**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR**




**DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERA EN SISTEMAS Y COMPUTACIÓN**

**Tema:**

# PROPUESTA DE UNA GUÍA DE SEGURIDAD ENFOCADA AL CÓDIGO SEGURO, BASADO EN LAS CONTRAMEDIDAS Y METODOLOGÍAS DE SEGURIDAD A NIVEL DE LA CAPA DE APLICACIÓN

**Nombre:** María Valeria Rosero Muñoz

**DIRECTOR:** MSc. Damián Nicolalde

QUITO – ECUADOR

2019 febrero

## AGRADECIMIENTO

En primer lugar me gustaría agradecer a Dios, por brindarme fortaleza y su guía en el transcurso de mi carrera, a mi ángel por cuidarme y protegerme en todo momento.

A mi familia, por brindarme su apoyo incondicional, gracias a mi madre por estar siempre a mi lado, siendo la principal promotora de mis sueños, enseñándome a creer en mi misma y a luchar para conseguir mis metas.

A mi hermana por ser mi soporte y mi confidente, gracias por ayudarme a cumplir uno de mis de mis metas y en especial por brindarme tu amor incondicional.

A mi hermano por estar a mi lado y cuidarme, siendo uno de mis principales empujes en mi vida.

De manera especial a mi director MSc. Damián Nicolalde por su paciencia y su apoyo, para culminar el presente trabajo.

***Valeria Rosero***

## DEDICATORIA

Este presente trabajo está dedicado a mi familia, por su apoyo, sacrificio y guía en el transcurso de mi vida.

A mi madre Germania Muñoz, quien es mi modelo a seguir, gracias por ser mi fortaleza , por darme tu amor incondicional y por ser esa mujer luchadora que nunca se rinde. La persona que soy hoy en día, es gracias a tus enseñanzas y valores.

A mi hermana María Teresa Rosero, por brindarme la oportunidad de cumplir una de mis metas, por estar a mi lado siendo mi ángel y mi guía en todo momento. A mi hermano Carlos Rosero, por su apoyo y paciencia, por estar a mi lado dándome ánimos para seguir adelante.

***Valeria Rosero***

## RESUMEN

La tecnología ha afectado de gran manera a la sociedad en la que vivimos, esta ha ido formando parte de nuestra vida con el pasar de los años. En la actualidad toda máquina está controlada por un software, como: relojes, dispositivos móviles, electrodomésticos, automóviles, aviones , etc. (Miríada, 2016)

Tanto ha sido su crecimiento que ha influenciado en la mayoría de las facetas de la vida cotidiana de las personas, organizaciones y gobiernos. La tecnología ha hecho más cómoda y fácil nuestra vida, lo que ha causado que el ser humano se vuelva dependiente de ella, esto no solo ha ayudado al crecimiento de la tecnología sino también ha provocado el aumento de delitos informáticos.

Actualmente los ataques informáticos se han convertido en una de las mayores amenazas a nivel mundial, siendo difícil mantenerle el paso. El cybercrimen es considerado uno de los principales delitos según el Congreso de las Naciones Unidas sobre Prevención del Delito y Justicia Penal (Congreso de las Naciones Unidas sobre Prevención del Delito & y Justicia Penal, 2010). Un ejemplo de este tipo de delitos es: los ataques realizados a software, debido a la explotación de las vulnerabilidades que puede presentar el sistema. En esto influye considerablemente el factor humano, ya que este sistema es desarrollado por personas, y como toda persona cometemos errores.

Por esta razón se realizará una comparación de las diferentes guías y metodologías de desarrollo seguro, y según esto poder diseñar un prototipo de guía a nivel de la capa de aplicación, que permita que los programadores desarrollen un sistema con código seguro que pueda hacer a nuestro sistema menos vulnerable.

# ÍNDICE GENERAL

AGRADECIMIENTO ..... II

DEDICATORIA ..... III

RESUMEN ..... IV

ÍNDICE GENERAL ..... V

ÍNDICE GENERAL DE TABLAS ..... IX

ÍNDICE GENERAL DE ILUSTRACIONES ..... X

INTRODUCCIÓN ..... XII

OBJETIVOS ..... XIV

**Objetivos Generales ..... XIV**

**Objetivos Específicos ..... XIV**

CAPÍTULO 1: MARCO TEÓRICO ..... 1

**1.1. Ciberseguridad ..... 1**
**1.1.1. Ciberseguridad y Protección en la Red ..... 3**
**1.1.1.1. Hogares ..... 3**
**1.1.1.2. Personas ..... 4**
**1.1.1.3. Empresas ..... 7**

**1.2. Capas del Modelo OSI ..... 14**
**1.2.1. Capa física ..... 15**
**1.2.2. Enlace de datos ..... 16**
**1.2.3. Red ..... 16**
**1.2.4. Transporte ..... 16**
**1.2.5. Sesión ..... 16**
**1.2.6. Presentación ..... 16**
**1.2.7. Capa de Aplicación ..... 17**

**1.3. Capa de Aplicación ..... 18**
**1.3.1. Protocolos de Alto Nivel ..... 18**
**1.3.1.1. SNMP ..... 19**
**1.3.1.2. HTTP ..... 19**
**1.3.1.3. FTP ..... 19**
**1.3.1.4. TELNET ..... 20**

**1.3.1.5. DNS** ..... 20
**1.3.1.6. SMTP** ..... 21
**1.3.1.7. OTROS** ..... 21

**1.4. Servicios WEB ..... 21**
**1.4.1. Análisis y evaluación de los servicios Web** ..... 22
**1.4.1.1. Vulnerabilidades y riegos en los sistemas** ..... 22
**1.4.1.2. Principales ataques y amenazas a servicios web** ..... 27
**1.4.1.3. Acciones de seguridad o Contramedidas** ..... 34

CAPÍTULO 2: METODOLOGÍAS DE DESARROLLO SEGURO ..... 36

**2.1. Estado del Arte ..... 36**

**2.2. Metodologías de Desarrollo Web ..... 38**

**2.3. Metodologías de Desarrollo con Código Seguro ..... 39**

2.3.1. PROPIEDADES BÁSICAS DE SOFTWARE SEGURO ..... 39
**a) Confidencialidad** ..... 40
**b) Integridad** ..... 40
**c) Disponibilidad** ..... 40
**d) Rendición de cuentas** ..... 40
**e) No repudio** ..... 40

2.3.2. PROPIEDADES CONDUCENTES ..... 41
**a) Confianza** ..... 41
**b) Exactitud** ..... 41
**c) Predecible** ..... 42
**d) Confiabilidad** ..... 42
**e) Seguridad** ..... 42

2.3.3. METODOLOGÍAS ..... 43
**2.3.3.1. OWASP** ..... 43
**2.3.3.1.1. SAMM** ..... 46
**2.3.3.1.1.1. Definición** ..... 46
**2.3.3.1.1.2. Modelo** ..... 47
**a) Funciones de Negocio** ..... 49
**2.3.3.2. Microsoft** ..... 50
**2.3.3.2.1. SDL** ..... 50
**2.3.3.2.1.1. Definición** ..... 50
**2.3.3.2.1.2. Modelo** ..... 51

CAPITULO 3: CUADRO COMPARATIVO Y ANÁLISIS DE LA BRECHA QUE EXISTE ENTRE LAS CARACTERÍSTICAS DE LAS METODOLOGÍAS DE DESARROLLO SEGURO. ..... 57

3.1. DESCRIPCIÓN ..... 57
**3.1.1. Criterios Principales** ..... 57
**3.1.1.1. Análisis** ..... 58
**3.1.2. Criterios Secundarios** ..... 59
**3.1.2.1. Análisis** ..... 61
**3.1.3. Criterios de Seguridad** ..... 62
**3.1.3.1. Análisis** ..... 64

CAPITULO 4: PROPUESTA ..... 66

**4.1. PROPUESTA DE UN PROTOTIPO DE GUÍA DE SEGURIDAD ENFOCADO AL DESARROLLO SEGURO, BASADO EN LAS CONTRAMEDIDAS Y METODOLOGÍAS DE SEGURIDAD A NIVEL DE LA CAPA DE APLICACIÓN ..... 66**

**4.2. Caso de Estudio ..... 68**
**4.2.1. Niveles de Madurez** ..... 70
**4.2.2. Pruebas** ..... 70
**4.2.3. Fases y Paquetes de Trabajo** ..... 71

CAPITULO 5: CONCLUSIONES Y RECOMENDACIONES ..... 83

**5.1. Conclusiones ..... 83**

**5.2. Recomendaciones ..... 84**

ANEXOS ..... 85

**Anexo 1 - Tarjeta de Calificación SAMM ..... 85**

**Anexo 2 - Análisis de Riesgos ..... 93**

**Anexo 3 - Plan de Capacitación ..... 99**

**Anexo 4: Modelo de Amenazas ..... 103**

**Anexo 5: Modelo de Caso de abuso ..... 108**

**Anexo 6: Matriz de Control de Acceso ..... 110**

**Anexo 7: Proceso de respuesta a incidentes ..... 112**

**Anexo 8: Plan de Divulgación ..... 115**

**Anexo 9: Métricas ..... 116**

**Anexo 10: Plantilla Hoja de Vida de Indicador ..... 117**

**Anexo 11: Formato de Reporte de Incidentes ..... 119**

**Anexo 12: Plantilla de Casos de Abuso ..... 123**

**Anexo 13: Plantillas de Gestión de Riesgos..... 124**

**Anexo 14: Escaneo de Vulnerabilidades ..... 125**

**Análisis..... 130**

**BIBLIOGRAFÍA ..... 132**

# ÍNDICE GENERAL DE TABLAS

**Tabla 1: Metodologías de Desarrollo Web** ..... 39
**Tabla 2: Niveles de Modelo de Desarrollo seguro SAMM** ..... 48
**Tabla 3: Prácticas del modelo SDL** ..... 56
**Tabla 4: Cuadro comparativo – Criterios Principales** ..... 58
**Tabla 5: Cuadro comparativo – Criterios Secundarios** ..... 60
**Tabla 6: Cuadro comparativo – Criterios de Seguridad** ..... 63
**Tabla 7 : Características que no poseen ninguna de las dos metodologías** ..... 64
**Tabla 8: Características que solo posee SAMM** ..... 65
**Tabla 9: Recursos de Empresa de Caso de Estudio** ..... 68
**Tabla 10: Proceso y Responsabilidades** ..... 94
**Tabla 11: Lista de Escenarios** ..... 96
**Tabla 12: Probabilidad de Ocurrencia** ..... 97
**Tabla 13: Descripción de Niveles de Riesgo** ..... 98
**Tabla 14: Estimación informal de nivel de impacto** ..... 99
**Tabla 15: Capacitación Fase 1** ..... 99
**Tabla 16: Capacitación fase 2** ..... 100
**Tabla 17: Roles Generales** ..... 100
**Tabla 18: Capacitación Fase 3 – Fase 4** ..... 101
**Tabla 19: Presupuesto** ..... 101
**Tabla 20: Entregables luego de Capacitación** ..... 103
**Tabla 21: Modelo de Amenazas** ..... 103
**Tabla 22: Etapas del Modelo de Amenazas** ..... 104
**Tabla 23 : Plantillas Categorización de Amenazas** ..... 106
**Tabla 24: Plantillas Estimación Perfil de Atacante** ..... 108
**Tabla 25: Perfiles de Trabajo en la Organización** ..... 110
**Tabla 26 : Métricas generales** ..... 116
**Tabla 27: Hoja de Vida de Indicador** ..... 118
**Tabla 28: Reporte de Incidentes** ..... 119
**Tabla 29: Categorización del Incidente** ..... 121
**Tabla 30: Detalle Solución del Incidente** ..... 122
**Tabla 31: Plantilla de Casos de Abuso** ..... 123
**Tabla 32: Plantilla de Gestión de Riesgo** ..... 124

