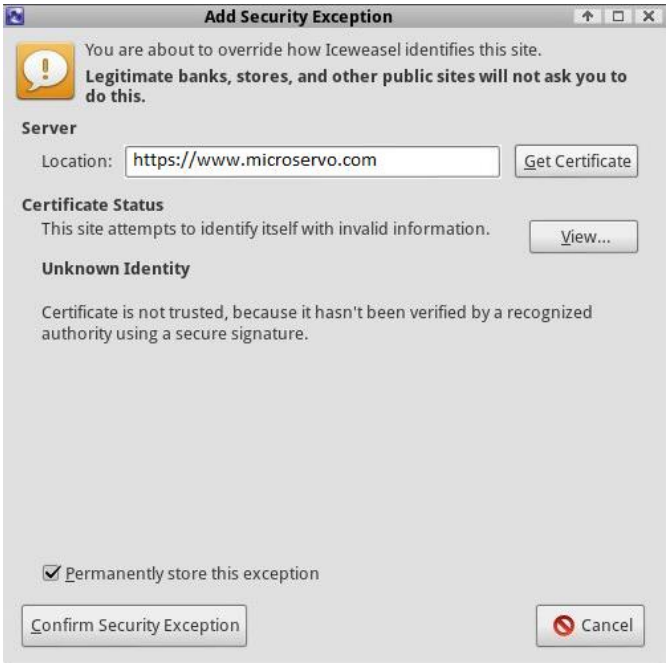


Figura 16-5: Excepción de seguridad del certificado ssl
Realizado por: Sánchez Wilian, 2017

Debemos dar clic en confirmar excepción de seguridad y listo, la comunicación con nuestro dominio es segura (siempre y cuando la conexión se mantenga bajo https://), podremos observar el candado en la barra de navegación que lo confirma.

Figura 17-5: Certificado de seguridad ssl generado
Realizado por: Sánchez Wilian, 2017

Para poder tener más información sobre nuestro certificado o de cualquier otro certificado de seguridad SSL, podemos ver su información dando clic en el candado, como, por ejemplo: podemos ver la información que ingresamos en el CSR.

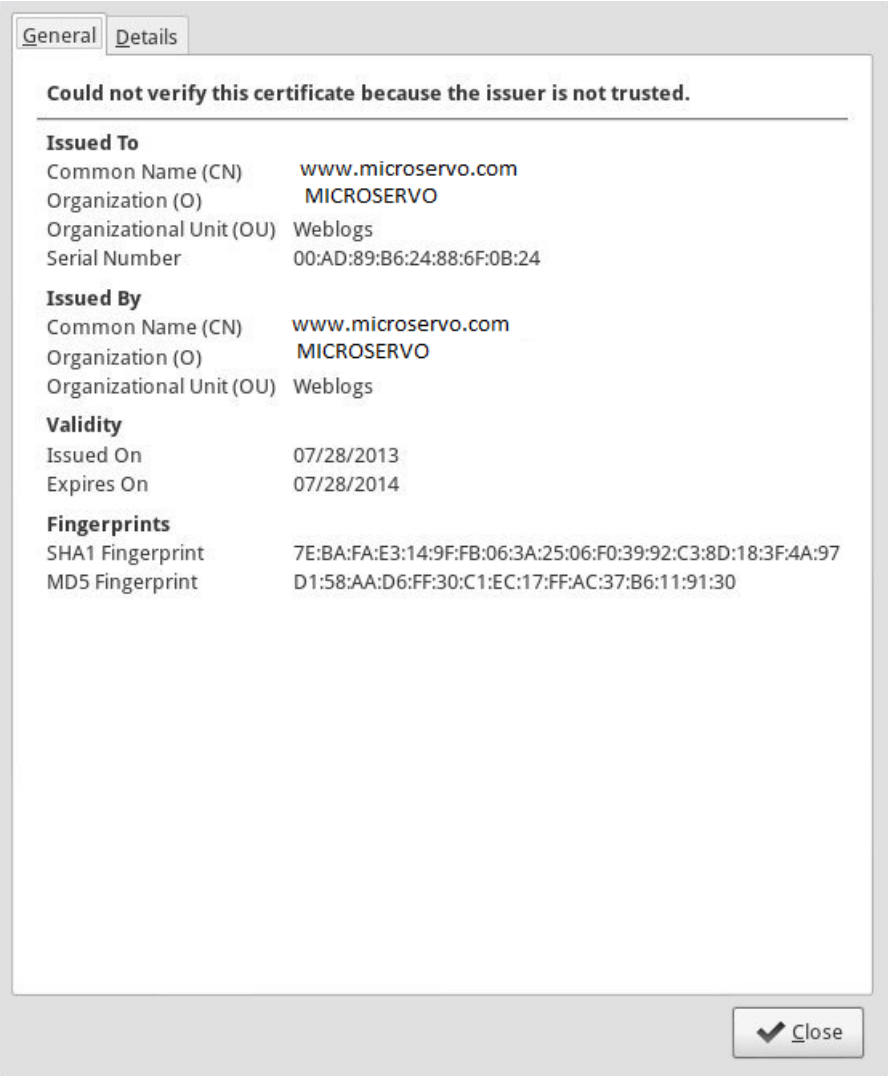


Figura 18-5: Información de csr del certificado ssl
Realizado por: Sánchez Wilian, 2017

CONCLUSIONES

- A mayor dificultad del nivel de desarrollo de un algoritmo criptográfico a ser implementado e implantado mayor será el nivel de seguridad de la información hacer transmitida por algún medio de comunicación. El nivel de seguridad aplicado con el algoritmo de encriptación AES-256 con SHA-256 se obtuvo mediante velocidades de cifrado con componentes hardware y software**.**

- Mediante la utilización de algoritmos criptográficos existentes en la actualidad se puede mejorar el nivel de seguridad de la información que será transmitida por cualquier medio de comunicación y asegurando la integridad de la información mediante la creación de certificados de seguridad SSL entre un usuario desde un navegador web y un servidor.

- El uso del algoritmo de cifrado asimétrico SHA-256 con clave de 256 bits, se aplicó a una implementación de software para mitigar los ataques realizados a la aplicación por medio de un dispositivo electrónico microservo, logrando dar seguridad a la información.

- Mientras la tecnología avanza; los nuevos ataques se fortalecen, por lo cual en la actualidad no existe un algoritmo criptográfico con máxima seguridad, por ende, al aplicar el algoritmo criptográfico AES-256 o SHA-256 el nivel de seguridad es 1,8 veces mejor respecto a la implementación del software.

- La utilización de herramientas de monitoreo de redes para realizar ataques a sistemas informáticos nos ayuda a encontrar las vulnerabilidades existentes, descubrir estándares de ataques e información sobre el atacante, y así establecer políticas de seguridad que minimizan los riesgos de hurto de información o ataques a las redes de cualquier empresa.

RECOMENDACIONES

- Para futuros trabajos de investigación se debería incorporar nuevos algoritmos criptográficos con mayores bloques de bits y así incrementar aún más la seguridad en la transferencia de información.

- Se recomienda la instalación de agentes móviles para poder generar alertas hacia dispositivos para la generación de alarmas de intrusión en las comunicaciones de información.

- El desarrollo de los sistemas de interacción humano-robot se puede realizar mediante la utilización de voz como medio de comunicación y así proporcionar la posibilidad de una comunicación más natural con los robots como trabajo futuro.

- Promover la investigación para temas de seguridad en la transmisión de datos a impulsar sobre las medidas de seguridad que debe tener las empresas al momento de enviar información**.**

- Se recomienda a futuro para este trabajo de investigación realizar un prototipo de un robot humanoide y realizarle un nivel de seguridad en los movimientos que éste pueda tener.

BIBLIOGRAFÍA

Bauer Marco (s. f.). U8.3 Funciones HASH. Recuperado 3 de marzo de 2017, a partir de http://virtual.itca.edu.sv/Mediadores/cms/u83_funciones_hash.html

Bauer Michael (2001). Funciones Hash. Recuperado 7 de marzo de 2017, a partir de http://virtual.itca.edu.sv/Mediadores/cms/u83_funciones_hash.html

Brady Michael R. P. (s. f.). Realización de un Sistema para transferencia segura de información utilizando un DSP56002 11-17 - Buscar con Google. Recuperado 3 de marzo de 2017, a partir de https://www.google.com.ec/?gfe_rd=cr&ei=lW-5WNivI4yxzQK9oap4&gws_rd=ssl#q=Realizaci%C3%B3n+de+un+Sistema+para+transferencia+segura+de+informaci%C3%B3n+utilizando+un+DSP56002+11-17&*

Cabrera Aldaya, A., Sarmiento, C., & José, A. (2013b). Diseño e integración de algoritmos criptográficos en sistemas empotrados sobre FPGA. Ingeniería Electrónica, Automática y Comunicaciones, 34(3), 41-51.

Digital_24689.pdf. (s. f.). Recuperado a partir de https://repository.upb.edu.co/bitstream/handle/20.500.11912/2104/digital_24689.pdf?sequence=1&isAllowed=y

Diego Andrés Guffanti Martínez ,2013, Control remoto por voz del robot móvil pioneerp3-dx, Tesis_Completa - T-ESPE-047250.pdf. (s. f.). Recuperado a partir de http://repositorio.espe.edu.ec/bitstream/21000/6872/1/T-ESPE-047250.pdf

Erika Aguillón Martínez, 2012, Laboratorio de Redes y Seguridad. (s. f.). Fundamentos de Criptografía. Recuperado 3 de marzo de 2017, a partir de http://redyseguridad.fi-p.unam.mx/proyectos/criptografia/criptografia/index.php/1-panorama-general/11-concepto-de-criptografia

Fahn Paúl (s. f.). Respuestas a las preguntas más frecuentes sobre encriptado. Recuperado 3 de marzo de 2017, a partir de http://www.cnv.gob.ar/infoadicional/rsafaq.htm

Fernández Diliana. (2014, julio 30). Las 5 Fases o Etapas de un Ataque Informático. Recuperado a partir de https://recordandoeinnovando.wordpress.com/2014/07/29/las-5-fases-o-etapas-de-un-ataque-informatico/

García Emanuel, F. R. (2012). Criptoanálisis y Ataques a Sistemas Criptograficos - Criptografia. Recuperado 7 de marzo de 2017, a partir de https://sites.google.com/site/isp6criptografia/criptoanalisis-y-ataques-a-sistemas-criptograficos

Guimi Arturo. (2009, mayo). 9.3 Técnicas criptográficas y de seguridad. Recuperado 6 de marzo de 2017, a partir de http://guimi.net/monograficos/G-Redes_de_comunicaciones/G-RCnode61.html

Gutiérrez Pablo. (2013, enero 15). ¿Qué son y para qué sirven los hashes?: funciones de resumen y firmas digitales. Recuperado 6 de marzo de 2017, a partir de https://www.genbetadev.com/seguridad-informatica/que-son-y-para-que-sirven-los-hash-funciones-de-resumen-y-firmas-digitales

Gutiérrez Pedro. (2013, enero 3). Tipos de criptografía: simétrica, asimétrica e hibrida. Recuperado 6 de marzo de 2017, a partir de https://www.genbetadev.com/seguridad-informatica/tipos-de-criptografia-simetrica-asimetrica-e-hibrida

Héctor A. Flores Fernández, 2012, Desarrollo de robot móvil de exploración dirigido mediante transferencia de video, HCA066SN.pdf. (s. f.). Recuperado a partir de http://www.iiisci.org/Journal/CV$/risci/pdfs/HCA066SN.pdf

Herrera Carlos, Infosegur. (2014, enero 30). Criptografía simétrica y asimétrica. Recuperado a partir de https://infosegur.wordpress.com/unidad-4/criptografia-simetrica-y-asimetrica/

Jiménez Marlon Bazán. (s. f.). 5.1 ¿Qué es un robot? Recuperado 3 de marzo de 2017, a partir de http://platea.pntic.mec.es/vgonzale/cyr_0708/archivos/_15/Tema_5.1.htm

Luis Arcos, 2012, Criptografía simétrica y asimétrica. (2012, enero 17). Recuperado a partir de http://userexception.blogspot.com/2012/01/criptografia-simetrica-y-asimetrica.html

Mariiss Saul. (2012, noviembre 15). Algoritmos de Encriptación Simétrica y Asimétrica. Recuperado a partir de https://mariiss15.wordpress.com/2012/11/14/algoritmos-de-encriptacion-simetrica-y-asimetrica/

Ribagorda Garnacho, Arturo. (s. f.). Que es la Criptografia. Recuperado 3 de marzo de 2017, a partir de http://www.informatica-hoy.com.ar/seguridad-informatica/Criptografia.php

Saravia J. V. (s. f.-a). Coordinación y control de robots móviles basado en agentes. Recuperado a partir de http://www.academia.edu/10625585/Coordinaci/n_y_control_de_robots

Urrego Jonatan. (2013, mayo). Tipos de ataque y cómo prevenirlos. Recuperado 6 de marzo de 2017, a partir de https://colombiadigital.net/actualidad/articulos-informativos/item/4801-tipos-de-ataque-y-como-prevenirlos.html