# ÍNDICE GENERAL DE ILUSTRACIONES

ILUSTRACIÓN 1: HOGARES CON ACCESO A INTERNET ..... 4
ILUSTRACIÓN 2: DENSIDADES INTERNET DE ECUADOR ..... 4
ILUSTRACIÓN 3: PERSONAS QUE HAN USADO INTERNET ..... 5
ILUSTRACIÓN 4: PERSONAS QUE HAN USADO INTERNET EN EL 2017, SEGÚN ÁREA Y SEXO ..... 5
ILUSTRACIÓN 5: PERSONAS ANALFABETAS DIGITALES ..... 6
ILUSTRACIÓN 6: PERSONAS QUE USAN TIC ..... 6
ILUSTRACIÓN 7: TIC EN EMPRESAS GRANDES ..... 7
ILUSTRACIÓN 8: TIC EN EMPRESAS GRANDES ..... 8
ILUSTRACIÓN 9: ESPECIALISTAS EN TIC EN EMPRESAS GRANDES ..... 9
ILUSTRACIÓN 10: ESPECIALISTAS EN TIC EN EMPRESAS PEQUEÑAS ..... 9
ILUSTRACIÓN 11: EMPRESAS QUE INVIERTEN EN TIC ..... 9
ILUSTRACIÓN 12: INVERSIONES POR SECTOR ECONÓMICO ..... 10
ILUSTRACIÓN 13: COMPRAS Y VENTAS POR INTERNET ..... 10
ILUSTRACIÓN 14: TIC EN EMPRESAS, INTERNET Y APLICATIVA WEB ..... 11
ILUSTRACIÓN 15: MEDIDAS DE SEGURIDAD ..... 13
ILUSTRACIÓN 16: MODELO OSI ..... 17
ILUSTRACIÓN 17: MODELO OSI Y TCP/IP ..... 18
ILUSTRACIÓN 18: PROTOCOLOS DE ALTO NIVEL ..... 18
ILUSTRACIÓN 19: TOP 10 2017 RIESGOS EN SEGURIDAD DE APLICACIONES ..... 24
ILUSTRACIÓN 20: INCIDENTES HASTA SEPTIEMBRE 2017 ..... 28
ILUSTRACIÓN 21: INCIDENTES HASTA SEPTIEMBRE 2017 ..... 29
ILUSTRACIÓN 22: ACTORES QUE EXPLOTAN VULNERABILIDADES ..... 31
ILUSTRACIÓN 23: PRINCIPALES ATAQUES ..... 32
ILUSTRACIÓN 24: MOTIVOS DE ATAQUES ..... 32
ILUSTRACIÓN 25: ATAQUES A LA CAPA DE APLICACIÓN Y RED ..... 33
ILUSTRACIÓN 26: ATAQUES MÁS FRECUENTES (POSITIVE TECHNOLOGIES) ..... 34
ILUSTRACIÓN 27: CAPACIDADES DE SEGURIDAD (CISCO) ..... 35
ILUSTRACIÓN 28: ACCIONES PARA RESPALDAR LA SEGURIDAD ..... 35
ILUSTRACIÓN 29 : PROPIEDADES DE SOFTWARE SEGURO ..... 39
ILUSTRACIÓN 30: PROPIEDADES CONTUNDENTES ..... 41
ILUSTRACIÓN 31: METODOLOGÍAS DE DESARROLLO SEGURO ..... 43
ILUSTRACIÓN 32: MODELO DE AMENAZAS ..... 45
ILUSTRACIÓN 33: MODELO DE DESARROLLO SEGURO DE SAMM ..... 47
ILUSTRACIÓN 34: NIVELES DE MADUREZ SDL ..... 51
ILUSTRACIÓN 35: FASES DEL MODELO SDL ..... 52
ILUSTRACIÓN 36: CALIFICACIONES DE REVISIÓN ..... 68
ILUSTRACIÓN 37: TARJETA DE CALIFICACIÓN ..... 69
ILUSTRACIÓN 38: PLAN DE IMPLEMENTACIÓN ..... 71
ILUSTRACIÓN 39: FASES ..... 72
ILUSTRACIÓN 40: PLANTILLA PARA PLAN DE IMPLEMENTACIÓN DE PROVEEDORES INDEPENDIENTES DE SOFTWARE ..... 73
ILUSTRACIÓN 41: FLUJO DE ANÁLISIS DE RIESGOS ..... 93
ILUSTRACIÓN 42: FLUJO DE ATAQUES ..... 94
ILUSTRACIÓN 43: GRAVEDAD DE RIESGO (PROBABILIDAD E IMPACTO) ..... 96

**ILUSTRACIÓN 44: SEVERIDAD DE RIESGO** ..... 97
**ILUSTRACIÓN 45: FLUJO DE "THREAT ANALYSIS AND MODELING"** ..... 103
**ILUSTRACIÓN 46 : FLUJO DE CASOS DE ABUSO** ..... 108
**ILUSTRACIÓN 47 : CASO DE ABUSO** ..... 109
**ILUSTRACIÓN 48: RESPUESTA A INCIDENTES** ..... 112
**ILUSTRACIÓN 49: NIVELES DE ALERTAS HERRAMIENTA ZAP OWASP** ..... 125
**ILUSTRACIÓN 50: ALERTAS DE LA APLICACIÓN "PINDAL"** ..... 131

# INTRODUCCIÓN

En la actualidad la tecnología forma parte de nuestra vida, proyectándose en diferentes áreas de la sociedad, ayudando al desarrollo de la misma. Lamentablemente no se ve libre de atacantes o terceros que pueden provocar daños aprovechando las vulnerabilidades que puede presentar.

La información se ha vuelto uno de los recursos más valiosos para una persona, organización o empresa. Por lo cual se han visto obligados a tomar medidas preventivas para evitar el robo de la misma.

No solo las organizaciones han tomado medidas preventivas, sino también los programadores o desarrolladores de sistemas o páginas web. Debido a que muchas veces nuestra forma de programar no es del todo segura. Programamos de acuerdo con cómo nos sintamos cómodos, lo cual muchas veces causa que nuestros sistemas no poseen la debida seguridad que puedan evitar o prevenir un ataque o amenaza de un tercero. Esto puede causar pérdidas a la organización que implemente nuestro sistema, como también desprestigio a nuestra persona como programador. Ya que no tomamos las medidas necesarias para que un atacante no puede aprovecharse de alguna de las vulnerabilidades que pueda existir.

La mayoría de los sistemas que se han desarrollado no presentan un código seguro, lo cual ha causado no solo la explotación de las vulnerabilidades que presenta el sistema, sino también fallos en el software, como, por ejemplo: en misiones espaciales, un error de programación en el cohete Ariane 5 provoco su autodestrucción en menos de un minuto luego de su despegue, en 1996 (Enrique Jaziel Ulloa Castillón, n.d.). En la medicina un error en el código de la maquina Therac 25 que se encargaba del control de radiación a la que se exponían los pacientes en sesiones de radioterapia, entre 1985 y 1987, causo un exceso de radiación en 6 personas (Nancy G. Leveson, 1993). Estos y

entre otros problemas que han surgido a lo largo del tiempo, causados por programadores que no realizan un código seguro.

Unas de las principales razones de porque un programador no desarrolla programas con código seguro es debido a tres factores que son: tiempo, costo y esfuerzo. Ya que esto requeriría de emplear métodos de verificación formal del código, como también realizar las pruebas pertinentes para verificar el funcionamiento del sistema.

# OBJETIVOS

## Objetivos Generales

Propuesta de una guía de seguridad enfocada al código seguro, basado en las contramedidas y metodologías de seguridad a nivel de la capa de aplicación

## Objetivos Específicos

- Analizar la seguridad a nivel de la capa de aplicación del modelo OSI
- Examinar las principales vulnerabilidades en la capa de aplicación
- Evaluar los principales ataques y sus patrones en la capa de aplicación
- Comparar las metodologías de prevención de desarrollo seguro
- Diseñar un prototipo para la capa de aplicaciones que permita el desarrollo seguro.

# CAPÍTULO 1: MARCO TEÓRICO

## 1.1. Ciberseguridad

La búsqueda de conocimiento del ser humano para la creación de instrumentos que puedan satisfacer sus necesidades ha impulsado la constante evolución y desarrollo de la tecnología. Muchos de los inventos que ha creado el hombre han sido esenciales para el avance de la humanidad.

La tecnología ha evolucionado de tal manera de que se ha convertido en algo casi imprescindible para el ser humano. Hoy en día es parte de nuestra vida, debido a que realizamos uso de ella la mayor parte del día. Por ejemplo, un estudio que fue realizado en el 2010, a 2002 estudiantes, por Kaiser Foundation en los Estados Unidos, sobre el uso de medios de comunicación. Muestra que jóvenes de 8 a 18 años, pasaban más de siete horas y media al día utilizando algún equipo tecnológico en su tiempo de ocio. Entre ellos TV, computadora, juegos de video, reproductores de sonido, etc. (Rideout, Foehr, & Roberts, 2011)

La ciberseguridad se le conoce como el uso responsable de las TIC, esto incluye Internet, dispositivos móviles y de comunicación, así como también dispositivos utilizados para guardar, compartir o recibir información. (Giant, 2016) Llevándolo a cabo por medio de herramientas, políticas, conceptos de seguridad, salvaguardas de seguridad, métodos de gestión de riesgos, etc. Garantizando la disponibilidad, integridad, autenticidad, confidencialidad y no repudio del sistema, así como la protección de la privacidad de los usuarios del ciberentorno. (De, 2015)

Se considera como parte del ciberentorno no solo a los usuarios que realizan uso de este, sino también a los dispositivos informáticos, sus aplicaciones, servicios y sistemas que están directamente o no conectados al internet o al entorno de las redes de próxima generación (NGN). De igual forma incluye el software que se encuentra ejecutándose en estos dispositivos, su información que se

encuentra almacenada, transmitida o que estos generan, las instalaciones y edificios en los que se encuentran. (De, 2015)

La ciberseguridad se centra en el conjunto de actividades por medio de mecanismo de defensivos y ofensivos, que se encargan la protección sobre el uso indebido que se realiza en el ciberespacio, y al mismo tiempo de proteger los servicios y la información de los usuarios, ante riesgos de ataques informáticos o de amenazas. (Servitja Roca, 2013)

El avance tecnológico, trae consigo la creación y el desarrollo de nuevos dispositivos tecnológicos, que en la actualidad ya no solo se encuentran accesibles para una pequeña parte de la población, sino tanto adultos como jóvenes hacen uso de estos dispositivos, ya sea en el uso de redes sociales, chats en línea o el uso de correos electrónicos, etc.

Por ejemplo, el internet, en donde se pueden encontrar conectas miles de millones de personas, de diferentes lugares del mundo. Esta herramienta nos permite obtener una gran variedad de información al alcance de nuestras manos. Pero no muchas personas somos conscientes de la responsabilidad que traen el uso de dispositivos tecnológicos, haciendo uso de estos de manera imprudente y muchas veces sin las precauciones necesarias.

Debido a que la tecnología no solo trae beneficios, debemos estar conscientes de los riegos que puede conllevar su uso. Por eso es recomendable que, al hacer uso de las TIC, lo hagamos con sentido común y de responsabilidad, así como también teniendo una apreciación moral de lo que es correcto e incorrecto. (Jordá Capitán, Priego Fernández, & Sádaba, 2014)

### 1.1.1. Ciberseguridad y Protección en la Red

El Internet se ha convertido en una herramienta, la cual puede influir en el comportamiento que posee la población y las empresas. La red es un nuevo ámbito de actividad que permite a miles de millones de personas conectarse al ciberespacio. (Jordá Capitán et al., 2014)

La Red cuenta con aproximadamente **4.156.932.140** usuarios en la actualidad. De los cuales **345.660.847** pertenecen a Norte América. (Internet World Stats, 2017)

#### 1.1.1.1. Hogares

Ecuador cuenta con una población de **17.192.571**, de los cuales **10586476** son usuarios que han usado de Internet hasta diciembre del 2017. ("INEC," 2016)

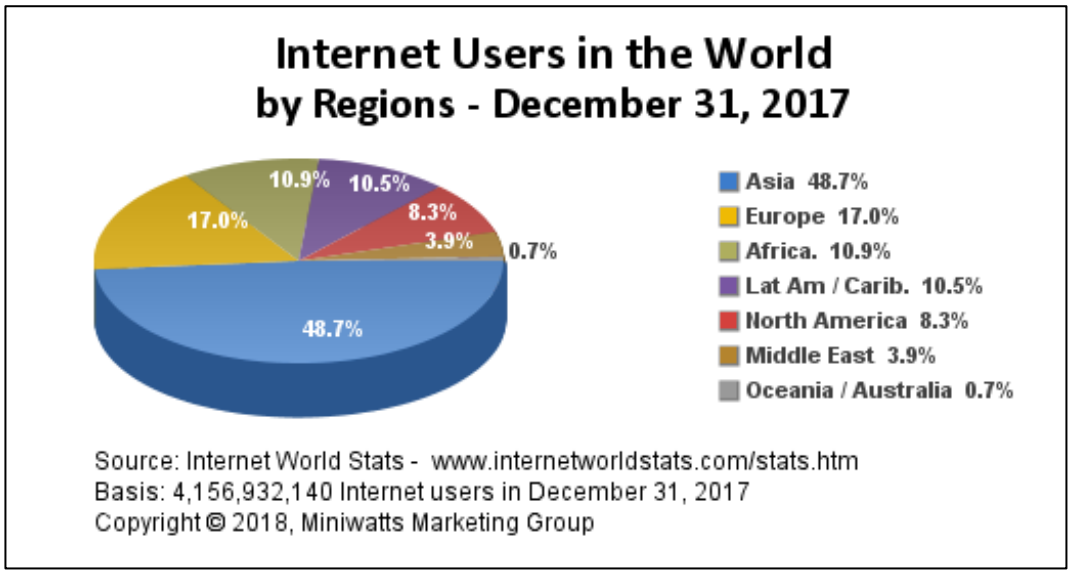



***Ilustración 0: Distribución Geográfica de los Usuarios de Internet 2017***

***Fuente:*** *(Internet World Stats, 2017)*

#### 1.1.1.2. Personas

De acuerdo con el Instituto Nacional de Estadísticas "INEC". Según las estadísticas realizadas en el 2017, el **37,20 %** de los hogares ecuatorianos tienen acceso al Internet. De los cuales 8 de cada 10 jóvenes entre **16 y 24 años** son el grupo de personas con más uso de internet. (Observatorio TIC, 2017)




***Ilustración 1: Hogares con Acceso a Internet***

***Fuente:*** *(Observatorio TIC, 2017)*

- La mayor densidad de Internet es la de banda ancha fija + móvil, con un porcentaje de 63,10. (Observatorio TIC, 2017)

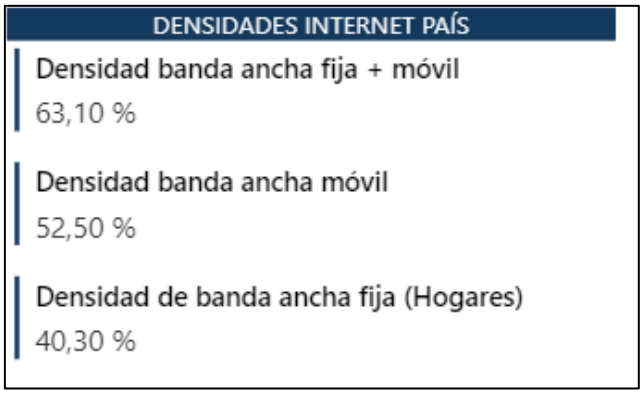



***Ilustración 2: Densidades Internet de Ecuador***

***Fuente:*** *(Observatorio TIC, 2017)*

- 1.779.397 ecuatorianos utilizan internet con acceso fijo, siendo una cantidad mayor a los usuarios que utilizan internet por medio de dispositivo móvil (8.807.079 usuarios).
- En el 2017 se registra que el 91,98% de la población ecuatoriana utiliza Internet en su Teléfono, incrementado en un 22,15% desde el 2011.
- El 21.11% de personas utiliza el internet para Educación y el 3,26% lo utiliza para el Trabajo
- El 58,30 % de la población ecuatoriana han usado Internet en el 2017, aumentando un 22.71% desde el 2009.




***Ilustración 3: Personas que han usado Internet***

***Fuente:*** *(Observatorio TIC, 2017)*

- La población que ha hecho más uso del internet es la población masculina con 59.32 %, con una diferencia de 2% ante la población femenina.
- El área Urbana realiza más uso de internet con 66.87%

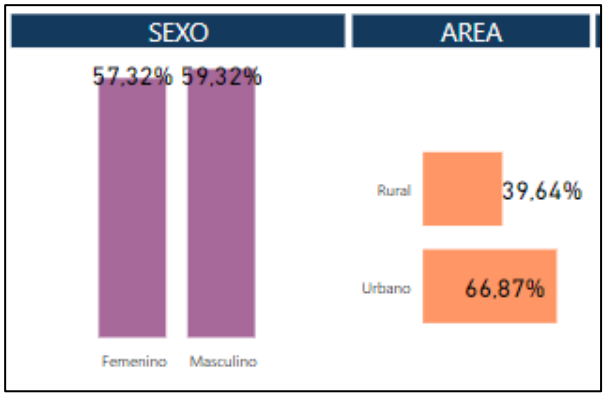



***Ilustración 4: Personas que han usado internet en el 2017, según área y sexo***

***Fuente:*** *(Observatorio TIC, 2017)*

En el 2017, el 10,48% de las personas en el Ecuador son analfabetas digitales de 15 a 49 años. Se considera analfabetas digitales a las personas que:

- No tiene celular activado
- No ha utilizado computadora
- No ha utilizado internet.

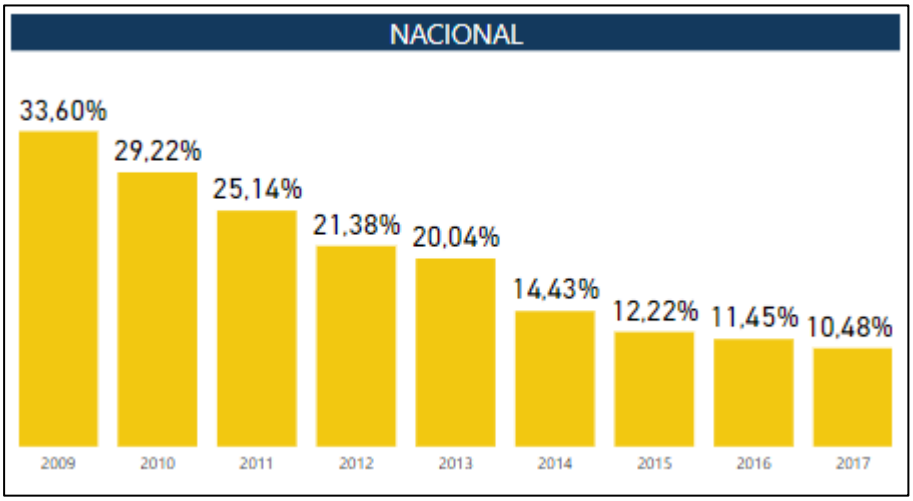



***Ilustración 5: Personas analfabetas digitales***

***Fuente:*** *(Observatorio TIC, 2017)*

- El 60,68 de la población ecuatoriana hacen uso de las TIC, incrementando un 26.66% desde el 2009.

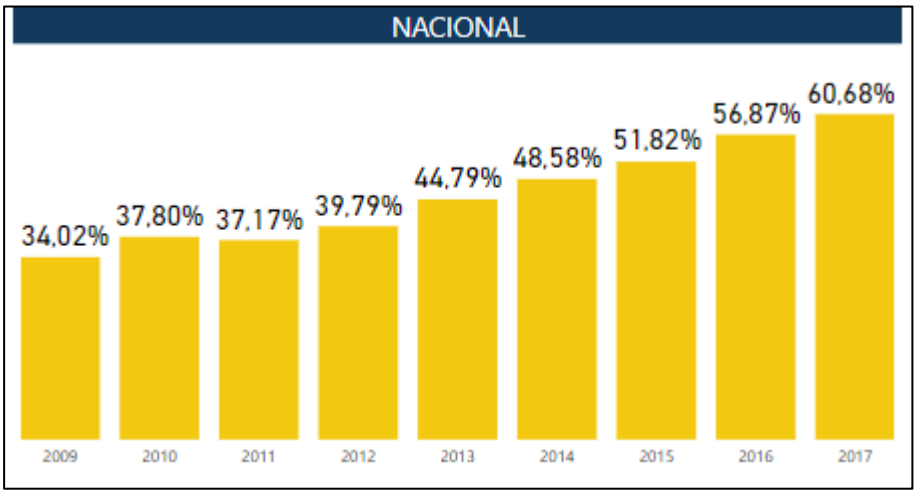



***Ilustración 6: Personas que usan TIC***

***Fuente:*** *(Observatorio TIC, 2017)*

#### 1.1.1.3. Empresas

Con base a los datos objetivos del INEC en el 2015, sobre las TIC y las empresas. Se toma una población objetivo a **3777 empresas,** los datos recolectados son anuales desde el 2012 hasta el 2015**.** (Observatorio TIC, 2017)

- El 98.1% de las empresas ecuatorianas con conexión a internet usan navegadores de Internet, incrementando en 2.9% desde el 2012. Y el 18.8% de estas usan aplicaciones con código abierto, con una diferencia de 11.3% desde el 2012.
- En el 2015 se registra que el 98.5% de empresas grandes utilizan navegadores de Internet.
    - El 98.1% de las empresas grandes comerciales utilizan navegadores de internet, y de empresas grades de servicios es el 98.9%.
- El 26.1% de Empresas Grandes utiliza aplicaciones con código abierto.
    - El 30.7% de empresas grandes comerciales utilizan aplicaciones con código abierto, y de empresas grandes de servicios el 29.4%.

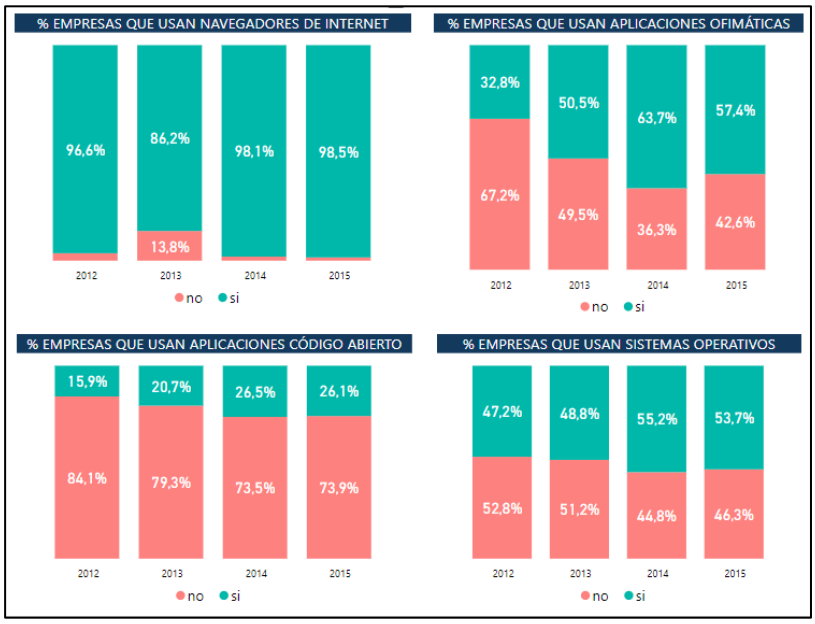



***Ilustración 7: TIC en Empresas Grandes***

***Fuente: (Observatorio TIC, 2017)***

- En las empresas pequeñas en el 2015 se registra que el 97.3% utilizan navegadores de Internet.
    - El 97.7% de las empresas pequeñas comerciales utilizan navegadores de internet, y de empresas pequeñas de servicios es el 96.3%.
- El 9.1% de empresas pequeñas utiliza aplicaciones con código abierto.
    - El 13.7% de empresas pequeñas comerciales utilizan aplicaciones con código abierto, y de empresas pequeñas de servicios el 5.9%.

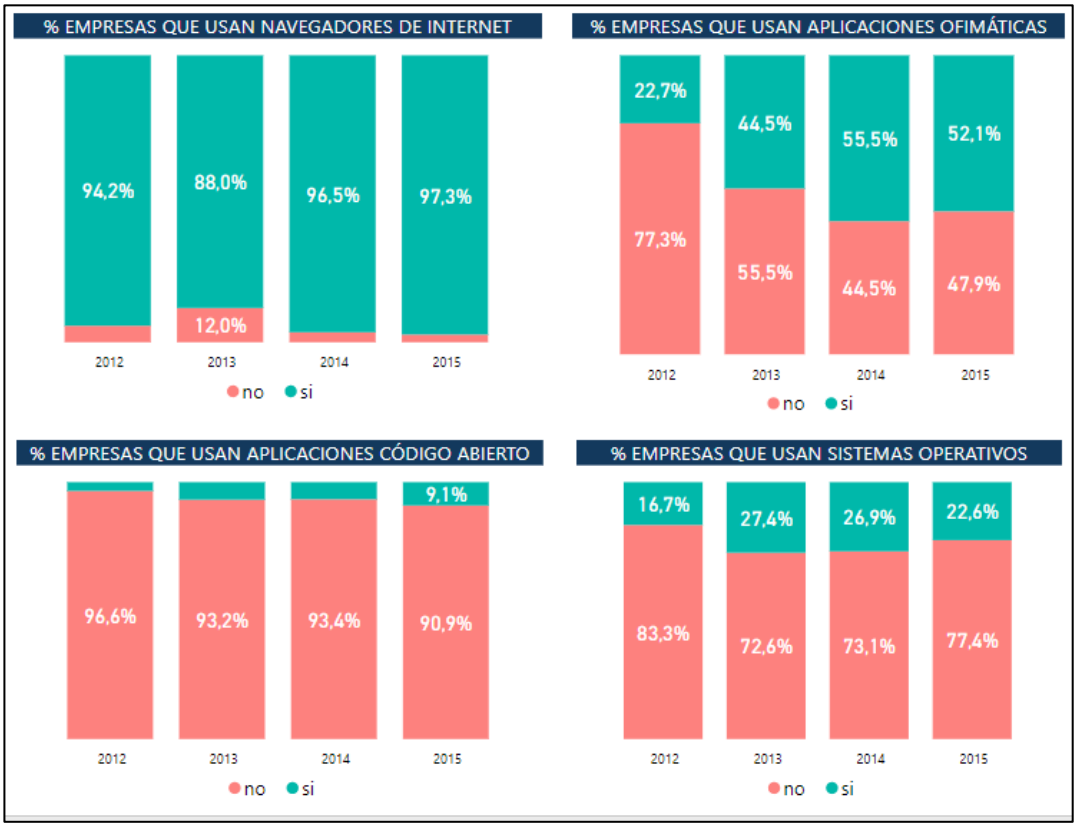



***Ilustración 8: TIC en Empresas Grandes***

***Fuente:*** *(Observatorio TIC, 2017)*

- El 38.9% de empresas ecuatorianas utilizan especialistas en TIC, aumentando un 3.4% desde el 2012.
    - El 63.9% pertenecen a empresas grandes, y el 12.5% a empresas pequeñas.

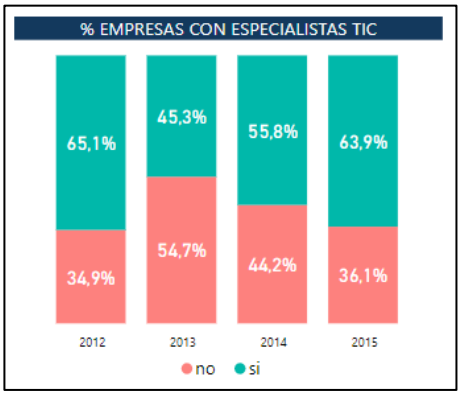



***Ilustración 9: Especialistas en TIC en Empresas Grandes***

***Fuente:*** *(Observatorio TIC, 2017)*

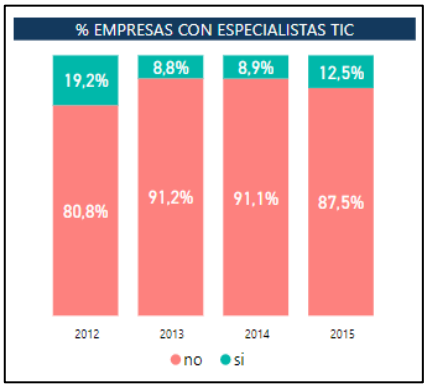



***Ilustración 10: Especialistas en TIC en Empresas Pequeñas***

***Fuente:*** *(Observatorio TIC, 2017)*

- En el 2015, 76,1% de empresas invierten en TIC. Las inversiones engloban: compra de dispositivos físicos, software o aplicaciones informáticas. Desde el 2012 se incrementó un 22%.
- 2164 empresas realizan inversiones en TIC. De las cuales 983 son empresas grandes, 291 medianas A, 552 medianas B, 8 microempresa y 330 pequeñas empresas.