Velasco Ernesto. (2015, diciembre 5). Las mejores 10 herramientas para hacking ético de este 2015. Recuperado 3 de marzo de 2017, a partir de https://www.redeszone.net/2015/12/05/las-mejores-10-herramientas-para-hacking-etico-de-este-2015/

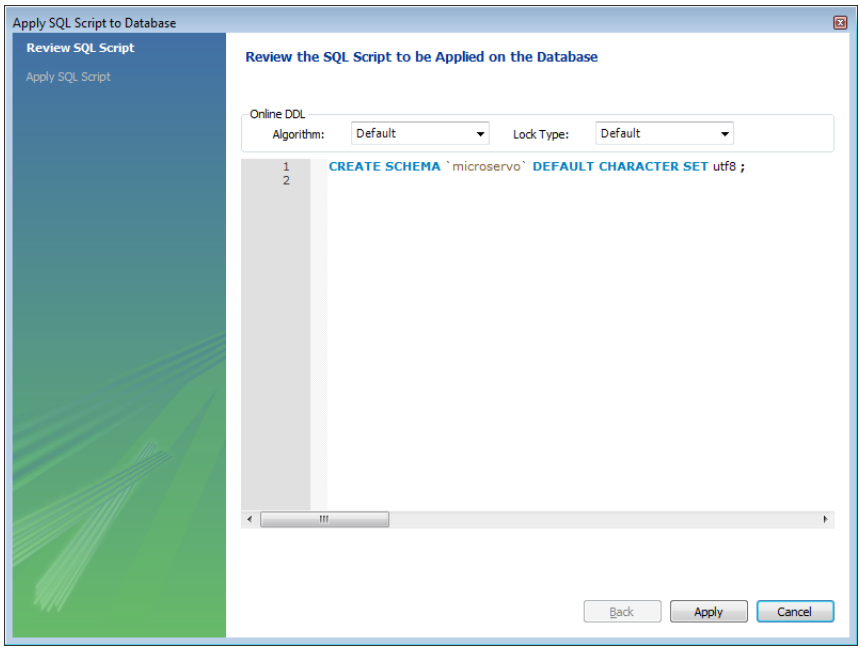
Villegas Juan. P., Madrigal, C. A., & Serna, S. (2012). ZigBee profile specification for mobile robotics. En 2012 IEEE Colombian Communications Conference (COLCOM) (pp. 1-5). https://doi.org/10.1109/ColComCon.2012.6233673

ANEXOS

Anexo A: Creación de base de datos, tablas e ingreso de registros

CREACIÓN DE BASE DE DATOS

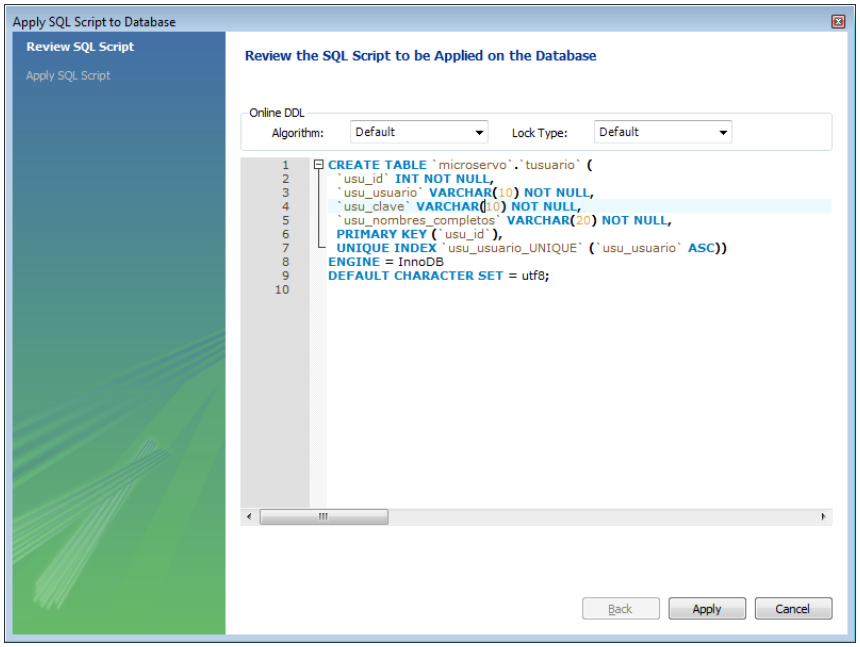
Creamos la base de datos microservo



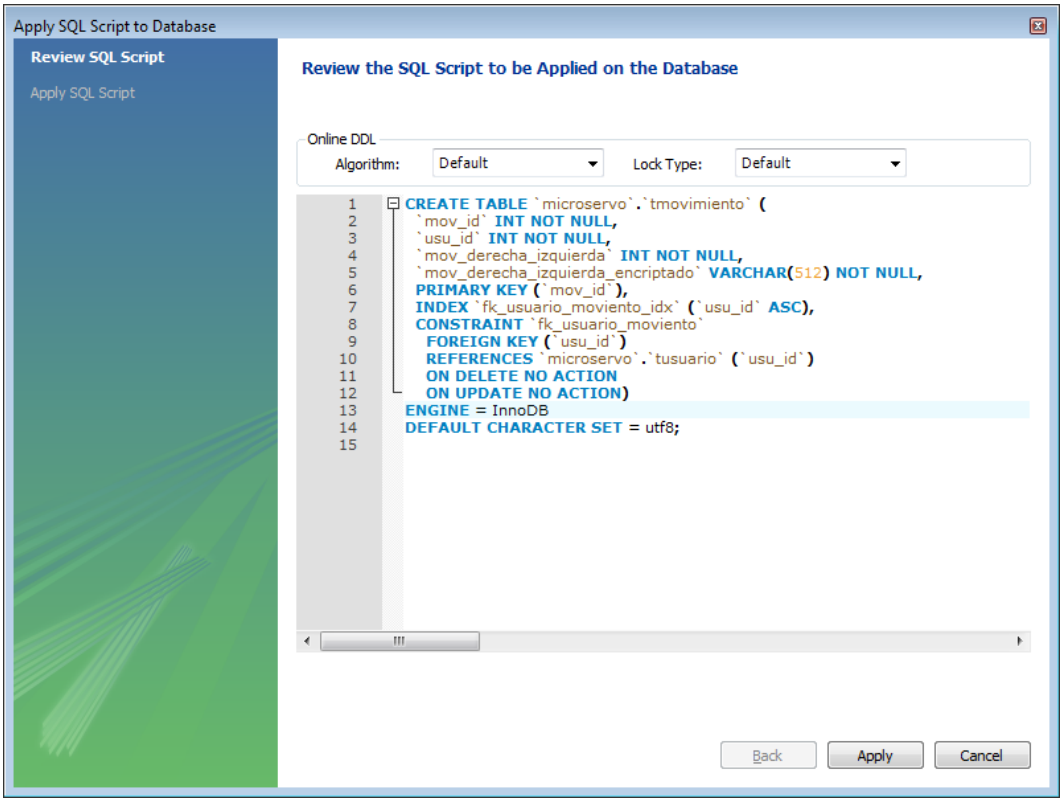
Realizado por: Sánchez Wilian, 2017

CREACIÓN DE TABLAS

Creación de la tabla tusuario

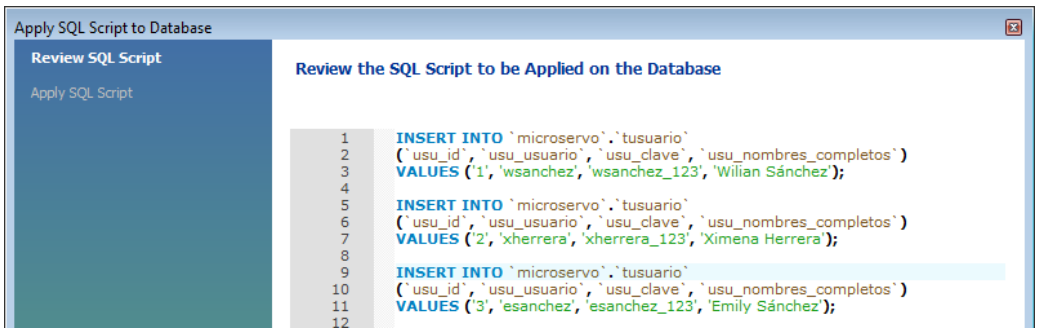


Creación de la tabla tmovimiento

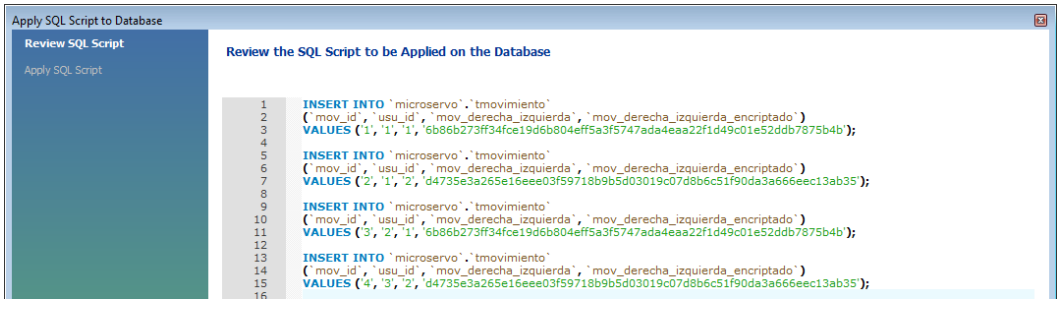


INGRESO DE REGISTROS

Ingreso de registros en la tabla tusuario

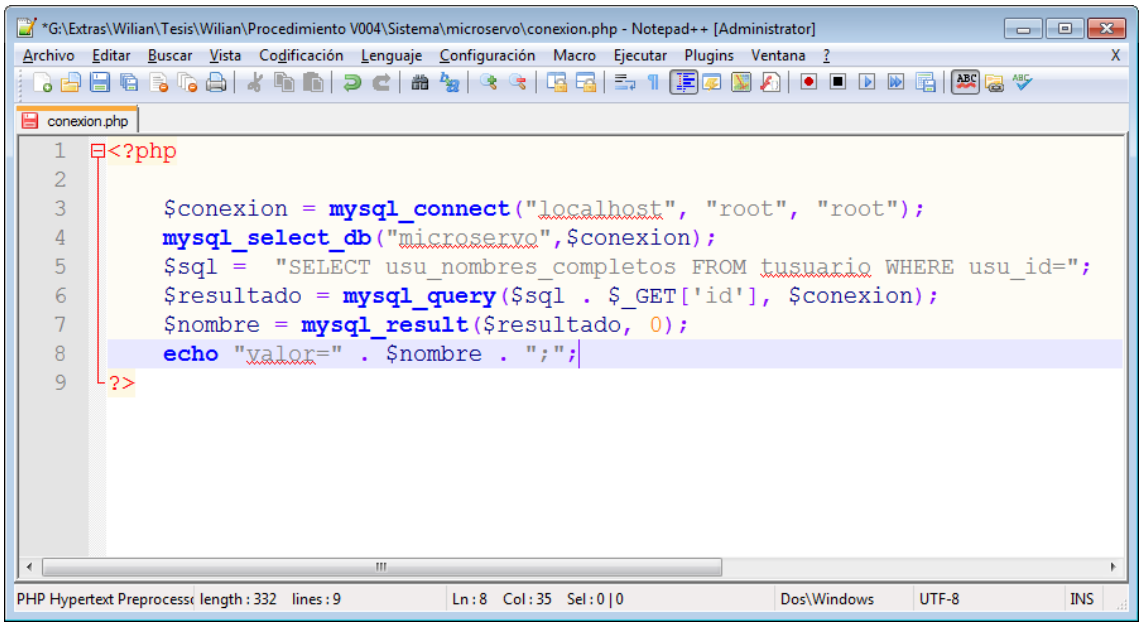


Ingreso de registro en la tabla tmovimiento



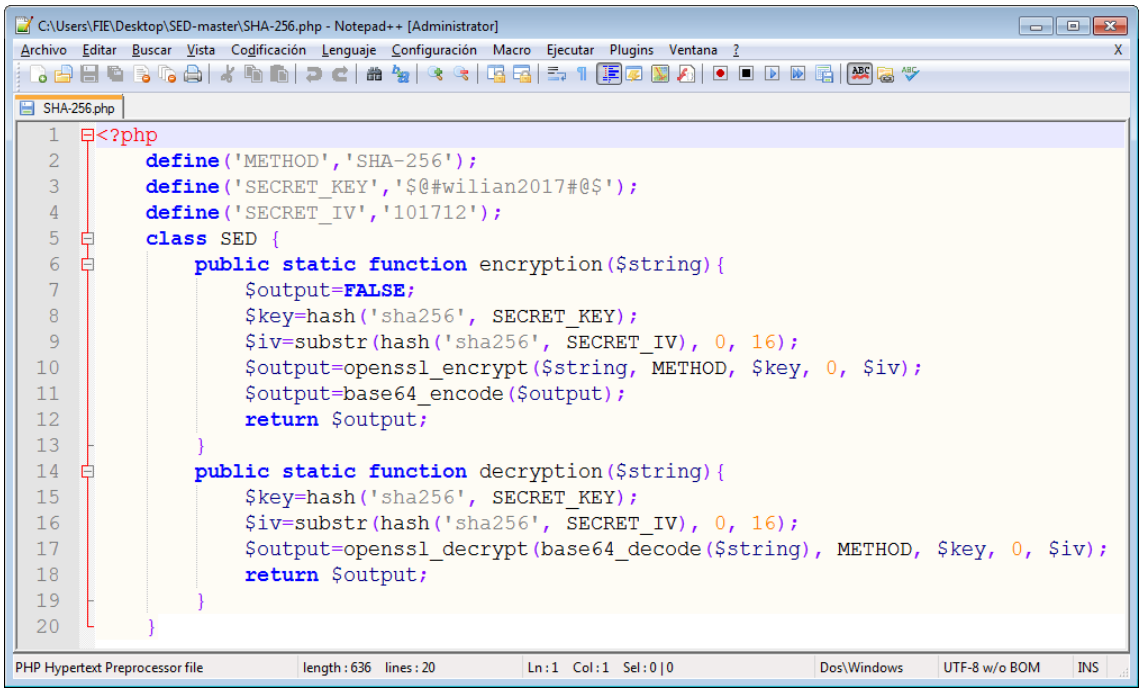
Anexo B: Conexión de mysql con php

Creamos una página web en php llamada microservo y dentro de la página web creamos un archivo llamado conexión.php.