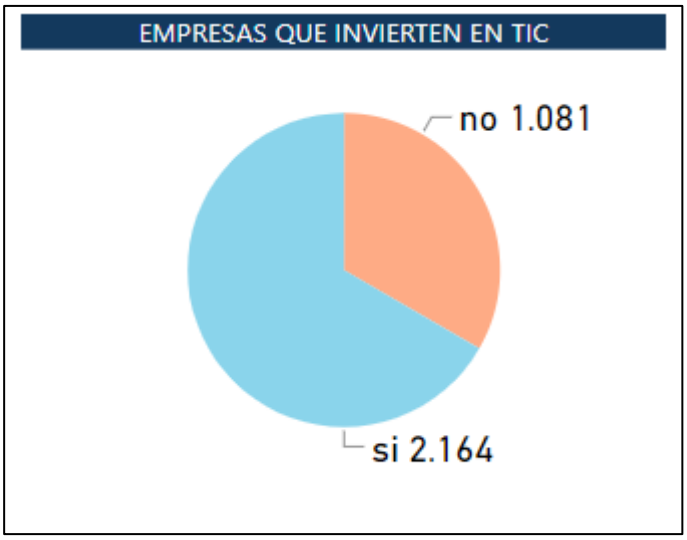



***Ilustración 11: Empresas que invierten en TIC***

***Fuente:*** *(Observatorio TIC, 2017)*

- En el 2015 las empresas de Servicios tuvieron más inversiones en TIC, con un total de 125.7 mil.

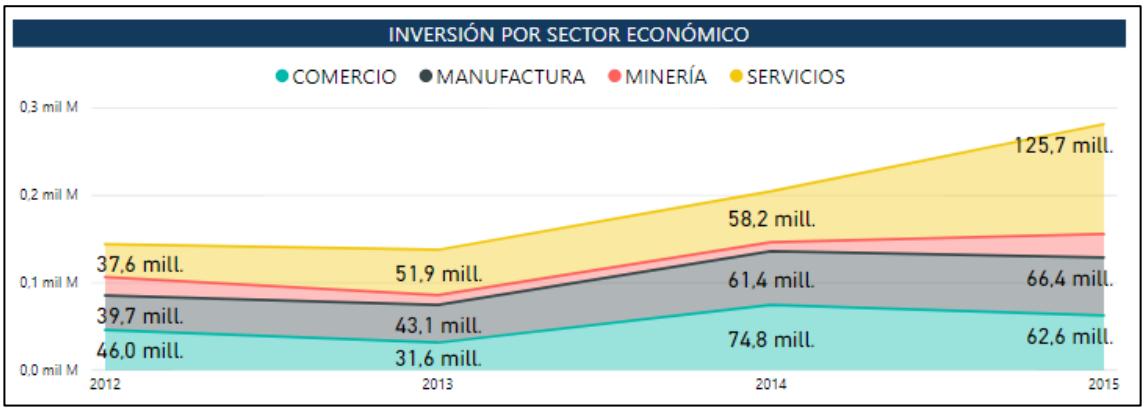



***Ilustración 12: Inversiones por Sector Económico***

***Fuente:*** *(Observatorio TIC, 2017)*

- 541 empresas realizan transacciones por Internet. Desde el 2012 se disminuyó el número de empresas que realiza transacciones por Internet en un 1,47%.
- En el 2015 el sector económico con más compras de Internet es el de Comercio con 28.33% y el sector más ventas de internet es el de Minería con 50%. Hubo una reducción en las compras por internet en un 16,12% desde el 2012 (Se mantiene el sector comercial). Se incrementa en un 41% las ventas en el sector de Minería desde el 2012, se debe tomar en cuenta que en el 2012 se registra al sector de servicios con mayores ventas con un total de 36,51%.

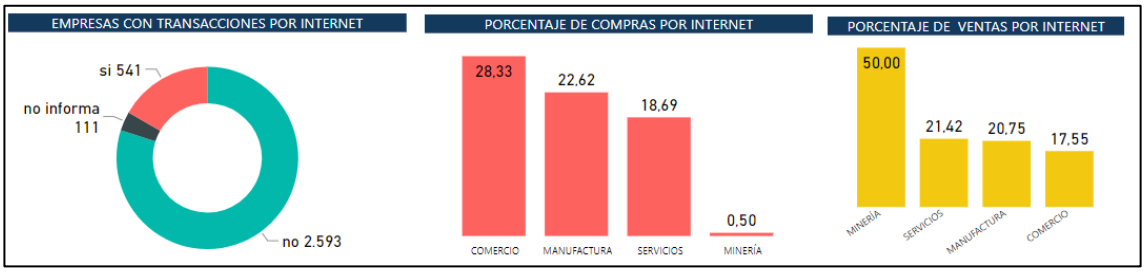



***Ilustración 13: Compras y ventas por Internet***

***Fuente:*** *(Observatorio TIC, 2017)*

- En el 2015 se registra al 96.6% de empresas ecuatorianas con conexión a internet, con un incremento de 0,78% desde el 2012.
- El ingreso de empresas a internet es mayor por medio de una banda de ancha fija con 96.8 % y 17.2 % por medio de banda ancha móvil.
- 61,4% de empresas con conexiones a internet disponen de página web
- 95,3% de las empresas utilizan correo electrónico para recibir o enviar información.

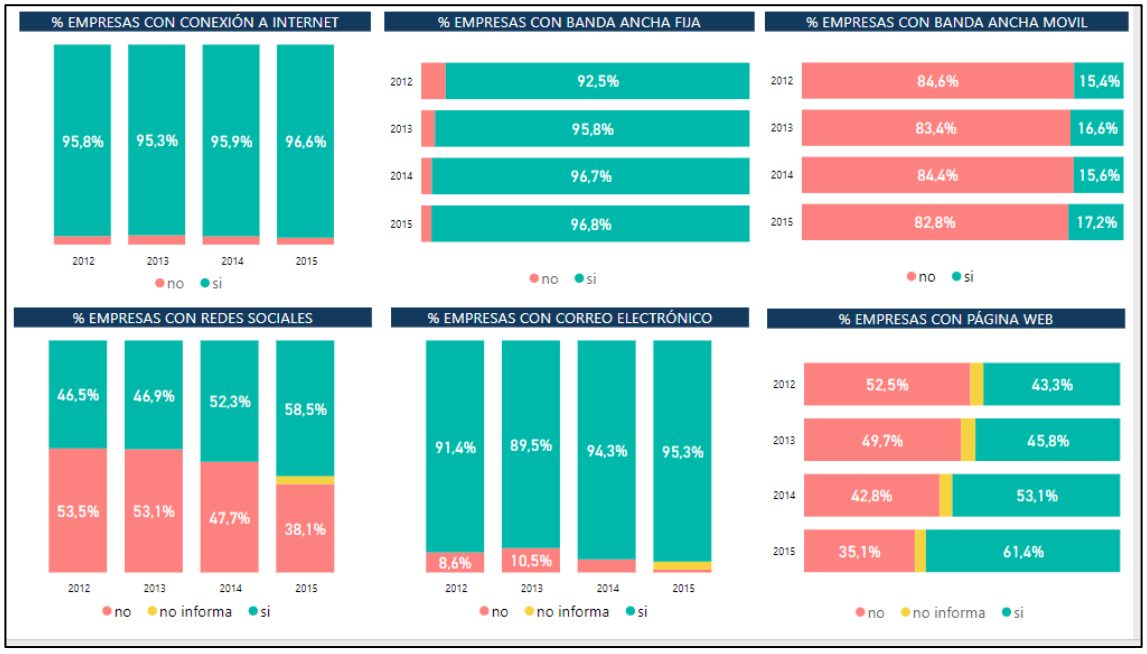



***Ilustración 14: TIC en empresas, internet y aplicativa web***

***Fuente: (Observatorio TIC, 2017)***

Ecuador es un país que se ha desarrollado lentamente en el ámbito de tecnología a diferencia de las grandes potencias mundiales. En la actualidad tanto personas como organizaciones han adoptado el uso de la tecnóloga en sus vidas. Como se puede observar en las estadísticas realizadas por el INEC, el uso de Internet y de TIC ha ido incrementado potencialmente con el paso de los años. (“INEC,” 2016) En el 2009 se registra al 33.6% de la población ecuatoriana como analfabeta digital, hasta el 2017 esta población ha decrecido en un 23.12%. (Observatorio TIC, 2017)

En el mundo existen diferentes tipos de usuarios, con pocos o varios conocimientos sobre el universo de los servicios que nos ofrece la Red, como también en los diferentes tipos de tecnologías de la información. Pero como se mencionó anteriormente no solo son beneficios los que provee. Sino también preocupaciones y riesgos, es por eso por lo que se debe destacar que la seguridad y la confianza son esenciales en el momento que nosotros utilizamos la Red y los servicios que provee. (Jordá Capitán et al., 2014)

La ciberseguridad adopta diferentes medidas de seguridad con el propósito que el usuario no solo haga uso responsable de las TIC, sino también prevenir incidencias que pueden causar riesgos de seguridad sobre su información. Tanto personas como organizaciones quieren hacer uso de la red de manera segura para la extracción de información, por eso es importante la seguridad y la confianza que pueda proveer al usuario al momento de usarla, así como también sus servicios. (Jordá Capitán et al., 2014)

Intelco realizó un estudio sobre la Ciberseguridad y la confianza que genera en los hogares españoles y el uso de internet, en el cual el 45,1% de la población tiene bastante y mucha confianza en el internet y el 1,1% desconfían de este. Se realizó el análisis de la adopción de medidas de seguridad y los niveles de incidencias en situaciones que generar riesgos de seguridad. (Inteco, 2014)

Para este estudio se utilizó software iScan desarrollado por el mismo, el cual analiza los sistemas y detecta el malware que puede estar en los equipos, así como también la recomendación de políticas y medidas de seguridad para la generación de confianza en el uso de la red. (Inteco, 2014)

Intelco propone 4 tipos de medidas de seguridad, entre ellas:

- Las que tienen intervención directa del usuario.
    - **Automatizables:** Medidas de carácter pasivo que no requieren acción del usuario

- **No automatizables:** Medidas de carácter activo que si requieren de intervención del usuario.

- Medidas de antes o después que ocurra una incidencia.
  - **Medidas Proactivas:** Prevenir y evitar incidencias de seguridad.
  - **Medidas Reactivas:** Eliminar amenazas o incidencias conocidas.

(Inteco, 2014)

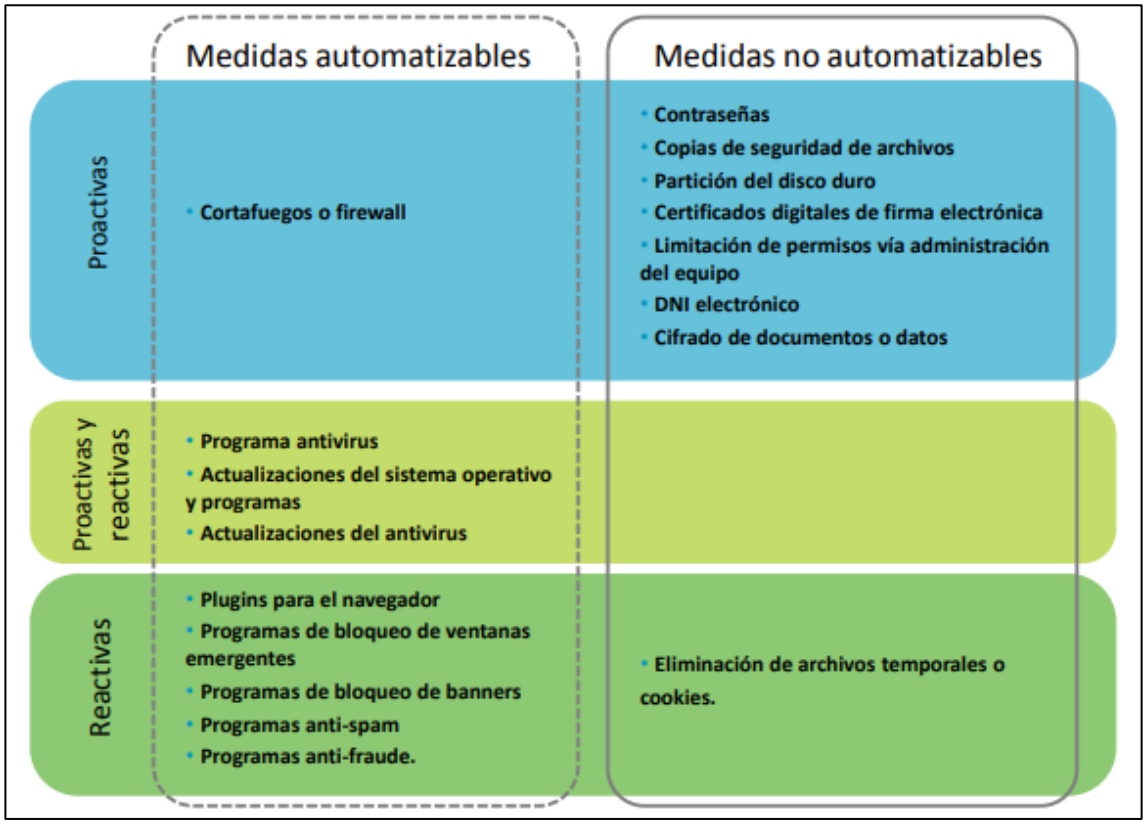



***Ilustración 15: Medidas de Seguridad***

***Fuente: (Inteco, 2014)***

El 45,1% de la población tiene bastante y mucha confianza en el internet y el 1,1% desconfían de este. En el 2013 según indica las estadísticas del estudio como medida automatizable más usada a programas de antivirus con un 83**,9%** de la población, y la que se encuentra en segundo lugar es la actualización de antivirus con el **71,3%**, esto se puede generar debido a que los usuarios generalmente piensan que el antivirus es la única o mejor solución. La medida

menos usada es los Programas o configuración para bloqueo de publicidad con el **18,6%** de la población. Entre las medidas más usadas de las medidas de seguridad no automatizable están, las contraseñas con un 59**,4 %** y la eliminación de archivos temporales y cookies con el **55,2%.** Y la medida menos usada es el Cifrado y documentos de Datos con el **6.3%**. (Inteco, 2014)

El uso de la Red ha incrementado con forme ha evolucionado la humanidad, es una de las razones porque es importante la ciberseguridad ya que esta tiene como objetivo principal la protección del ciberentorno y la generación de seguridad y confía a los usuarios que lo utilizan, por medio de herramientas y medidas de seguridad que han evolucionado conforme pasan los años. Aunque muchas los usuarios tomamos solo las medidas de seguridad que son conocidas y omitimos las soluciones que pueden ser iguales o mejores a estas.