Anexo C: Encriptar y desencriptar función sha-256 con php

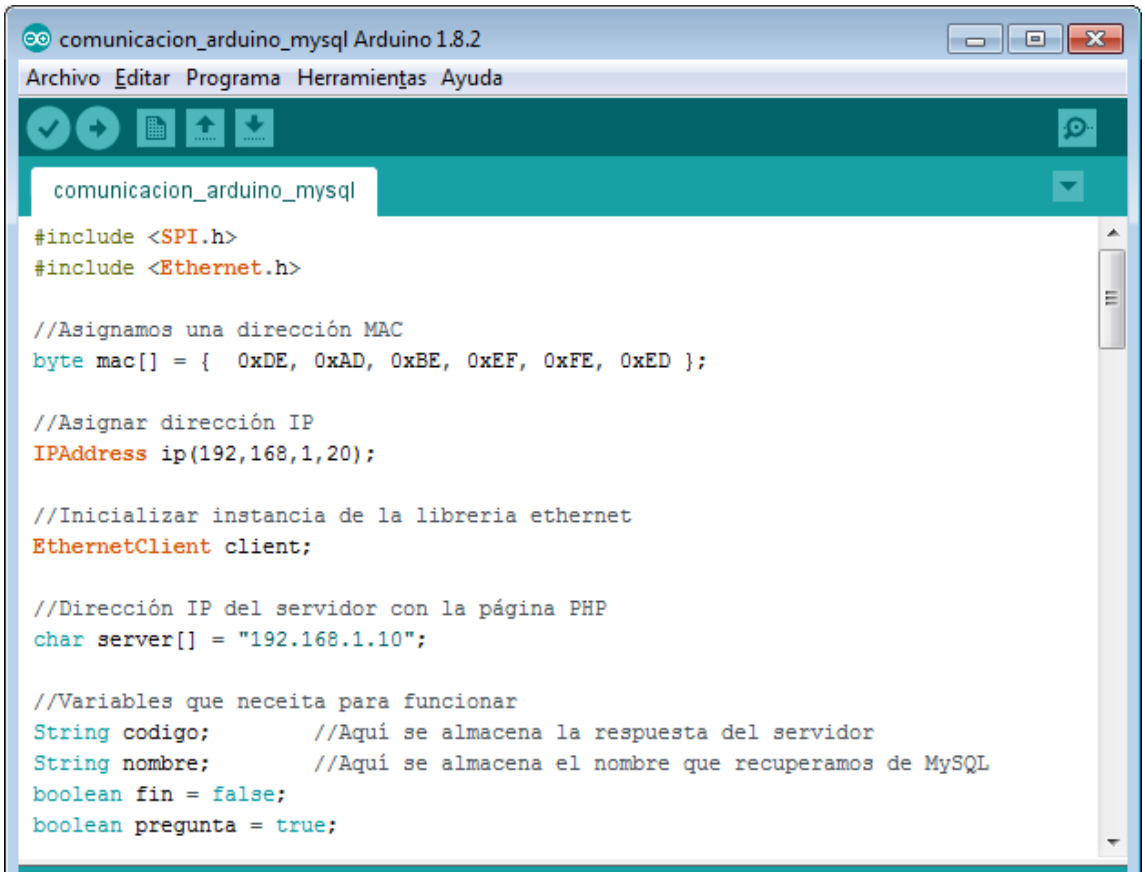
Creamos un archivo llamado SHA-256.php, para la encriptación de los datos.



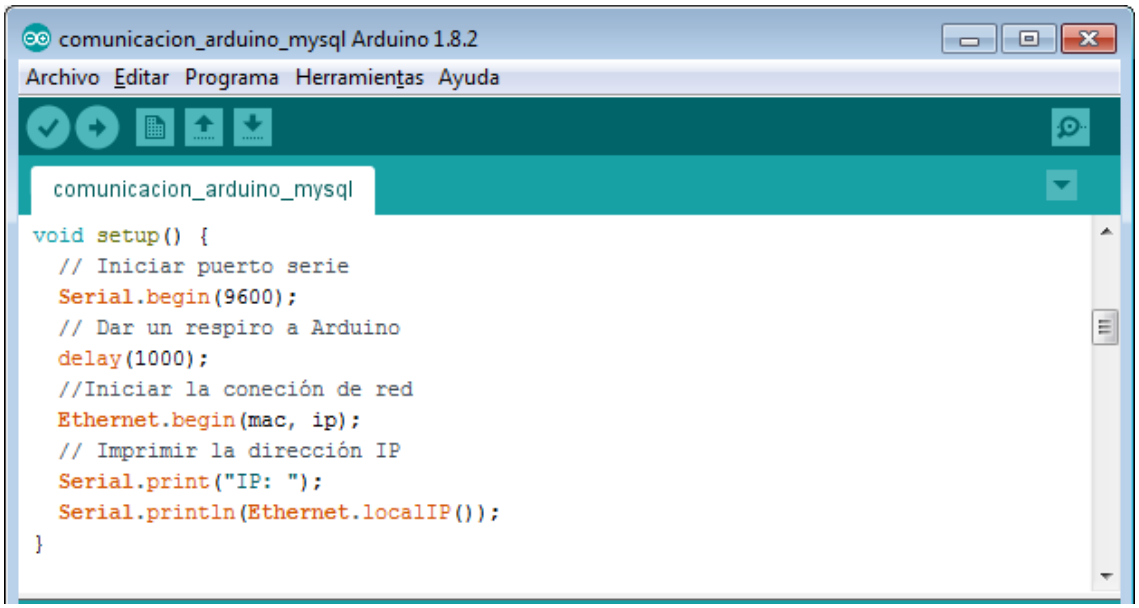
Anexo D: Conexión y lectura de registros entre mysql y arduino

Programamos en la placa del arduino para poder tener conexión entre mysql y arduino

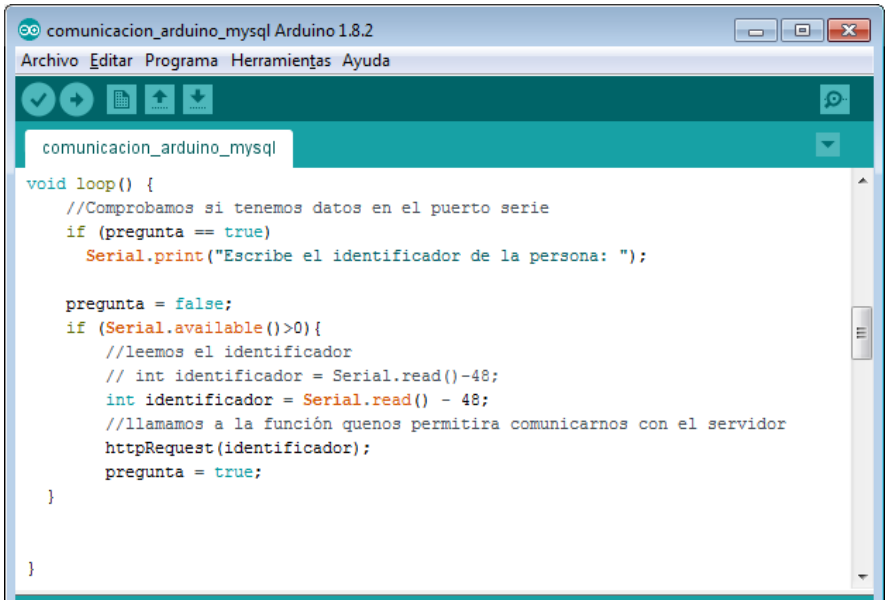
INICIALIZACIÓN DE IP LOCAL Y VARIABLES



INICIALIZACIÓN DE PUERTO SERIAL E IP LOCAL

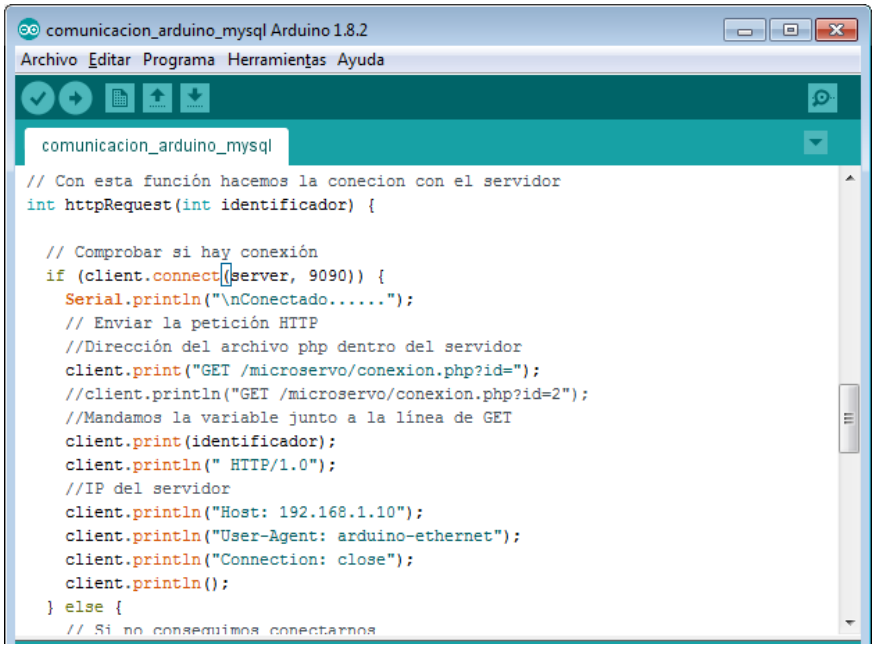


PETICIÓN DE IDENTIFICACIÓN ID MEDIANTE LA FUNCIÓN LOOP()



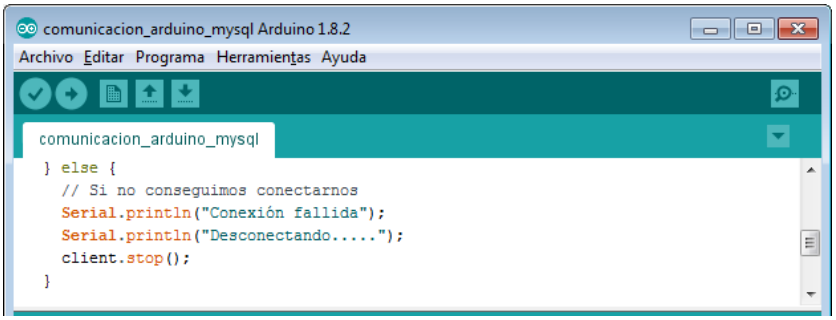

```
void loop() {
    //Comprobamos si tenemos datos en el puerto serie
    if (pregunta == true)
      Serial.print("Escribe el identificador de la persona: ");

    pregunta = false;
    if (Serial.available()>0){
        //leemos el identificador
        // int identificador = Serial.read()-48;
        int identificador = Serial.read() - 48;
        //llamamos a la función quenos permitira comunicarnos con el servidor
        httpRequest(identificador);
        pregunta = true;
  }


}
```