### 1.2. Capas del Modelo OSI

El modelo OSI (Modelo de Interconexiones de Sistemas Abiertos) fue desarrollado por la ISO (Organización Internacional de Estándares). Convirtiéndose en un marco de referencia para el desarrollo de protocolos estándares OSI. (Aldana, 2009) Es decir, fue creado con el propósito de que se convierta en un modelo estándar para futuros diseños de protocolos que están basados en capas. (Merinero, 2010)

ISO define una estructura para el modelo OSI, la cual consiste en: debe ser una estructura multinivel, que permita que cada nivel resuelva una parte de la comunicación, existen interfaces o puntos de acceso a los servicios entre los niveles, cada nivel será dependiente tanto del nivel superior como del inferior y por último que se incorporara un formato de control o encabezado. (Gerónimo, 2009)

OSI fue desarrollado con el propósito que crear una arquitectura que permia la comunicación entre diferentes computadoras. (Aldana, 2009) Este modelo

consiste en siete capas: La Capa de Aplicación, La Capa de Presentación, La Capa de Sesión, La Capa de Transporte, La Capa de Red, La Capa de Enlace de Datos y La Capa Física. (Aldana, 2009)

Cada capa tiene sus funciones, ofreciendo sus servicios a las superiores y ocultando la dificultad de las capas inferiores. Las reglas que rigen la comunicación entre capas se las define como protocolos, existen dos tipos de protocolos: los protocolos orientados a conexión, que realizan una conexión antes del envió de información por el canal de comunicación, de igual forma realizan una desconexión una vez se finalice la transición. Y los protocolos no orientados a conexión, estos protocolos a diferencia de los anteriores no necesitan el establecimiento previo de conexión para él envió de información. (Lopéz, 2005)

OSI es un modelo diseñado bajo ciertos principios. Entre los cuales se establece que cada capa que constituya el modelo deber realizar una función, la cual se encuentre bien definido. De igual forma, las funciones de cada capa deben ser definidas con la idea de determinar los protocolos estándares que se usaran en ella. Los límites de cada paca deben ser establecidos de forma que se minimice el flujo entre las interfaces. Y, por último, que el número de capas sea suficiente alto que permita abarcar las funciones que se necesitan y a su vez lo bastante pequeño para que la arquitectura de la red se pueda manejar fácilmente. (Lopéz, 2005)

Como se mencionó anteriormente este modelo está formado por 7 capas:

### 1.2.1. Capa física

Tiene como función principal la transición de cadenas de bits a través del medio de transmisión físico. (Hillar, 2004)

### 1.2.2. Enlace de datos

Su función es transmitir los datos en forma de tramas, que se obtienen al transformar la cadena de bits enviada por la capa física por medio del medio de trasmisión físico. (Hillar, 2004)

### 1.2.3. Red

Se encarga del enrutamiento de paquetes desde su origen hasta el destino entre redes homogéneas o heterogéneas. Es decir que esta capa tiene el control de operación de la red, determinando el camino del paquete por medio de los nodos de la red. (Hillar, 2004)

### 1.2.4. Transporte

Su función es la trasferencia de datos entre su origen y el destino, brindando una comunicación sea fiable y libre de errores. (Hillar, 2004)

### 1.2.5. Sesión

Su función gestionar el control (abre, administra, mantiene y cierra la conexión) de las comunicaciones entre las usuarios o aplicaciones, un protocolo frecuente en esta capa es la utilización de texticos, el cual indica que estreno es el que tiene derecho a hablar. (Hillar, 2004)

### 1.2.6. Presentación

Esta capa se encarga del manejo de la sintaxis y la semántica de la información que se trasmitió. (Hillar, 2004)

### 1.2.7. Capa de Aplicación

Esta capa es el punto de ingreso al modelo de capas. Se implementan protocolos que permitan la interconexión entre varios equipos. Uno de los protocolos más usados es el HTTP (Hypertext Transfer Protocol). (Hillar, 2004)




***Ilustración 16: Modelo OSI***

***Fuente: (Hillar, 2004)***

Sin embargo, el modelo del que más se realiza uso es el TCP/IP (Protocolo de control de la trasmisión), el cual surgió de la red **ARPANET**. (Hillar, 2004) Este modelo se utiliza más para fines prácticos, este modelo al igual que OSI consta de capas para el proceso de comunicación, su principal ventaja es que permite el enrutamiento y una salida transparente de un LAN a una WAN. (Hillar, 2004) A diferencia del modelo OSI este consta de 5 capas.

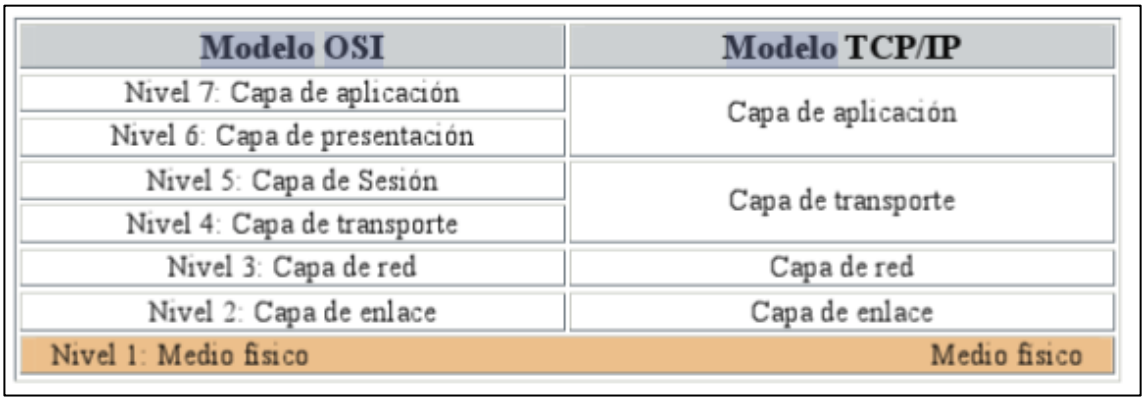



***Ilustración 17: Modelo OSI y TCP/IP***

***Fuente: (Lopéz, 2005)***

## 1.3. Capa de Aplicación

La capa de aplicación es una de las capas que compone el modelo OSI, esta capa está encargada de proveer el acceso al entorno OSI para los usuarios y los servidores de información. La capa de aplicación contiene todos los protocolos de nivel más alto como: TELNET, FTP y SMTP. (Aldana, 2009)

### 1.3.1. Protocolos de Alto Nivel

Tanto TCP/IP como el Modelo OSI imponen protocolos para cada una de las capas que estos poseen. En la siguiente imagen se especificará algunos de los protocolos que maneja cada una de las capas.

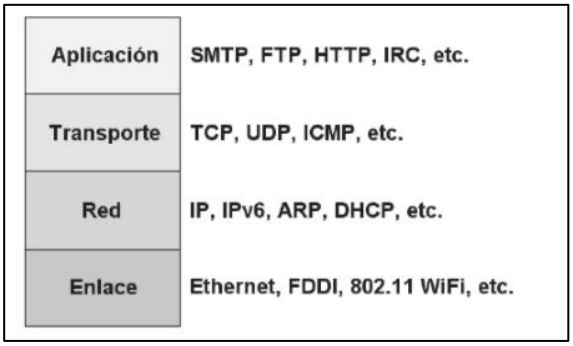



***Ilustración 18: Protocolos de alto nivel***

***Fuente:*** *(Marañón & García, 2004)*

En este caso se enfocará en la última capa, la capa de aplicación, esta capa como se mencionó anterior mente es la que vera el usuario final. Contiene una gran variedad de protocolos los cuales están orientados a diferentes servicios. Entre los protocolos que utiliza esta capa se mencionaran los más utilizados, tales como:

#### 1.3.1.1. SNMP

Simple Network Management Protocol. (Marañón & García, 2004) Este protocolo hace uso del puerto 161 y el 162, (Gregg et al., 2006) permitiendo al usuario el monitoreo y supervisión de redes, como también la recopilación de información de estas y el reporte de su estado. (Manzuik, Pfeil, & Gold, 2007)Posee 3 versiones actualmente, la versión 3 es la que ofrece más seguridad, esta provee una encriptación que hace que la seguridad sea más robusta a la que ofrecen las demás versiones. (Gregg et al., 2006)

#### 1.3.1.2. HTTP

Hyper Text Transfer Protocol. (Marañón & García, 2004) Este protocolo es uno de los más utilizados, opera en el puerto 80. HTTP utiliza un protocolo de respuesta, el cual consiste en que un usuario envía la solicitud y el servidor la respuesta, esto ayuda al intercambio de información en la navegación web. (Gregg et al., 2006) Este protocolo no es fuerte en cuanto a seguridad se refiere, ya que muchos de los ataques que se realizan se dirigen a la aplicación web que está en ejecución en este protocolo. (Manzuik et al., 2007)

#### 1.3.1.3. FTP

Fil Transfer Protocol. (Marañón & García, 2004) Este protocolo es el encargado de proporcionar acceso remoto a los archivos, usualmente utilizado para el mantenimiento del servicio web. (Gregg et al., 2006) Realiza uso de dos puertos:

- Puerto 20: empleado para el flujo de datos, es el puerto de salida, se usa para la trasferencia de datos entre cliente y el servidor. (Gregg et al., 2006)
- Puerto 21: puerto de servicio de entrada, se encarga de procesar los comandos FTP del cliente, en donde los comandos se usan para el inicio de la trasferencia de datos. (Manzuik et al., 2007)

Este protocolo es uno de los que más comúnmente son atacados, entre los ataques más comunes en este protocolo son:

- La navegación anónima y explotación de los defectos del SW
- Ataques de fuerza bruta
- Explotación de vulnerabilidades

(Manzuik et al., 2007)

#### 1.3.1.4. TELNET

Protocolo usado para la administración remota de un sistema (Marañón & García, 2004), realiza uso del puerto 23. Permite que el usuario establezca sesión con el sistema desde otro lugar. (Gregg et al., 2006) En cuanto a seguridad este protocolo es muy débil, entre los ataques frecuentes a este protocolo están los siguientes:

- Ataques de Fuerza Bruta
- Explotación de fallas en el SW del servidor TELNET

(Manzuik et al., 2007)

#### 1.3.1.5. DNS

Domain Name Service. (Marañón & García, 2004) Usa el puerto 53, se encarga de realizar la traducción de las direcciones, es decir que convierte nombres de

dominio (FQDN) a una dirección IP o viceversa. Este protocolo usa UDP para consultas DNS y TCP para transferencias. (Gregg et al., 2006)

#### 1.3.1.6. SMTP

Simple Mail Transfer Protocol. (Marañón & García, 2004) Opera en el puerto 25, se utiliza para el intercambio de correo electrónico vía red. SMTP cuenta de dos partes: el encabezado y el cuerpo. (Gregg et al., 2006) Tiene como principal función el envío y recepción de correos electrónicos. Como una de las principales vulnerabilidades de este protocolo están: Spoofing y Spam. (Manzuik et al., 2007) Debido a que no es muy seguro se desarrolló el estándar MIME, el cual asegura los datos del correo electrónico mediante cifrado. (Gregg et al., 2006)

#### 1.3.1.7. OTROS

- **NFS**: Network File System
- **SSH:** Secure Shell
- **RCP:** Remote Procedure Call

(Marañón & García, 2004)

## 1.4. Servicios WEB

Un servicio web se puede definir como una interfaz de software que permite describir un conjunto de operaciones a ejecutar, se basan en lenguaje XML. Estos servicios permiten compartir información o invocar funciones entre aplicaciones. (IBM, n.d.)

### 1.4.1. Análisis y evaluación de los servicios Web

La evolución tecnológica no solo benéfica a las empresas y la seguridad de la información, sino también a los ciber-atacantes, debido a que actualmente existen herramientas las cuales aprovechan las vulnerabilidades los sitios web y de los sistemas, haciendo que estos sean fáciles de atacar y de evadir su seguridad.