CREACIÓN DE UNA FUNCIÓN PARA LA CONEXIÓN CON EL SERVIDOR



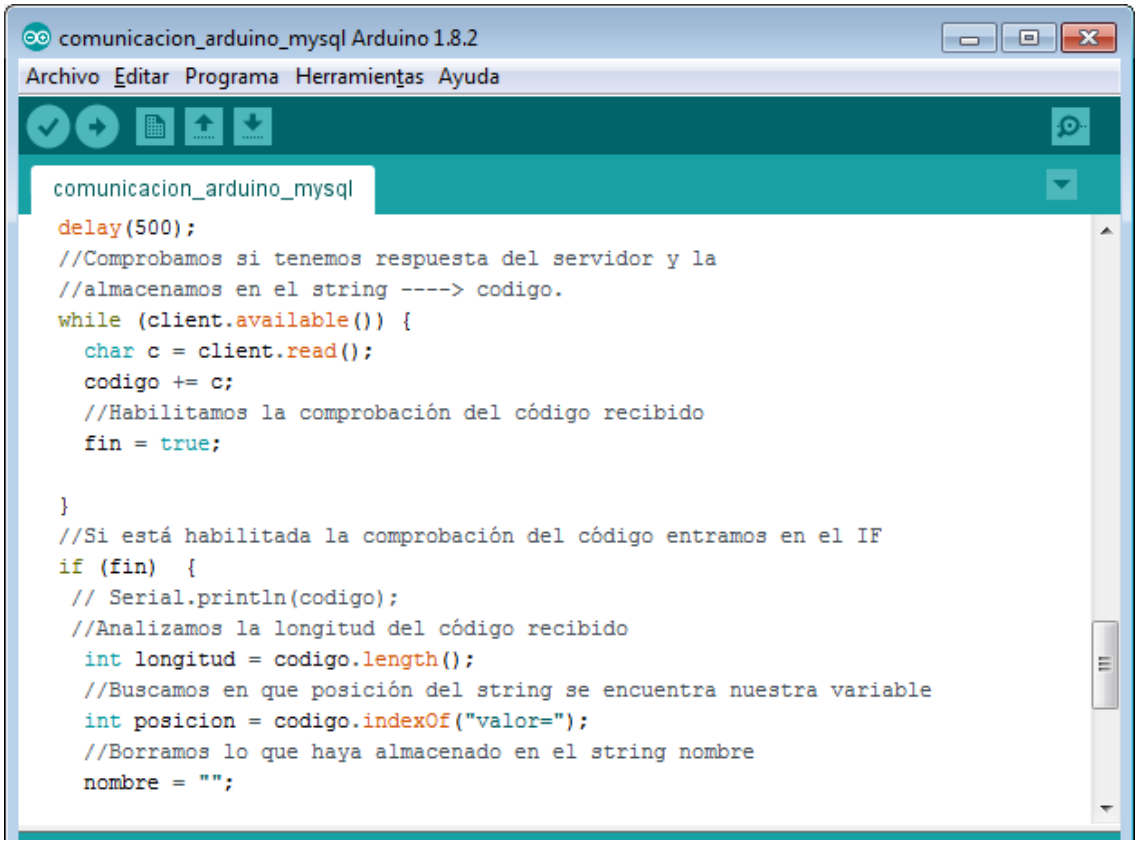

```
// Con esta función hacemos la conecion con el servidor
int httpRequest(int identificador) {

  // Comprobar si hay conexión
  if (client.connect(server, 9090)) {
    Serial.println("\nConectado.....");
    // Enviar la petición HTTP
    //Dirección del archivo php dentro del servidor
    client.print("GET /microservo/conexion.php?id=");
    //client.println("GET /microservo/conexion.php?id=2");
    //Mandamos la variable junto a la línea de GET
    client.print(identificador);
    client.println(" HTTP/1.0");
    //IP del servidor
    client.println("Host: 192.168.1.10");
    client.println("User-Agent: arduino-ethernet");
    client.println("Connection: close");
    client.println();
  } else {
    // Si no conseguimos conectarnos
```

MENSAJE ERROR DE CONEXIÓN

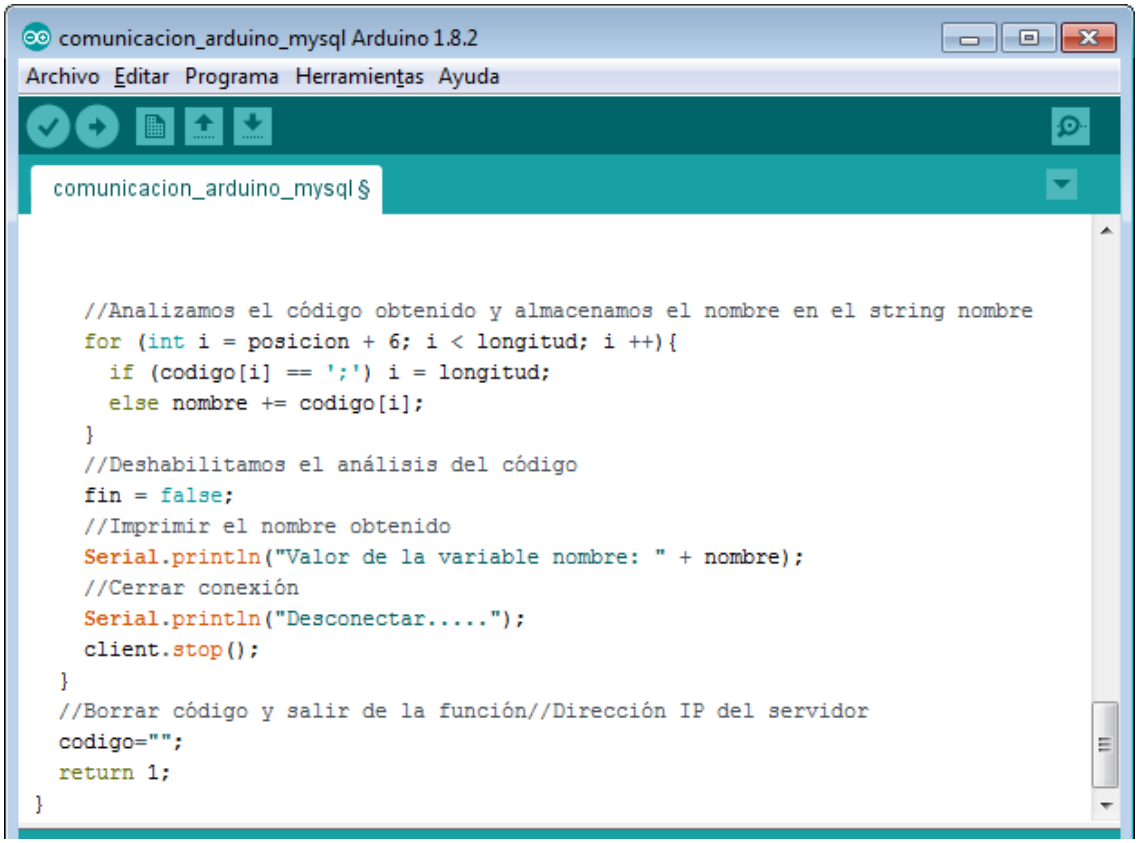



```
  } else {
    // Si no conseguimos conectarnos
    Serial.println("Conexión fallida");
    Serial.println("Desconectando.....");
    client.stop();
  }
```

COMPROBACIÓN SI EXISTE RESPUESTA CON EL SERVIDOR

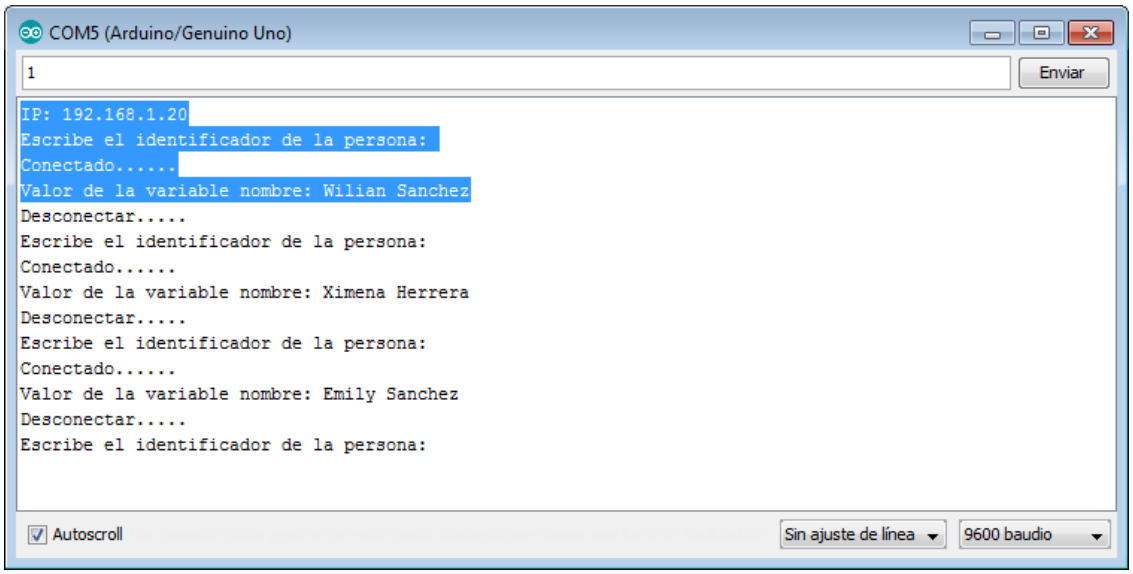


RECUPERACIÓN DE LA INFORMACIÓN



COMPROBACIÓN DE DATOS

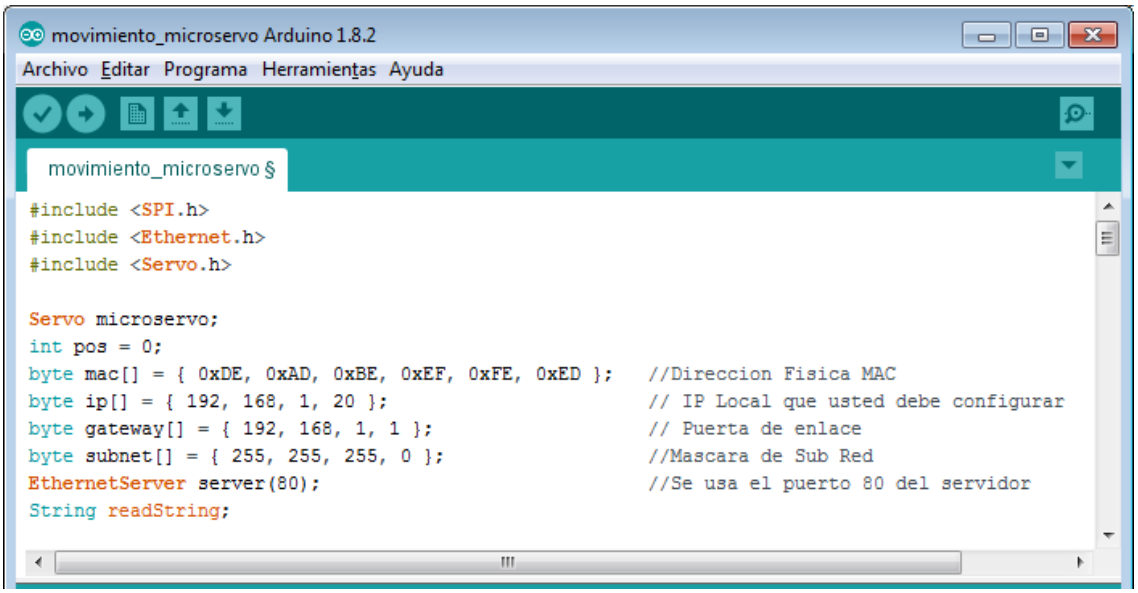
Mediante puerto serial monitor, comprobamos la comunicación y obtención de datos, como se indica a continuación:



Anexo E: Dispositivo electrónico microservo

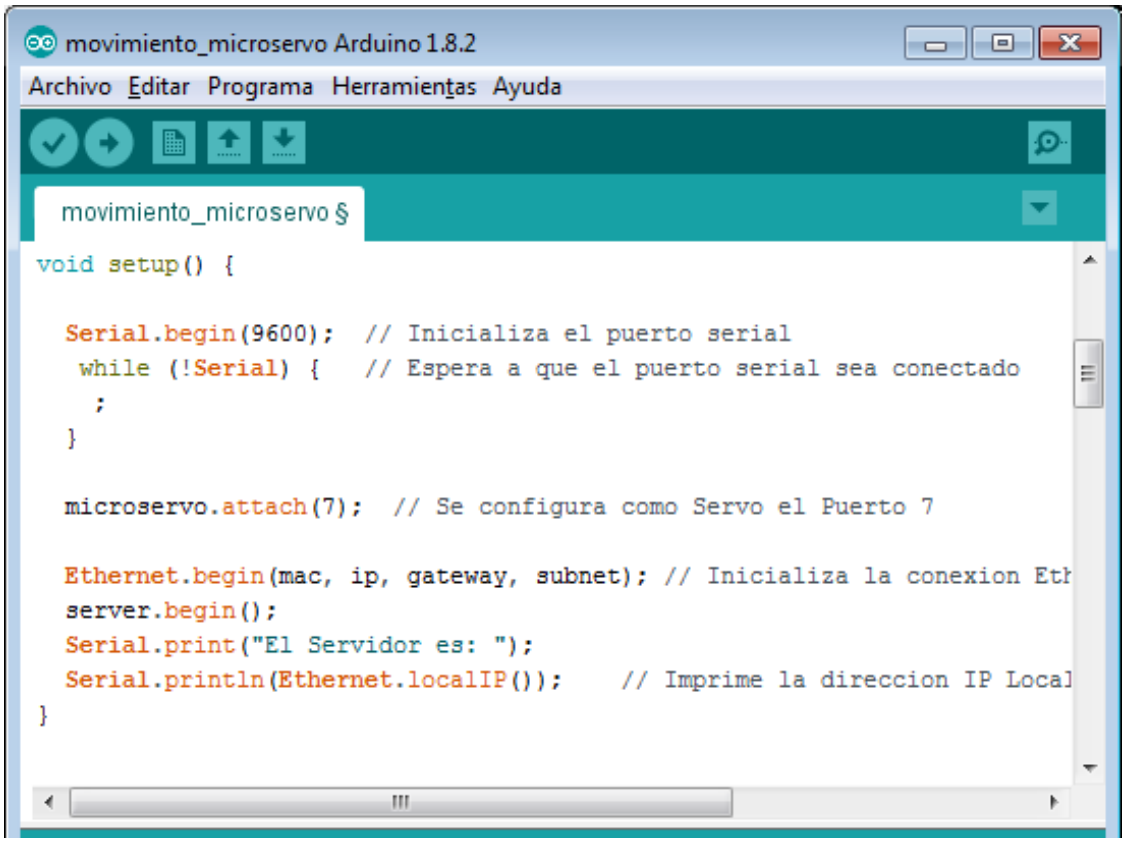
Comprobación de los movimientos izquierda y derecha en la placa del Arduino

INICIALIZACIÓN LAS LIBRERÍAS Y VARIABLES



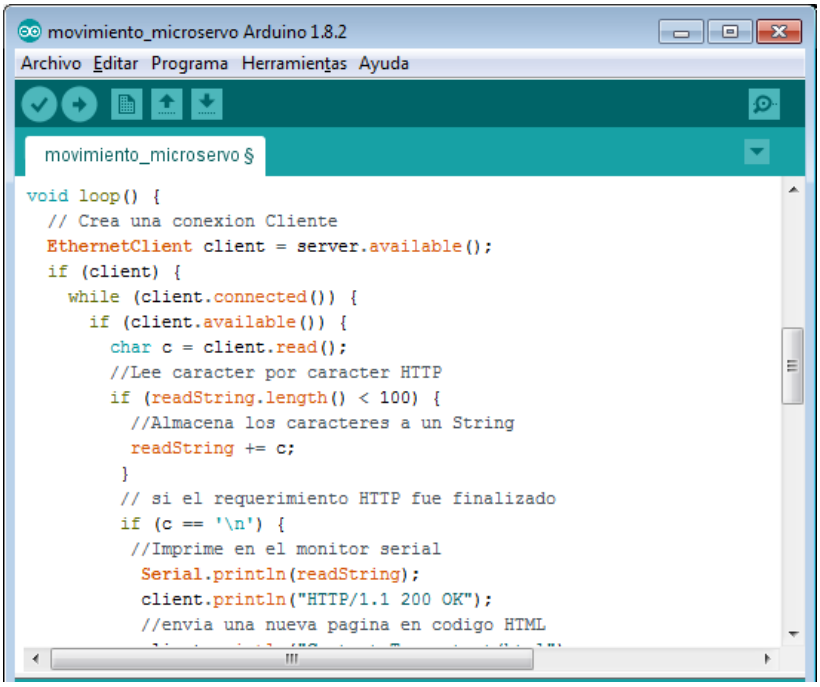
COMUNICACIÓN DE PUERTOS SERIALES

Creamos función setup(), para la comunicación de los puertos seriales



CONEXIÓN SERVIDOR Y CLIENTE

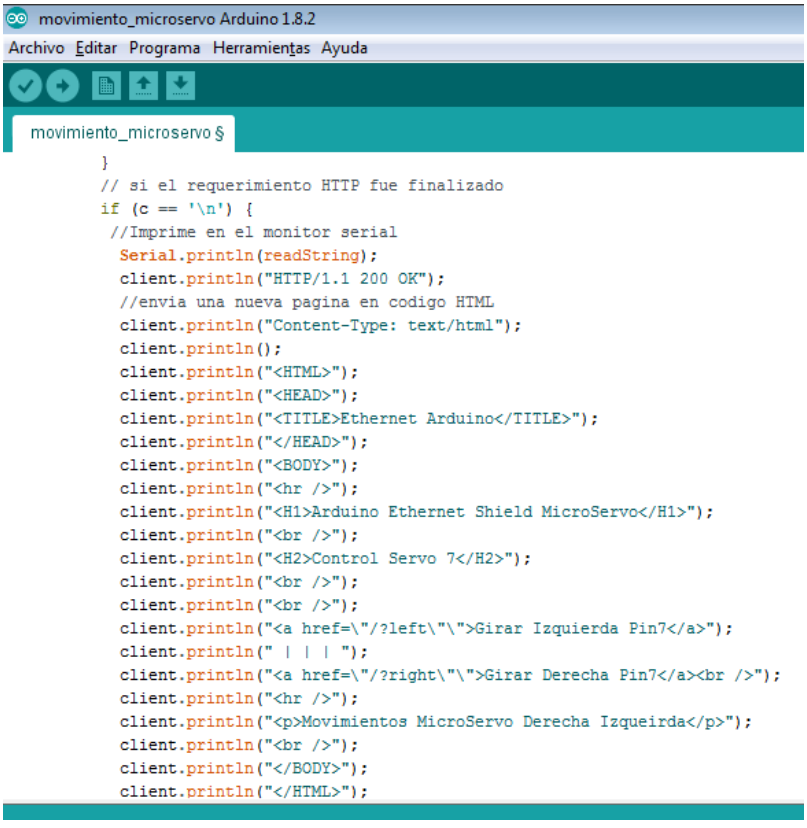
Creamos la función loop() para la conexión con el cliente




```
void loop() {
  // Crea una conexion Cliente
  EthernetClient client = server.available();
  if (client) {
    while (client.connected()) {
      if (client.available()) {
        char c = client.read();
        //Lee caracter por caracter HTTP
        if (readString.length() < 100) {
          //Almacena los caracteres a un String
          readString += c;
         }
         // si el requerimiento HTTP fue finalizado
         if (c == '\n') {
          //Imprime en el monitor serial
           Serial.println(readString);
           client.println("HTTP/1.1 200 OK");
           //envia una nueva pagina en codigo HTML
```

PÁGINA HTML

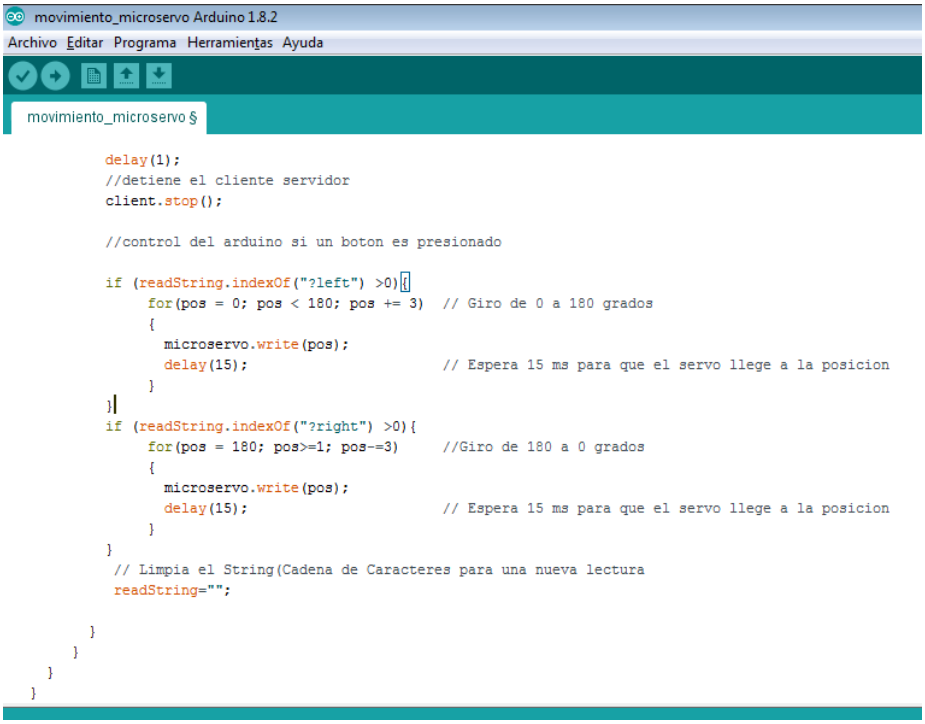
Creamos una página básica HTML, para los movimientos del microservo




```
        }
        // si el requerimiento HTTP fue finalizado
        if (c == '\n') {
         //Imprime en el monitor serial
          Serial.println(readString);
          client.println("HTTP/1.1 200 OK");
          //envia una nueva pagina en codigo HTML
          client.println("Content-Type: text/html");
          client.println();
          client.println("<HTML>");
          client.println("<HEAD>");
          client.println("<TITLE>Ethernet Arduino</TITLE>");
          client.println("</HEAD>");
          client.println("<BODY>");
          client.println("<hr />");
          client.println("<H1>Arduino Ethernet Shield MicroServo</H1>");
          client.println("<br />");
          client.println("<H2>Control Servo 7</H2>");
          client.println("<br />");
          client.println("<br />");
          client.println("<a href=\"/?left\"\">Girar Izquierda Pin7</a>");
          client.println(" | | | ");
          client.println("<a href=\"/?right\"\">Girar Derecha Pin7</a><br />");
          client.println("<hr />");
          client.println("<p>Movimientos MicroServo Derecha Izqueirda</p>");
          client.println("<br />");
          client.println("</BODY>");
          client.println("</HTML>");
```

CONTROL DE MOVIMIENTO

Creamos los controles de los giros del microservo



COMPROBACIÓN DE GIROS DEL MICROSERVO

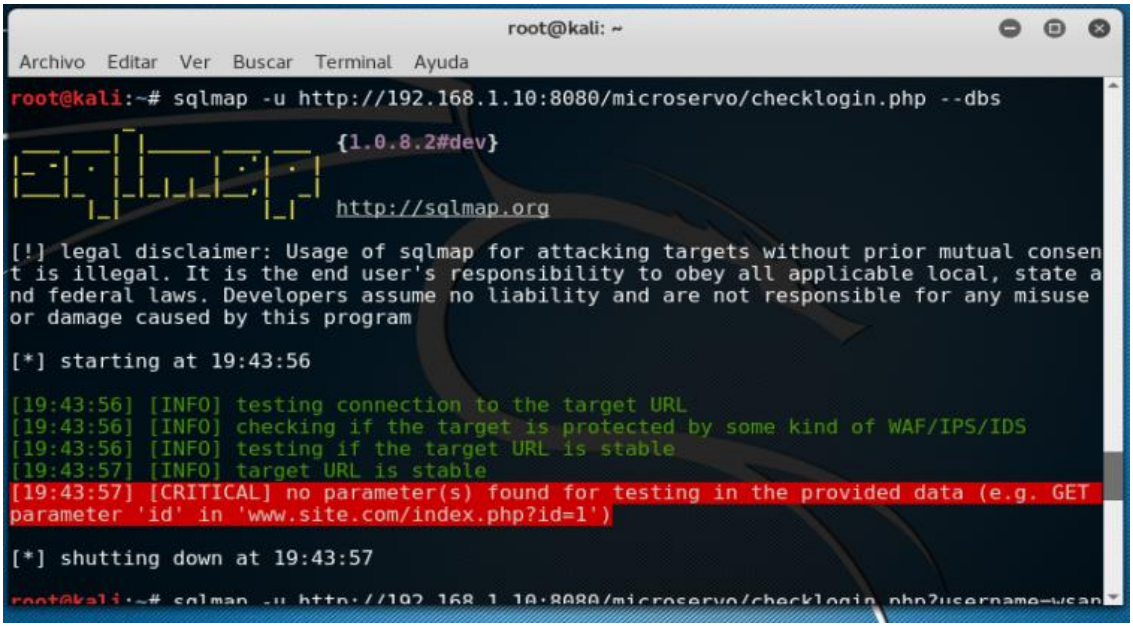
Abrimos un navegador para comprobar desde la página web los movimientos del microservo