La mayor parte de ataques que se realizan en el ciberentorno son dirigidos a servicios web, esto se debe a las vulnerabilidades que estos poseen, a su vez la falta de prevención y contramedidas que se emplean al momento de su desarrollo.

#### 1.4.1.1. Vulnerabilidades y riegos en los sistemas

Incibe define a la vulnerabilidad en términos de informática como, “una debilidad o fallo en un sistema informático” (Incibe, 2017), estos agujeros pueden causar que el sistema sea más susceptible al ataque y se encuentre en riesgo, lo que significa que se comprometerá la integridad, disponibilidad y confidencialidad de la información (Incibe, 2015). A mayor cantidad de vulnerabilidades que posea el sistema, mayor es la probabilidad de que este sea atacado (Marañón & García, 2004). Estas pueden ser causadas por defectos en el diseño o fallos en la implementación, etc.

Una vulnerabilidad se la define como debilidades que se pueden presentar un sistema, las cuales pueden permitir a amenazas provocar daños, los cuales a su vez provocaran perdidas en una empresa. (Vieites, 2007)

En lo que respecta a un sistema como principales causas de las vulnerabilidades que pueden afectar a un sistema tenemos:

- Las debilidades en el diseño de protocolos que se utilizan en las redes
- Errores de programación
- Mala configuración de los sistemas
- Políticas de seguridad que son deficientes, o que son inexistentes
- Falta de conocimiento y sensibilización por parte de usuarios y responsables de TI
- La disponibilidad de herramientas que pueden facilitar el ataque a las vulnerabilidades de los sistemas
- Limitaciones en el tamaño de claves por parte del gobierno (en algunos países como USA y los estados miembros de la Unión Europea)
- La existencia de puertas traseras en los sistemas
- Descuidos de fabricantes
- Factor Humano

(Vieites, 2007)

Como podemos observar una de las principales vulnerabilidades de los sistemas es los errores de programación, estas vulnerabilidades son las que provocan la mayoría de los problemas de seguridad en aplicaciones web seguras, debido que los atacantes usan estas vulnerabilidades para su beneficio.

OWASP “Open Web Application Security Project”, es una organización que tiene como objetivo en análisis de vulnerabilidades, así como también el de proveer herramientas y documentos relacionados con el campo de seguridad de aplicaciones web (“OWASP,” 2018). Esta organización se encarga de publicar y actualizar el top 10 de la vulnerabilidad y riesgos en una aplicación web. Entre las cuales tenemos:

| OWASP Top 10 2013 | ± | OWASP Top 10 2017 |
|---|---|---|
| A1 – Inyección | → | A1:2017 – Inyección |
| A2 – Pérdida de Autenticación y Gestión de Sesiones | → | A2:2017 – Pérdida de Autenticación y Gestión de Sesiones |
| A3 – Secuencia de Comandos en Sitios Cruzados (XSS) | ↘ | A3:2017 – Exposición de Datos Sensibles |
| A4 – Referencia Directa Insegura a Objetos [Unido+A7] | ∪ | A4:2017 – Entidad Externa de XML (XXE) [NUEVO] |
| A5 – Configuración de Seguridad Incorrecta | ↘ | A5:2017 – Pérdida de Control de Acceso [Unido] |
| A6 – Exposición de Datos Sensibles | ↗ | A6:2017 – Configuración de Seguridad Incorrecta |
| A7 – Ausencia de Control de Acceso a las Funciones [Unido+A4] | ∪ | A7:2017 – Secuencia de Comandos en Sitios Cruzados (XSS) |
| A8 – Falsificación de Peticiones en Sitios Cruzados (CSRF) | ☒ | A8:2017 – Deserialización Insegura [NUEVO, Comunidad] |
| A9 – Uso de Componentes con Vulnerabilidades Conocidas | → | A9:2017 – Uso de Componentes con Vulnerabilidades Conocidas |
| A10 – Redirecciones y reenvíos no validados | ☒ | A10:2017 – Registro y Monitoreo Insuficientes [NUEVO, Comunidad] |

***Ilustración 19: Top 10 2017 Riesgos en Seguridad de Aplicaciones***

***Fuente:*** *(OWASP Foundation, 2017)*

- **Inyección**: Por medio de comandos o consultas (SQL, NoSQL, OS o LDAP) enviadas por el atacante, se logra engañar al interprete, lo que provoca que este ejecute comandos o que pueda acceder a datos sin la debida autorización. (OWASP Foundation, 2017)

- **Perdida de Autentificación:** La implementación incorrecta de la autentificación y gestión de secciones en una aplicación, permitirá al atacante comprometer tanto a usuarios y contraseñas, token de sesión, lo que causará que este pueda asumir la identidad de los usuarios ya sea temporal o permanentemente. (OWASP Foundation, 2017)

- **Exposición de datos sensibles:** La no implementación de métodos de protección adicional en datos sensibles (Información financiera, de salud, etc.) en las aplicaciones, permitirá el robo o su modificación. Lo que causará que el atacante pueda cometer fraudes, robo de identidad, etc. (OWASP Foundation, 2017)

- **Entidades Externas XML (XXE):** Entidades externas a las cuales son referencias de XML antiguos o mal configurados, pueden ser utilizadas para revelar información tanto de archivos internos mediante URI como de archivos en servidores que no están actualizados, también permitirá el escaneo de puertos, la ejecución de código remotamente y ataques de denegación de servicios. (OWASP Foundation, 2017)

- **Pérdida de control de Acceso:** La no restricción de los usuarios autenticados correctamente, permite que el atacante pueda acceder de forma no autorizada a la información como: cuentas de otros usuarios, archivos sensibles, como también le permite la modificación de estos, cambiar derechos de accesos y sus permisos, etc. (OWASP Foundation, 2017)

- **Configuración de seguridad incorrecta:** Esta puede ser provocada debido a que muchas veces se realiza la configuración de forma manual o por omisión, como, por ejemplo: cabeceras HTTP mal configuradas, mensajes de error que poseen contenido sensible, la no colocación de parches y sus actualizaciones, Frameworks, etc. (OWASP Foundation, 2017)

- **Secuencia de comandos en sitios Cruzados (XSS):** Son causados cuando una aplicación toma información que no es confiable y la envía al navegador web sin validación ni la debida configuración, esto permitirá que se pueda ejecutar comandos en el navegador del usuario, provocando que el atacante pueda modificar el sitio web o redireccionar al usuario a un sitio malicioso. (OWASP Foundation, 2017)

- **Deserialización Insegura:** Son causados debido a que las aplicaciones reciben objetos serializados dañinos, los cuales el atacante los puede borrar o manipular, permitiéndole realizar ataques de repetición, inyección o elevar privilegios de ejecución. (OWASP Foundation, 2017)

- **Componentes con vulnerabilidades conocidas:** Frameworks, bibliotecas u otros módulos se ejecutan con los mismos privilegios de la aplicación, su explotación provoca la pérdida de datos o en el peor de los casos que el atacante pueda tomar el control del servidor. (OWASP Foundation, 2017)

- **Registro y monitoreo Insuficientes: La** falta de monitoreo y la de respuesta a un incidente, permite al atacante mantener el ataque en el tiempo, como también pivotear los sistemas y manipularlos, extraer información y distribuirlos. (OWASP Foundation, 2017)

Como se presenta en la Ilustración 19 desde el 2013 al 2017 el mayor riesgo para aplicaciones web es la Inyección seguido de la Pérdida de Autenticación y Gestión de Sesiones. En el 2017 se presentan nuevos riesgos como: Entidades Externa de XML (XXE), la deserialización insegura y el registro y monitoreo Insuficiente.

Cuando un sistema posee vulnerabilidades como las antes mencionadas, puede provocar que el atacante pueda acceder a este con mayor facilidad a este, causando daños a la empresa o usuario, tales como: el robar la información de la empresa, como también la modificación de esta, robo de identidad y de datos personales de los usuarios, espionaje, extorsión y fraude, estas entre otras opciones que el atacante puede realizar al obtener acceso a un sistema. Es por eso por lo que es importante al desarrollar un sistema realizar las pruebas debidas para que este posee la debida seguridad y sea confiable para el usuario.

A nivel de la capa de aplicación posee algunas deficiencias de seguridad debido a la gran variedad de protocolos que maneja. Entre los principales tenemos:

- DNS: La principal vulnerabilidad que presenta este protocolo es que un atacante podrá acceder o modificar los datos almacenados en el servidor DNS, es decir podrá modificar las peticiones, como también acceder a ellas para poder obtener información sobre las cuentas del usuario, de igual

manera podrá obtener información relacionada a la topología de la red de la empresa. (Riffo, 2008)

- Telnet: Este protocolo se encarga de la autentificación de usuario, es vulnerable ante tecinas de sinfín, las cuales permiten la captura de los datos transmitidos. (Riffo, 2008)
- FTP: Este protocolo posee la misma deficiencia de seguridad que posee Telnet. (Riffo, 2008)
- HTTP: Este protocolo es uno de los más usados, la principal vulnerabilidad vista en ese protocolo es errores de programación en el código que se ejecuta remotamente para dar formato tanto en la entra como la recepción de la información. (Riffo, 2008)

#### 1.4.1.2. Principales ataques y amenazas a servicios web

En el desarrollo de sistemas web, muchas veces se realiza sin que este posea código seguro, lo que puede causar que el sistema presente vulnerabilidades que pueden ser explotadas por los atacantes, como también fallos en el software. Durante el transcurso de la historia se han presentado varias situaciones en las que reflejan lo antes mencionado, como por ejemplo: en misiones espaciales, un error de programación en el cohete Ariane 5 provoco su autodestrucción en menos de un minuto luego de su despegue, en 1996 (Enrique Jaziel Ulloa Castillón, n.d.). En la medicina un error en el código de la maquina Therac 25 que se encargaba del control de radiación a la que se exponían los pacientes en sesiones de radioterapia, entre 1985 y 1987, causo un exceso de radiación en 6 personas (Nancy G. Leveson, 1993). Estos y entre otros problemas que han surgido a lo largo del tiempo, causados por programadores que no realizan un código seguro.

El número de incidentes ha variado según el paso de los años, CERT presenta un informe con las estadísticas con la evaluación de los incidentes desde el 2004 hasta el 2017. (“Estadísticas | UNAM-CERT,” 2017)

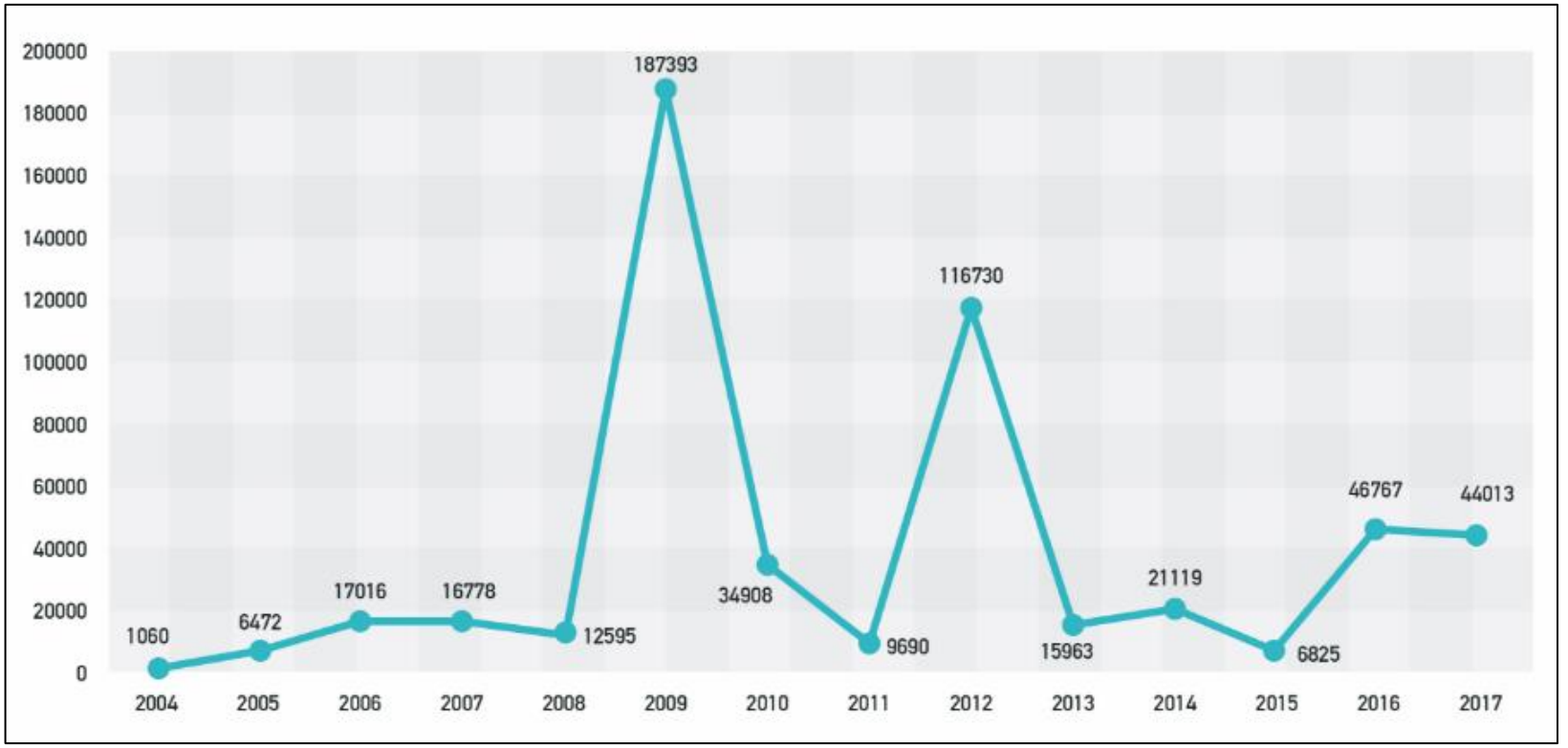
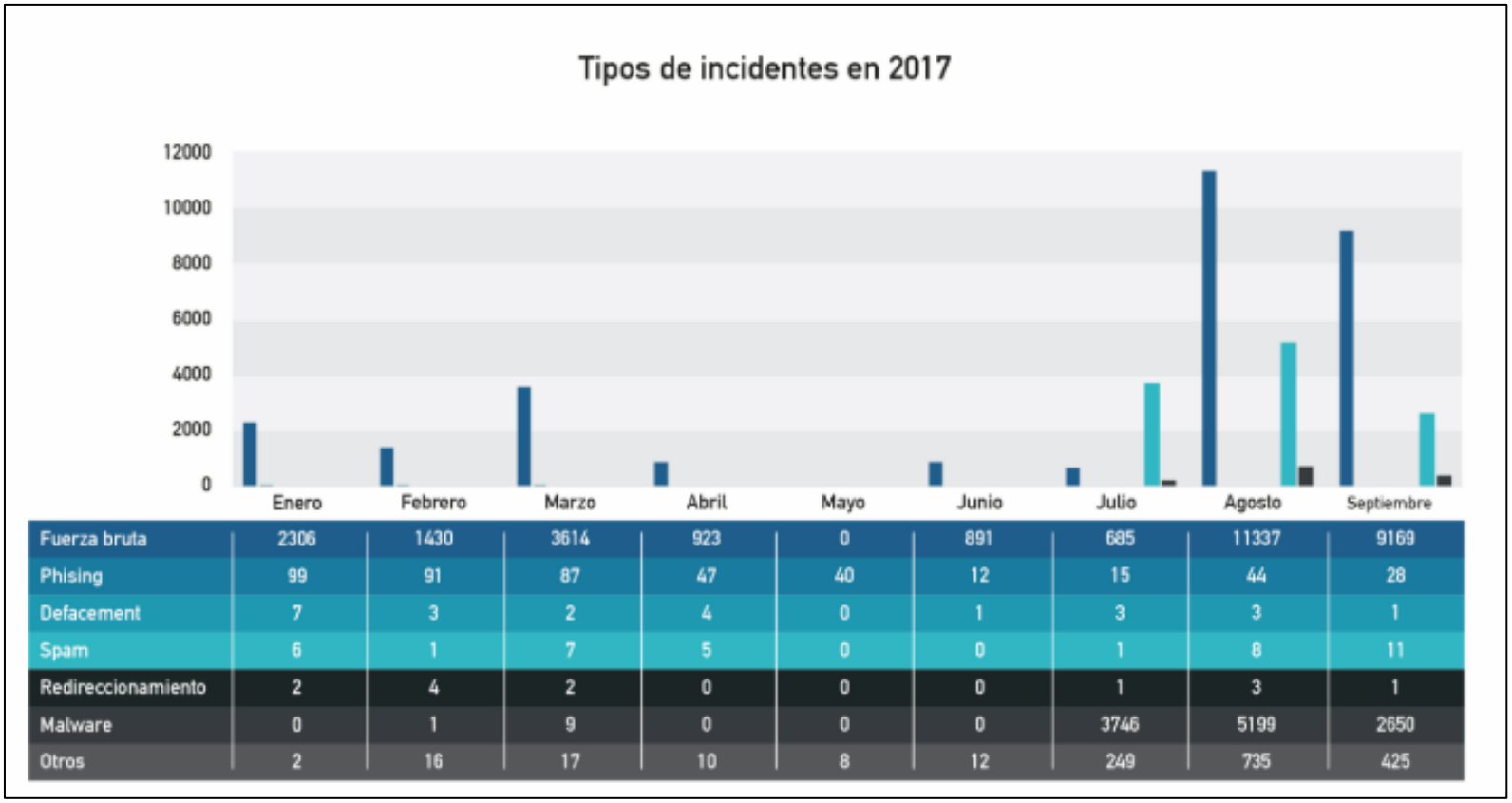



***Ilustración 20: Incidentes hasta septiembre 2017***

***Fuente:*** *("Estadísticas | UNAM-CERT," 2017)*

El mismo informe presenta los tipos de incidentes en el año del 2017, el mayor número de ataques fueron los ataques de fuerza bruta




| | Enero | Febrero | Marzo | Abril | Mayo | Junio | Julio | Agosto | Septiembre |
|---|---|---|---|---|---|---|---|---|---|
| Fuerza bruta | 2306 | 1430 | 3614 | 923 | 0 | 891 | 685 | 11337 | 9169 |
| Phising | 99 | 91 | 87 | 47 | 40 | 12 | 15 | 44 | 28 |
| Defacement | 7 | 3 | 2 | 4 | 0 | 1 | 3 | 3 | 1 |
| Spam | 6 | 1 | 7 | 5 | 0 | 0 | 1 | 8 | 11 |
| Redireccionamiento | 2 | 4 | 2 | 0 | 0 | 0 | 1 | 3 | 1 |
| Malware | 0 | 1 | 9 | 0 | 0 | 0 | 3746 | 5199 | 2650 |
| Otros | 2 | 16 | 17 | 10 | 8 | 12 | 249 | 735 | 425 |

***Ilustración 21: Incidentes hasta septiembre 2017***

***Fuente:*** *("Estadísticas | UNAM-CERT," 2017)*

***Nota: Otros "incidentes agrupados por su baja ocurrencia ". Bots, scam, SQL injection, guanos, XSS y web shells. Agrupados por su baja ocurrencia". Defacement, redireccionamientos, XSS, SQLi, phishing y Web Shell.***

Incibe define a una amenaza como una acción que se encarga de aprovechar las vulnerabilidades que tiene el sistema web y así poder atentar a la seguridad de este. (Incibe, 2017). Provocando pérdidas ya sean materiales o financieras a la empresa. Las amenazas se pueden dividir en diferentes categorías tales como:

- **Amenazas Humanas**: En esta categoría entran toda persona ya se interna o externa, que pueda comprometer la seguridad del sistema, tales como: ex – empleados, curiosos, hacker, cracker, etc. (Hackers, 2014)

- **Amenazas Lógicas**: En esta categoría se encuentran todo tipo de programa que pueda afectar al sistema, tales como: software malicioso, o explotas (utilizados para aprovechar de los errores de programación o blogs), herramientas de seguridad como NESSUS, SAINTA o SATAN, son herramientas que el atacante puede aprovechar para encontrar vulnerabilidades del sistema, otra de las amenazas lógicas son las puertas traseras, canales cubiertos u ocultos son canales de comunicación que transmiten información violando las políticas de seguridad que posee el sistema, en esta categoría también se puede encontrar amenazas como virus, gusanos, troyanos, etc. (Hackers, 2014)

- **Amenazas Físicas**: Son amenazas que puedan afectar a la seguridad y funcionamiento del sistema: tales como Robos, fraudes, cortes de energía, condiciones atmosféricas adversas y catástrofes naturales. (Hackers, 2014)

Si no tomamos las medidas de seguridad requeridas cuando desarrollamos un sistema o aplicación web, podría causar que el sistema se fácil de atacar debido a sus vulnerabilidades. Existen diferentes tipos de amenazas que pueden realizar a nivel de aplicación tales como: Validación de datos de

entrada, ataques de autentificación y autorización, gestión de configuración, ataques de criptografía, etc.

ISACA (Asociación de Auditoria y Control de Sistemas de Información) conjunto con RSA Conference, en el 2015 realizó un estudio sobre el estado actual de la ciberseguridad. Unos de los resultados que mostro este estudio fueron:

- Como principales actores que explotan las vulnerabilidades de los sistemas se encuentran los Cibercriminales los principales actores con 31%, y como segundo lugar se encuentran los Hackers con Hackers con 25%.

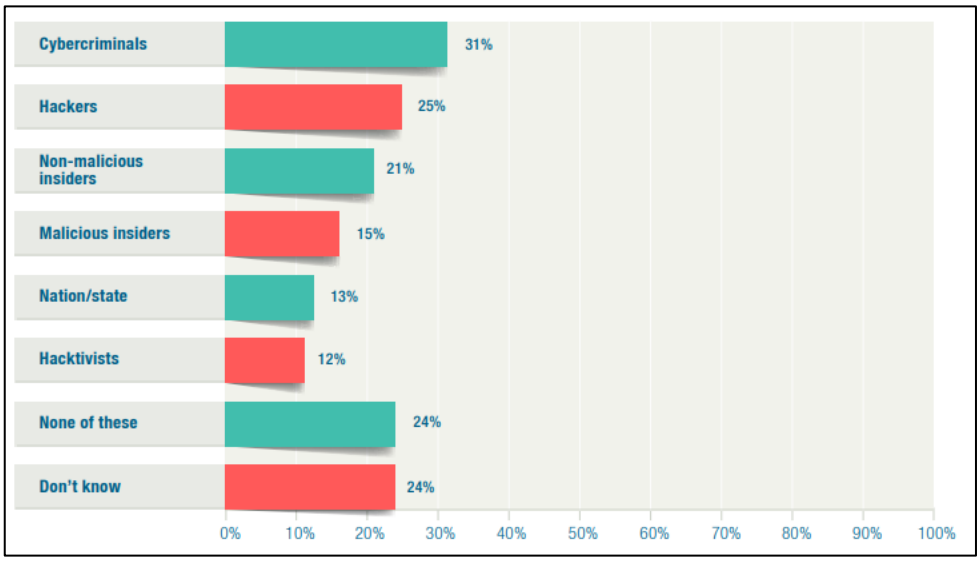



***Ilustración 22: Actores que explotan vulnerabilidades***

***Fuente:*** *(Isaca & Survey, 2016)*

1. Phishing es el principal ataque que explota las vulnerabilidades del sistema, con un 60 %, seguido del Malware con un 52%.

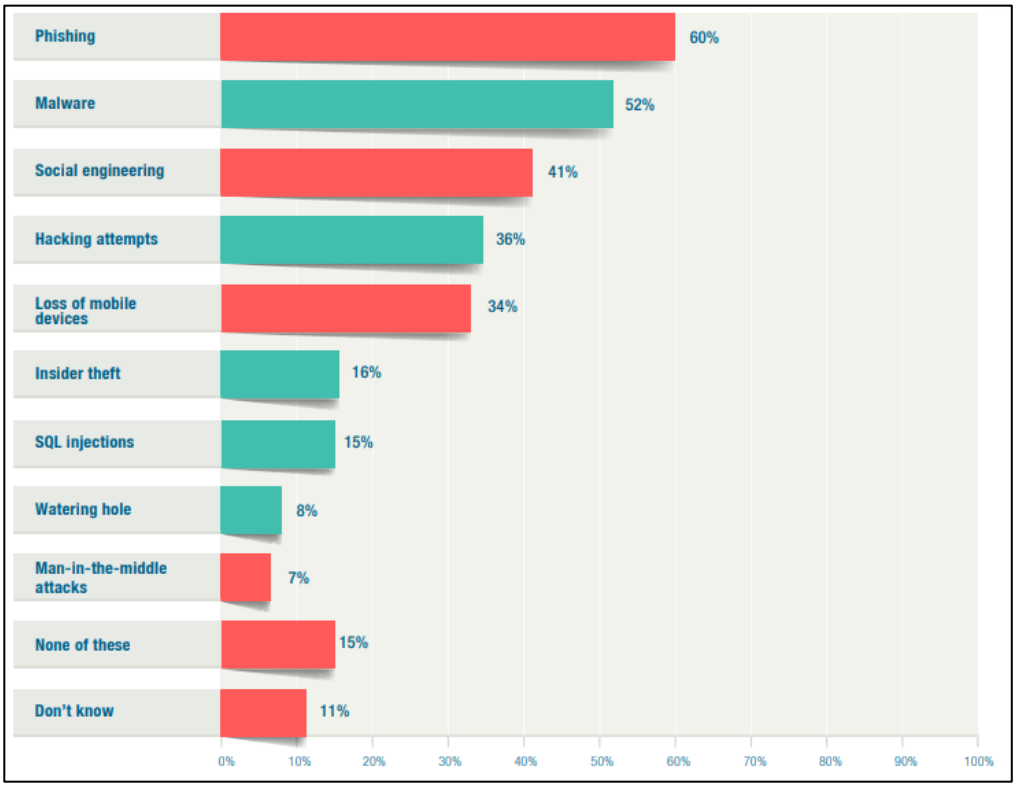



***Ilustración 23: Principales ataques***

***Fuente:*** *(Isaca & Survey, 2016)*

2. La encuentra también muestra que la mayor notificación para realizar ataques es por ganancias financieras.

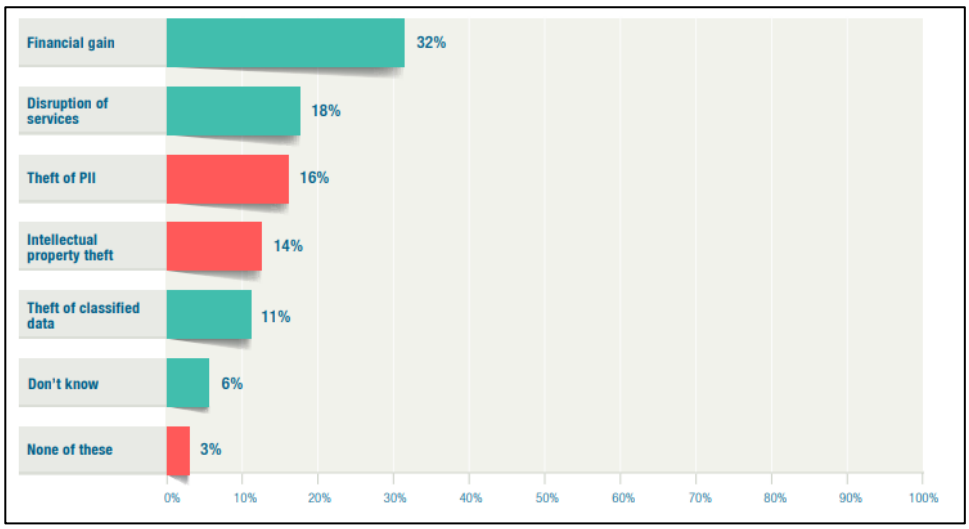



***Ilustración 24: Motivos de ataques***

***Fuente:*** *(Isaca & Survey, 2016)*

En el 2017, Cisco realizó un reporte anual de ciberseguridad, en el cual podemos observar el incremento de ataques en la capa de aplicación comparando a los ataques en la capa de red, estos ataques están dirigidos en su mayoría al protocolo HTTP (37%), protocolo de comunicaciones más utilizado en esta capa.