Anexo F: Vulnerabilidad de páginas web – sqlmap – kali linux

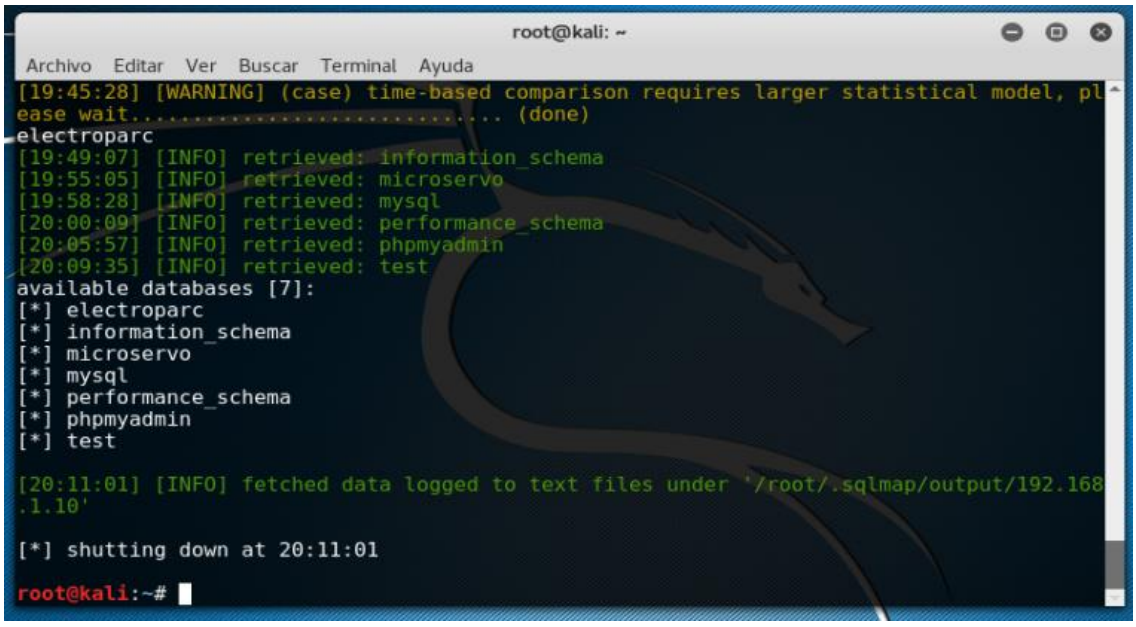
Para obtener información de una base de datos podemos realizar el ataque hacia una página web vulnerable, mediante el software Kali Linux.

1. Utilizamos el siguiente comando para verificar las bases de datos existentes

sqlmap -u [url] –dbs

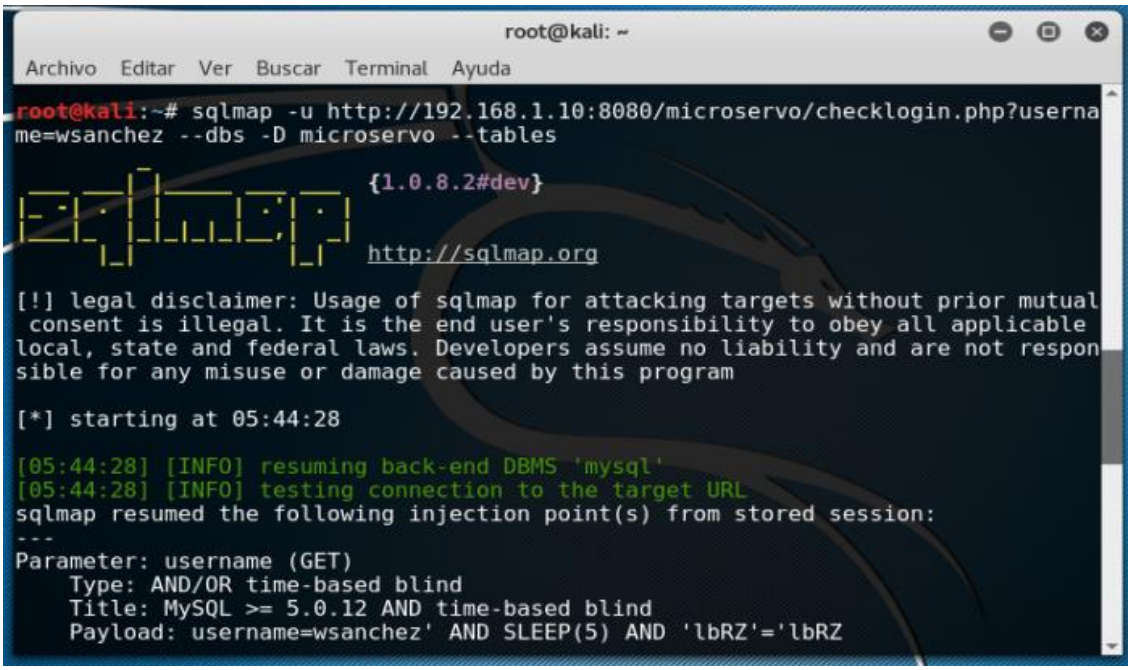


Una vez terminado el escaneo del DBMS mediante SQLMAP podemos observar el listado de las bases de datos Existentes

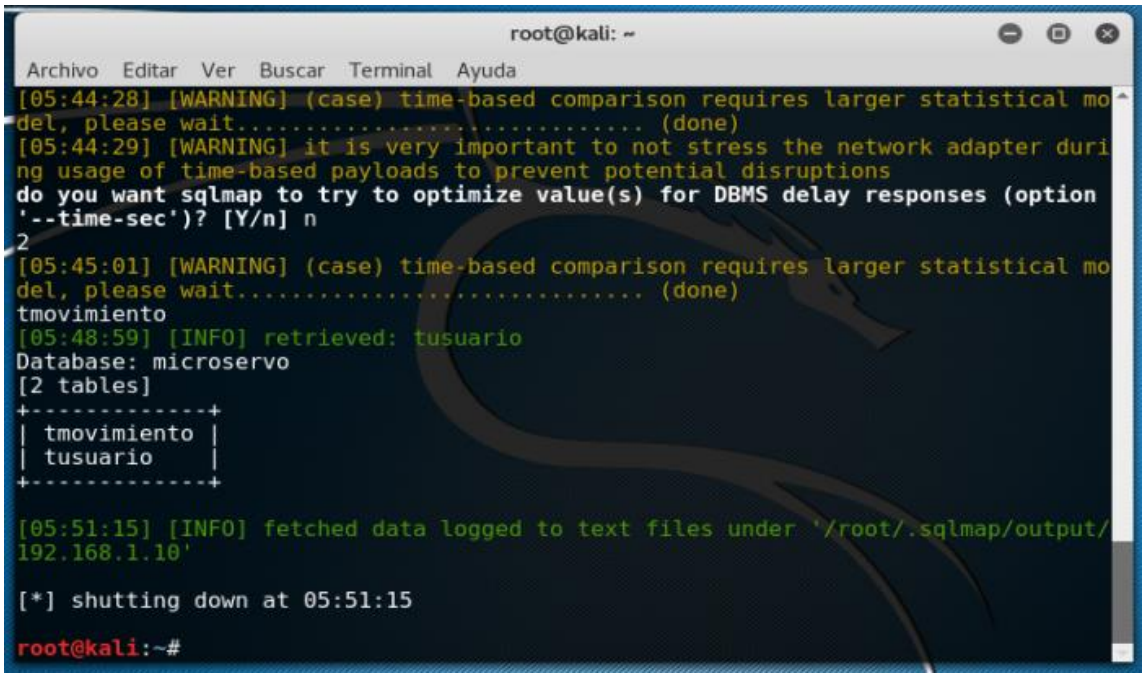


2. Una vez identificadas las bases de datos podemos acceder a la que se desee realizar algún ataque o cambio de sus registros; mediante el siguiente comando:

sqlmap -u [url] –D NombreBaseDatos –tables



Como resultado de la ejecución del comando anterior obtenemos la tabla de la base de datos seleccionada



3. Posteriormente podemos acceder a las columnas de una tabla de la dase datos a ser manipulada, mediante el siguiente comando:

sqlmap -u [url] -D NombreBaseDatos –T NombreTabla --columns

Tabla tusuario

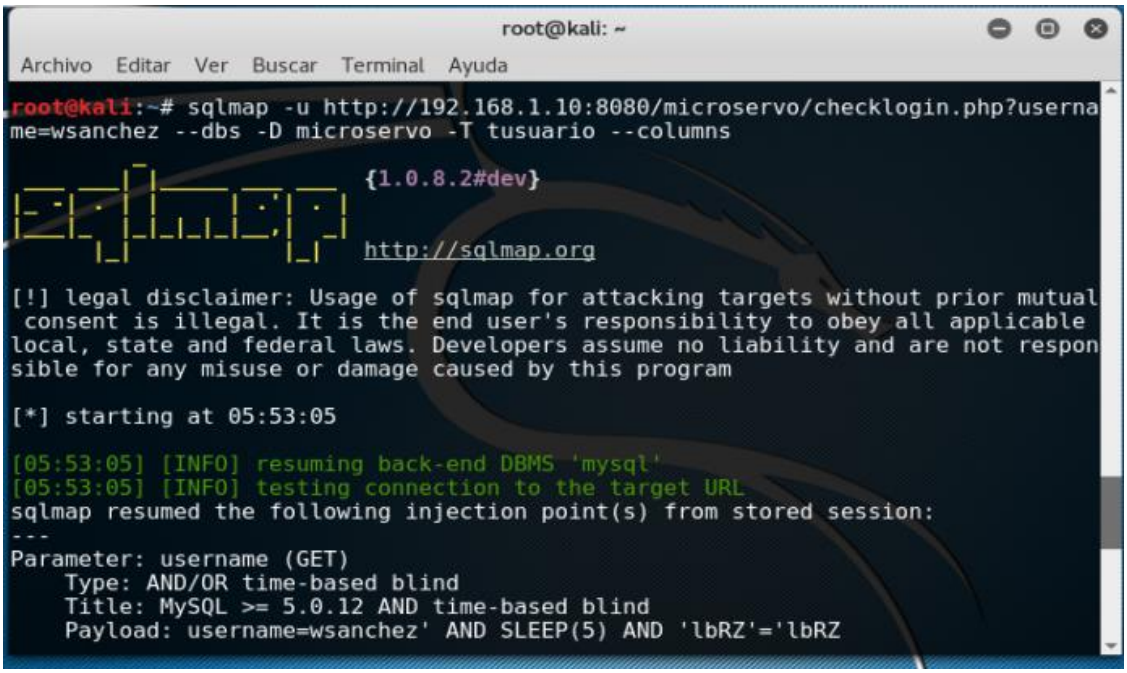
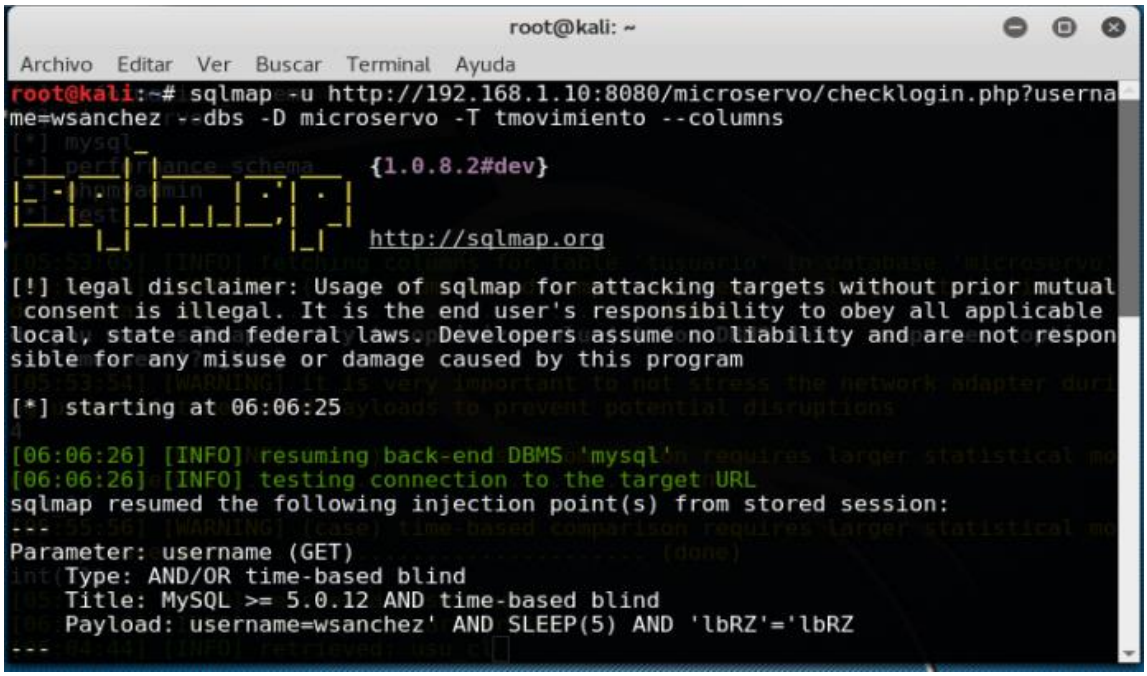
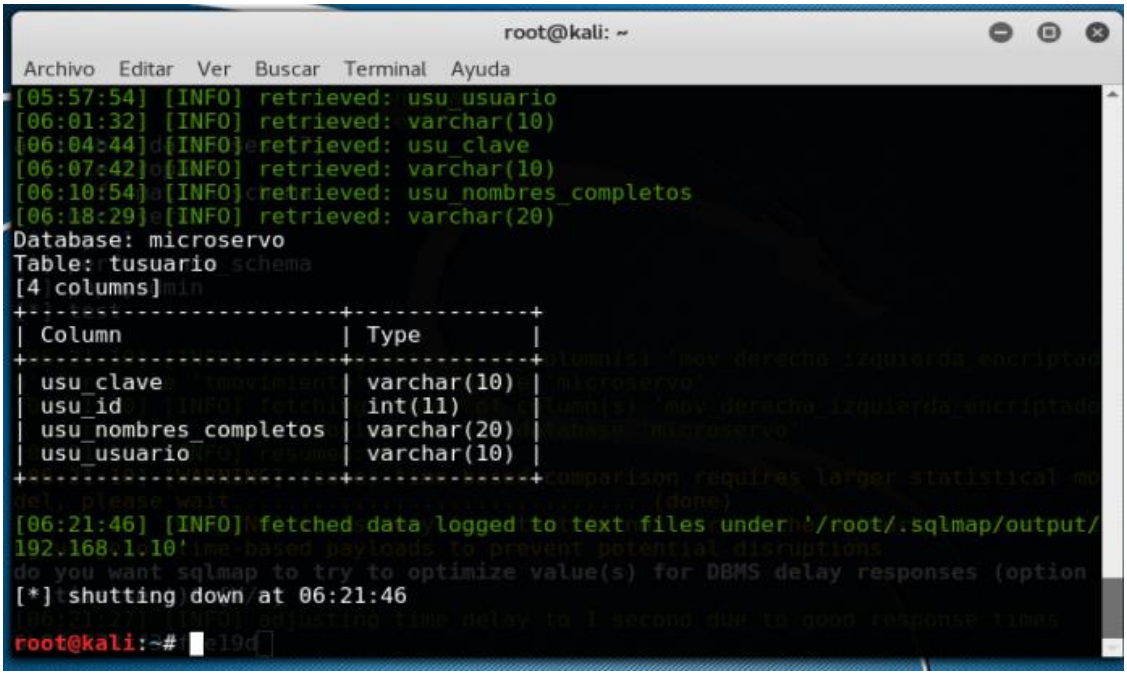


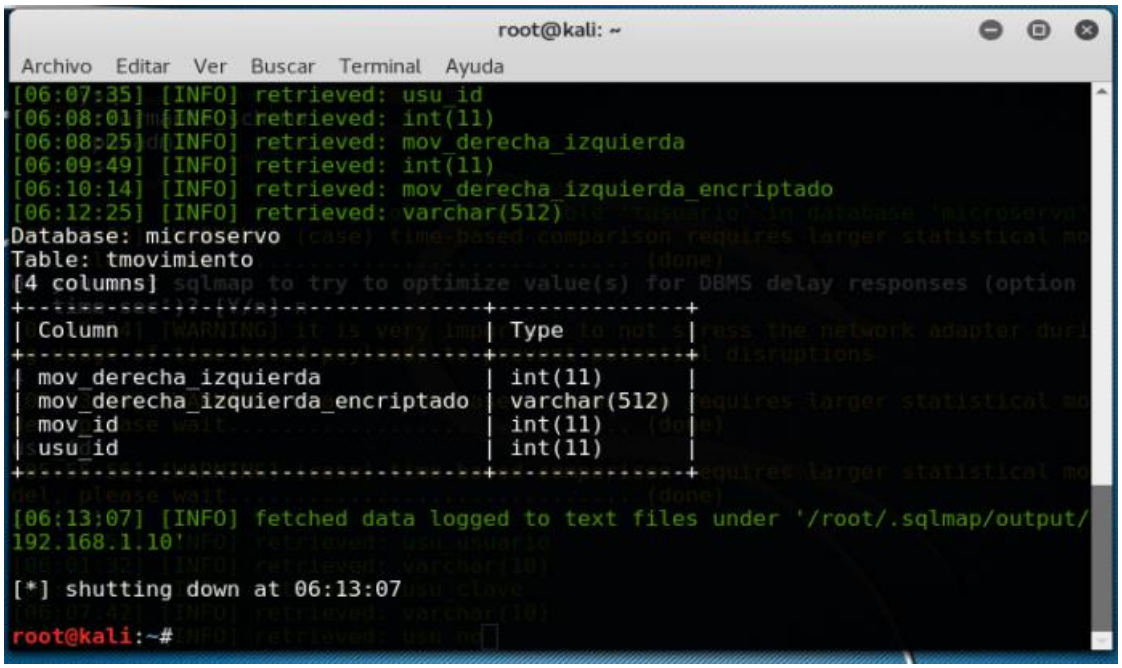
Tabla tmovimiento



La ejecución del comando anterior nos da como resultado las columnas existentes dentro de una tusuario, como se observa a continuación:

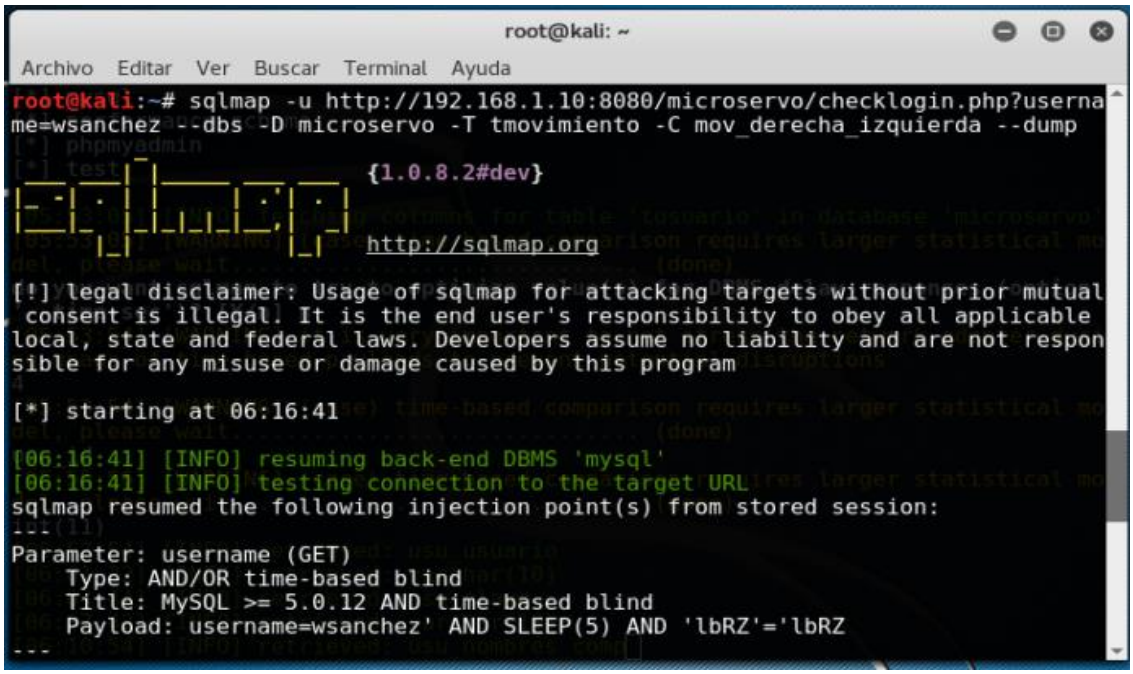


La ejecución del comando anterior nos da como resultado las columnas existentes dentro de la tabla tmovimiento, como se observa a continuación:



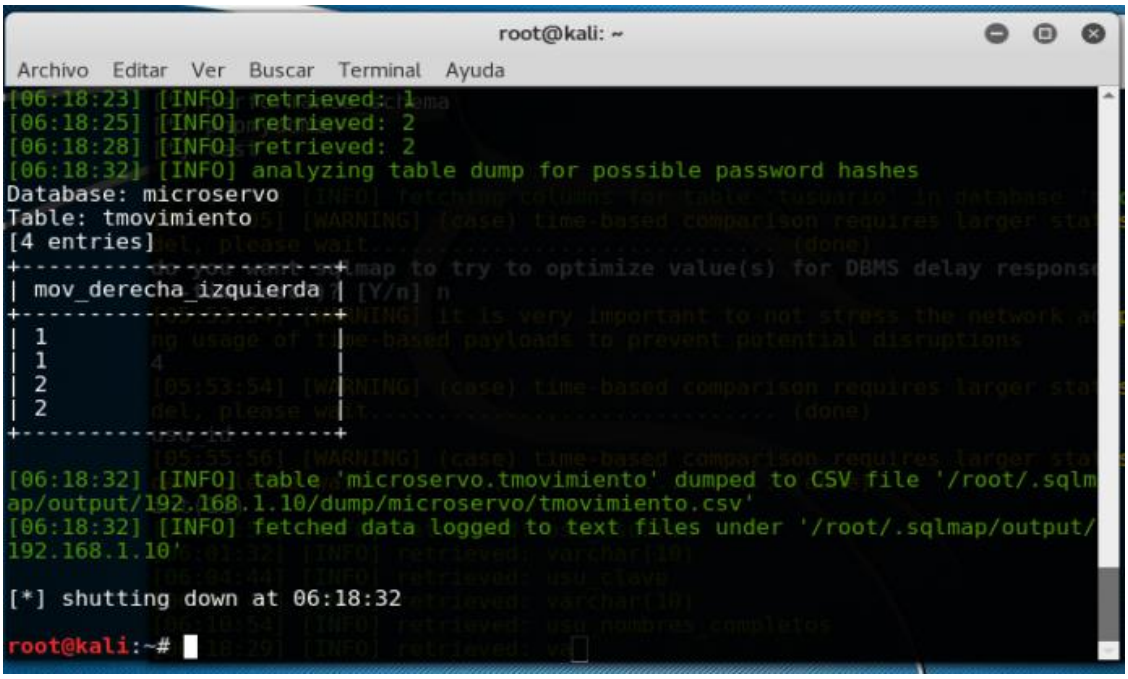
4. Por ultimo podemos observar la información de la columna que posee la base datos dentro de una tabla, mediante el siguiente comando:

sqlmap -u [url] -D NombreBaseDatos –T NombreTabla –C NombreComlumna –dump

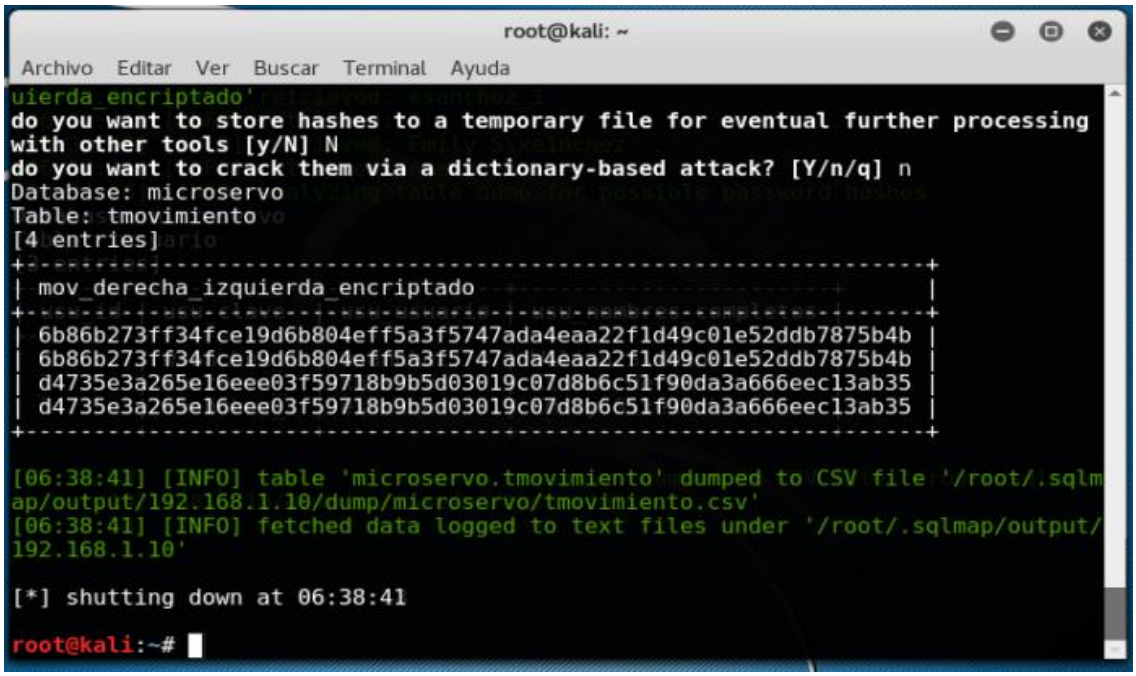


Una vez termina la ejecución del comando anterior podemos observar la información de la columna seleccionada, como se indica a continuación:

Columna mov_derecha_izquierda

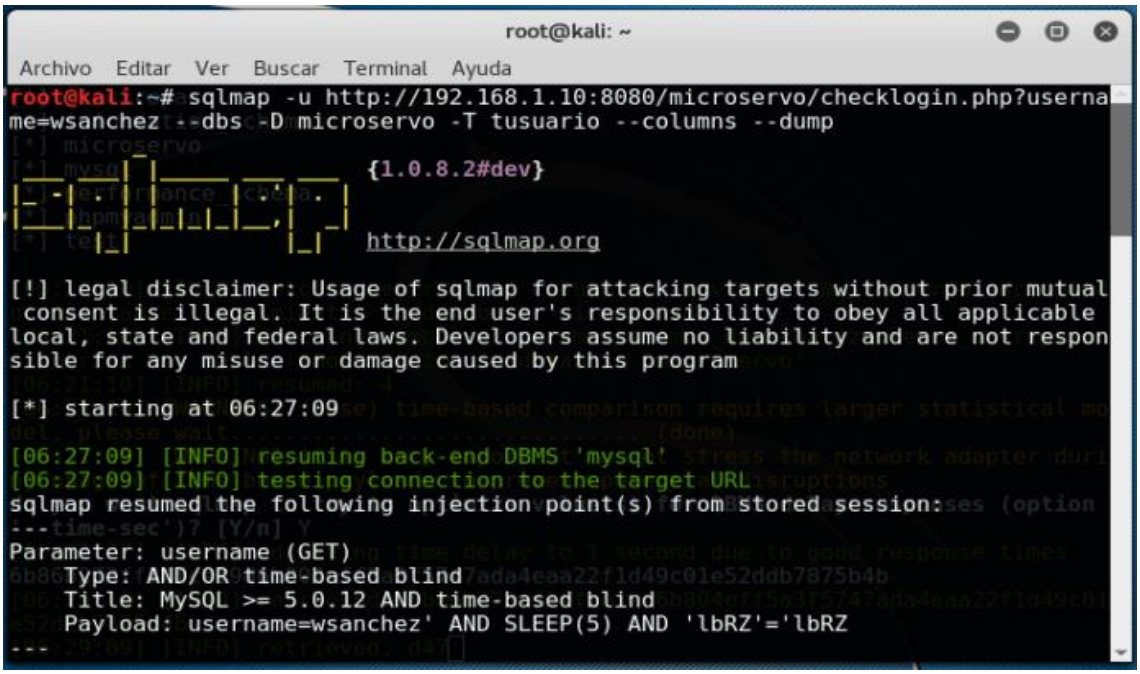


Columna mov_derecha_izquierda_encriptado

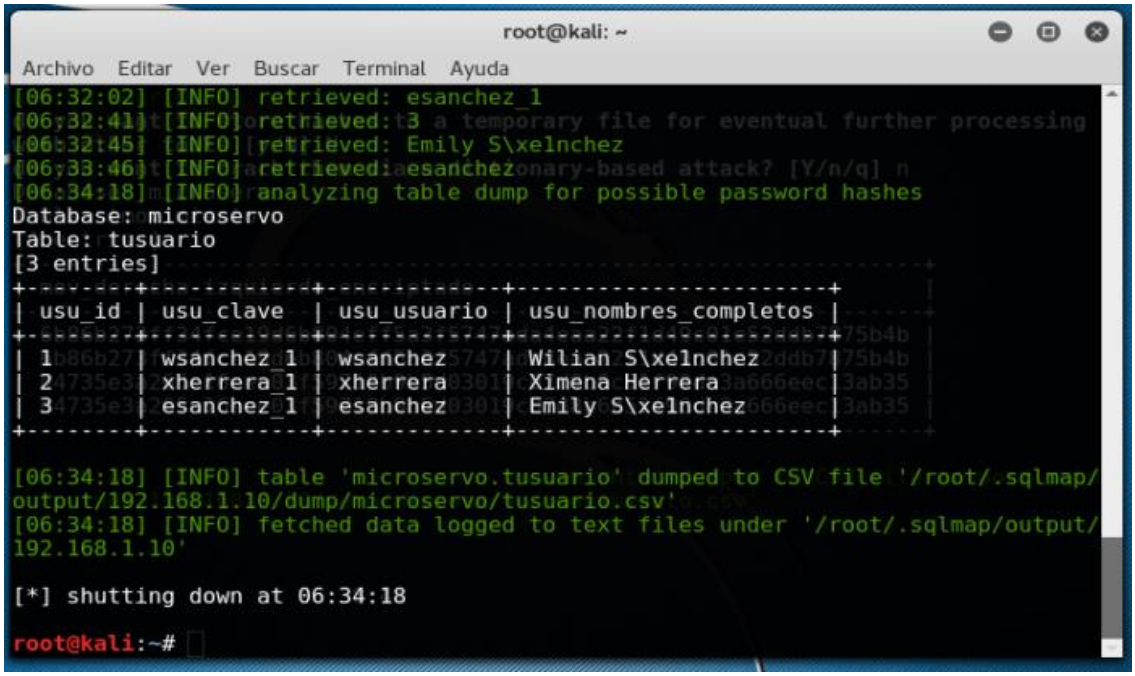


Para poder leer toda la información de una tabla específica podemos realizar con el siguiente comando:

sqlmap -u [url] -D NombreBaseDatos –T NombreTabla –columkns –dump



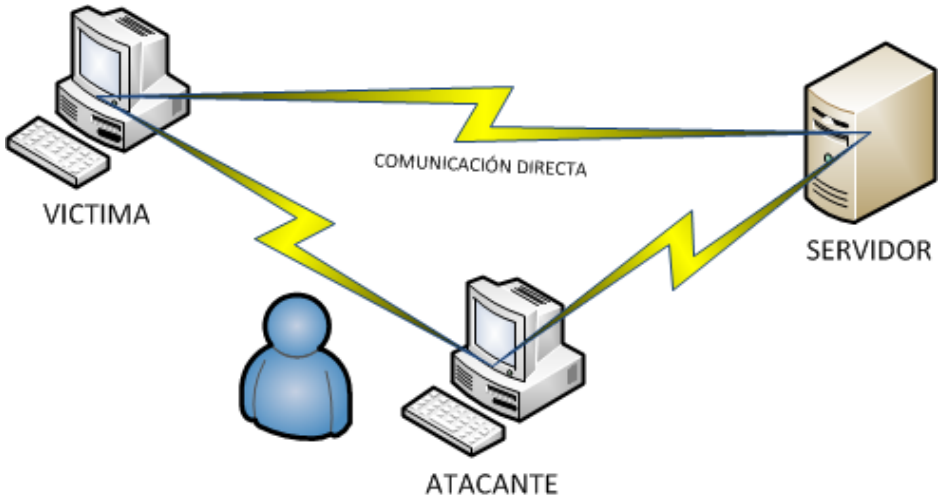
Una vez ejecutado el comando anterior obtenemos la siguiente información:



Anexo G: Ataque por man-in-the-middle

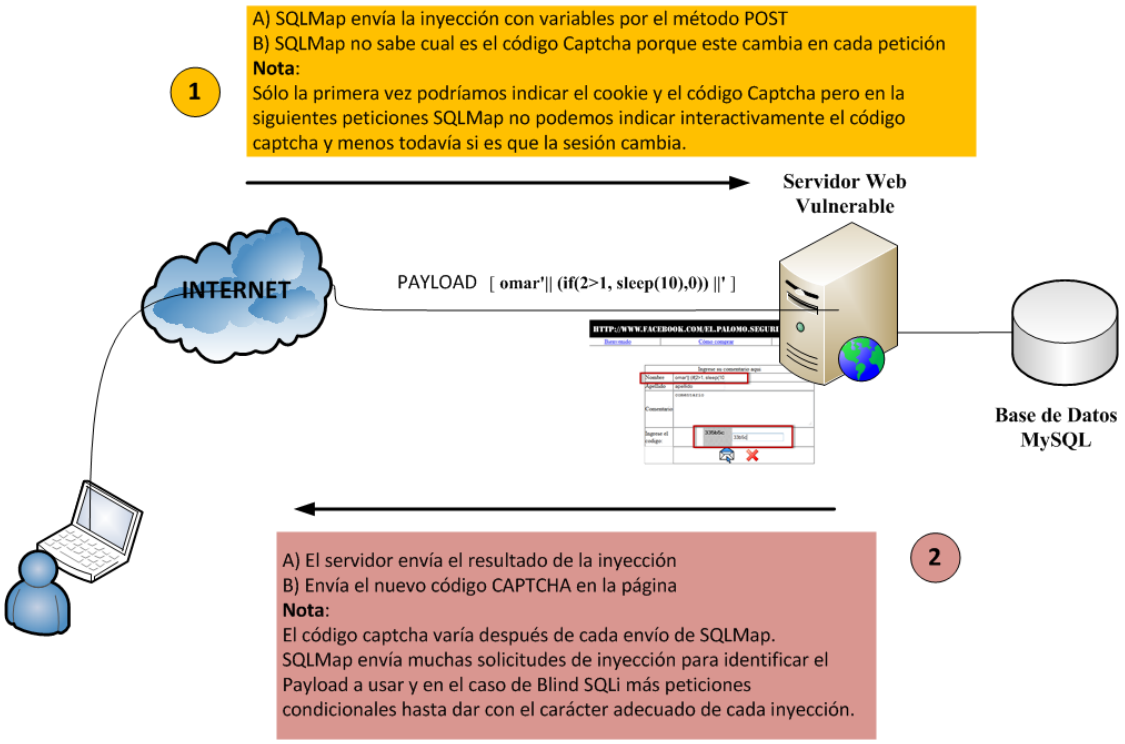
Mediante el ataque Man In The Middle (hombre en el medio), consiste en introducirse en la comunicación entre dos equipos para que todo el tráfico pase por nosotros y poder así desencriptar sus datos, contraseñas y toda la información que pueda obtener.

Existen varios tipos de defensa contra estos ataques MITM, estas defensas emplean técnicas de autenticación basadas en: infraestructura de claves públicas, autenticación mutua fuertes (claves secretas, contraseñas, examen de latencia (cálculos de la función hash criptográfica que conducen a decenas de segundos, si ambas partes toman normalmente 20 segundos y el cálculo de 60 segundos para llegar a cada parte, esto puede indicar a un tercero en la comunicación).



Anexo H: Ataque por inyección

Los ataques de inyección, más específicamente sqli (Structured Query Language Injection) es una técnica para modificar una cadena de consulta de base de datos mediante la inyección de código en la consulta. El SQLI explota una posible vulnerabilidad donde las consultas se pueden ejecutar con los datos validados.

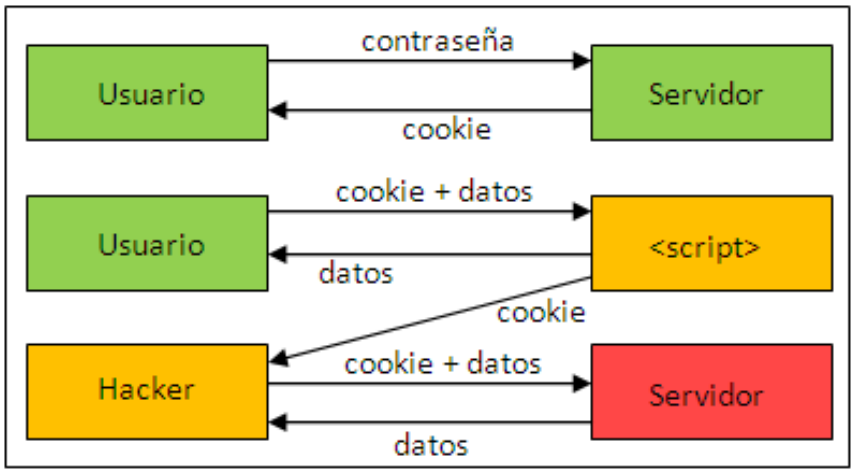


SQLI siguen siendo una de las técnicas de sitios web más usadas y se pueden utilizar para obtener acceso a las tablas de bases de datos, incluyendo información del usuario y la contraseña. Este tipo de ataques son particularmente comunes en los sitios de empresas y de comercio electrónico donde los hackers esperan grandes bases de datos para luego extraer la información sensible. Los ataques sqli también se encuentran entre los ataques más fáciles de ejecutar, que no requiere más que un solo PC y una pequeña cantidad de conocimientos de base de datos.

Anexo I: Ataque por cross-site scripting (XSS)

XSS es un ataque de inyección de código malicioso para su posterior ejecución que puede realizarse a sitios web, aplicaciones locales e incluso al propio navegador.

Se ejecuta cuando un usuario mal intencionado envía código malicioso a la aplicación web y se coloca en forma de un hipervínculo para conducir al usuario a otro sitio web, mensajería instantánea o un correo electrónico. Así mismo, puede provocar una negación de servicio (DDos).



Generalmente, si el código malicioso se encuentra en forma de hipervínculo es codificado en HEX (basado en el sistema de numeración hexadecimal, base 16) o algún otro, así cuando el usuario lo vea, no le parecerá sospechoso. De esta manera, los datos ingresados por el usuario son enviados a otro sitio, cuya pantalla es muy similar al sitio web original. De esta manera, es posible secuestrar una sesión, robar cookies y cambiar la configuración de una cuenta de usuario.