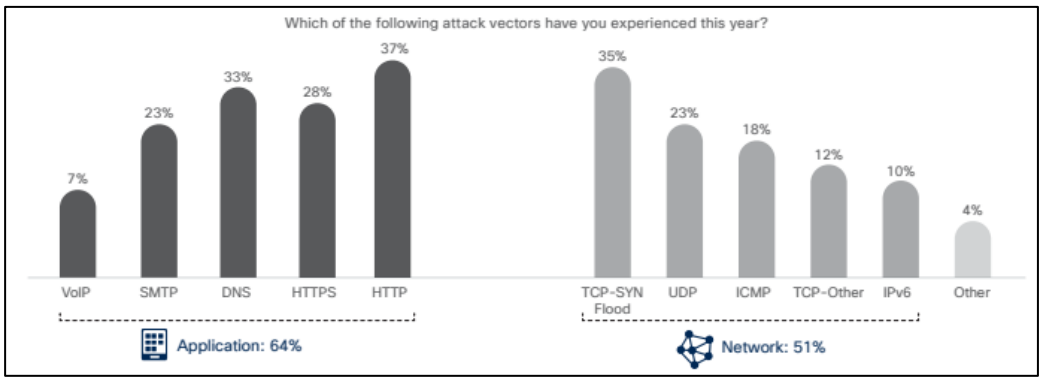



***Ilustración 25: Ataques a la capa de aplicación y red***

***Fuente:*** *(Ii & Cisco, 2018)*

El estudio realizado por Kaspersky indica que en el 2017 hubo un incremento del 16% de organizaciones con respecto al 2016 en ataques DDos, en el 2017 el 33% de estas empresas fueron víctimas de este ataque. CERT muestra que el 40% de los ataques web son dirigíos para la obtención de información, de los cuales el sector público y financiero son los que han recibido más ataques en el 2017. Los ataques más frecuentes para este año según la empresa Positive Technologies son XSS y SQLi. (Nacional & Nacional, 2018)

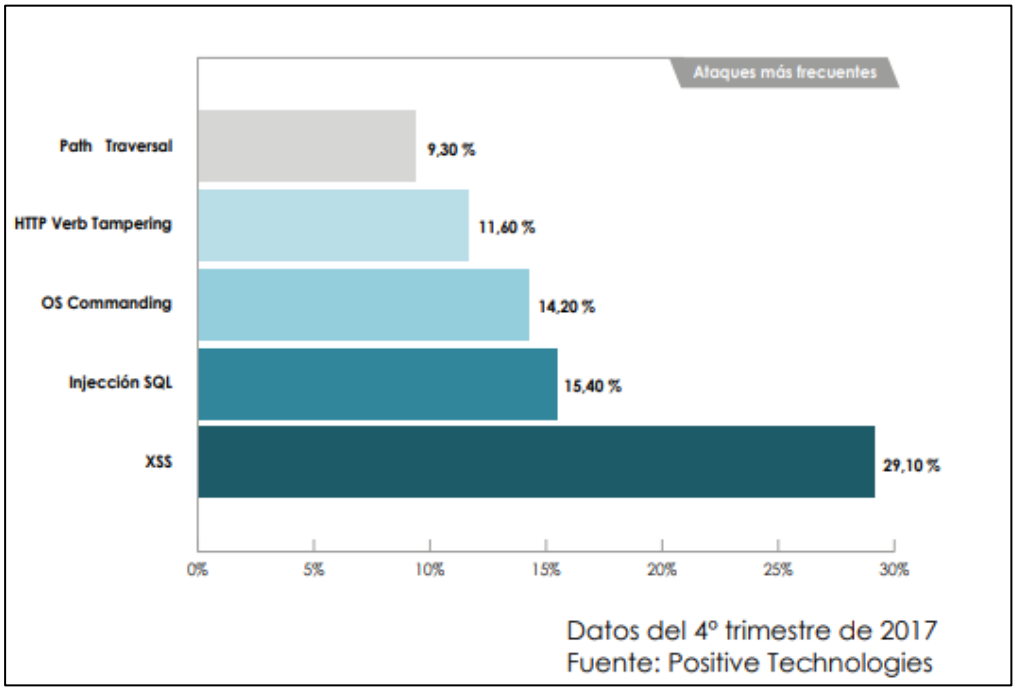



***Ilustración 26: Ataques más Frecuentes (Positive Technologies)***

***Fuente:*** *(Nacional & Nacional, 2018)*

### 1.4.1.3. Acciones de seguridad o Contramedidas

Un ciber - atacante analiza la mejor manera para explotar las vulnerabilidades de un sistema, los errores en un sistema pueden causar graves consecuencias al usuario como pérdidas financieras de una organización. En la actualidad con el avance tecnológico se han desarrollado herramientas y protocolos que permite tomar acciones en cuanto a la seguridad de un sistema, algunas de estas herramientas permiten la evaluación de las vulnerabilidades de dicho sistema, para que tras este análisis se pueda evaluar cómo se encuentra el sistema y he implementar medidas de seguridad en caso de que existan vulnerabilidades que puedan ser explotadas por los ciber atacantes.

Cisco realiza un estudio en el 2018 en donde muestra una comparación de las capacidades de seguridad, en este estudio como principal capacidad de seguridad se encuentra Endpoint Forensics con un 19% (Ii & Cisco, 2018), esta

herramienta permite prevenir y responder ciberataques de Endpoint (FireEye, 2018).

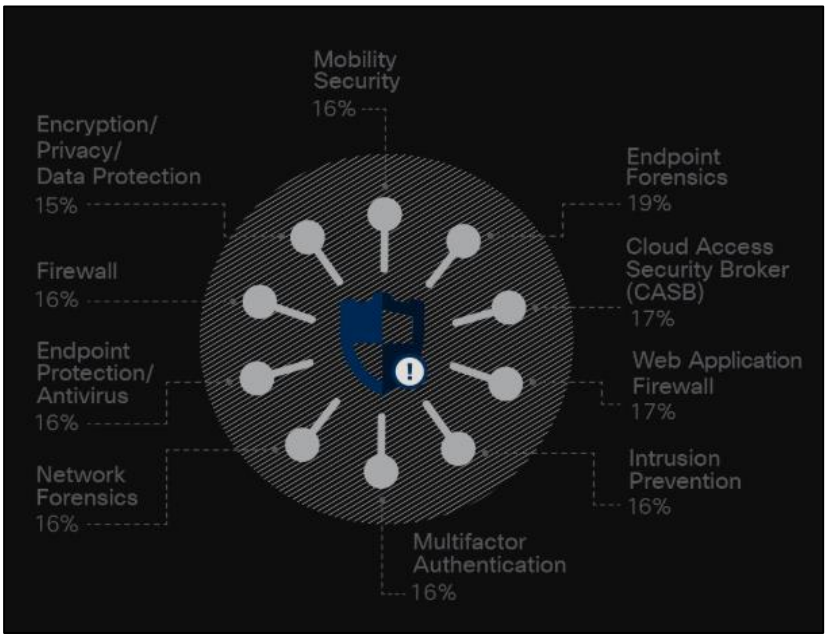



***Ilustración 27: Capacidades de Seguridad (Cisco)***

***Fuente:*** *(Ii & Cisco, 2018)*

De igual manera el estudio realizado por ISACA en el 2015, muestra las acciones que se toman para respaldar la seguridad de un sistema, como primera acción a tomar es la aplicación de políticas de seguridad (ISACA, 2015).

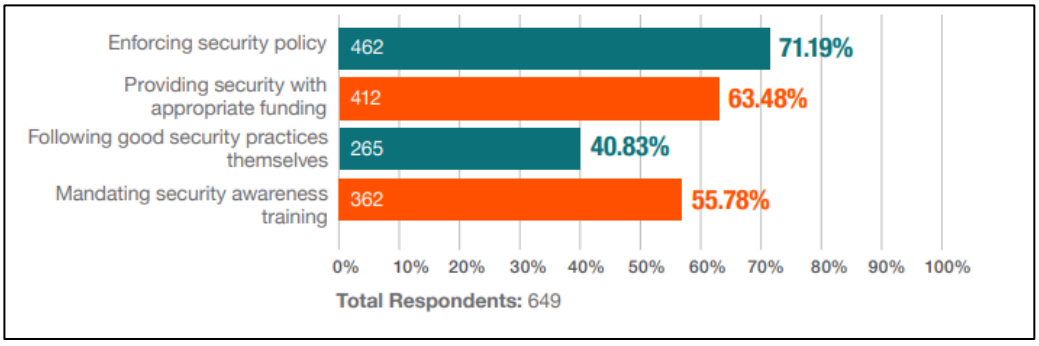



***Ilustración 28: Acciones para respaldar la seguridad***

***Fuente:*** *(ISACA, 2015)*

Las medidas que se toman en la capa de aplicación dependerán de las vulnerabilidades que posea el protocolo que usemos, uno de los protocolos de comunicación más seguros de www (World Wide Web) es el HTTP. (SSL, n.d.).

Otra de las medidas recomendadas son técnicas de criptografía que permitan la protección de la información del usuario que contiene el aplicativo, los módulos criptográficos permitirán contrarrestar las posibles fallas de filtrado de información, como también se recomienda que la información como nombres de archivos se deben generar mediante este módulo, etc. El protocolo HTTP permite la protección de la información mediante la encriptación de archivos conexiones. (Guamán Quinche, n.d.)

El uso de servicios de seguridad como autentificación, control de acceso, confidencialidad e integridad de datos, etc. De igual forma para un desarrollo seguro se recomienda también aplicar buenas prácticas de codificación basadas en seguridad, políticas de seguridad, diseño de la arquitectura y modelado de amenazas, etc. (Estándar de Verificación de Seguridad en, 2017)

# CAPÍTULO 2: METODOLOGÍAS DE DESARROLLO SEGURO

## 2.1. Estado del Arte

Se tomará como base el concepto de metodología de desarrollo de software, en donde se defiende a esta como un marco de referencia que ayuda a dar la estructura, la planificación y el control de un proyecto de desarrollo. (Nohemy, Velandia, Mesías, & López, 2015)

Estas metodologías ayudaran a mantener un proceso ordenado y disciplinado sobre el desarrollo por medio de una serie de pasos para que de esta manera se realice eficientemente y así poder cumplir con su objetivo principal, aumentar la calidad del software. Se debe tener en cuenta que una metodología no impone

que se deba cumplir rigurosamente las fases de esta, si no que estas deben ser adaptables a las características que posee el proyecto, teniendo en cuenta los recursos que se dispone para la realización de este. (Castañeda, n.d.)

Con forme al pasar del tiempo ha surgido aplicaciones Web que ofrecen diferentes tipos de servicios a los usuarios, por esta razón se ha desarrollado metodologías que permitan el modelado para el desarrollo de aplicaciones Web. En los años 70 surgen las metodologías de desarrollo de software, las cuales ofrecían soluciones a problemas que provocaban los métodos de desarrollos aplicados en ese tiempo, haciendo que el software no cumpla con las necesidades del usuario.

Estas metodologías sirven como una guía, en donde encontramos fases y procesos, que nos ayudaran a mejorar el proceso de desarrollo de aplicaciones Web, obteniendo no solo mejores resultados, sino también aumentado la calidad de esta aplicación.

Hoy en día todos realizamos uso de sistema o aplicativos que utilizan Internet, ya sea para realizar compras por medio de la red, o realizar transacciones bancarias, y uno de las más populares redes sociales. Al usar el Internet como un medio de comunicación muchas veces quedamos expuestos por las vulnerabilidades de las aplicaciones que usamos. (Adriana Matei, n.d.)

De igual forma con el desarrollo de aplicaciones web y la evolución de sus metodologías, se debe tomar en cuenta la seguridad, en la actualidad una de las mayores necesidades de una empresa es la seguridad de sus aplicaciones, esto se debe a la necesidad de resguardar su información. Con el pasar de los años no solo se ha desarrollado metodologías para desarrollo Web que mejoren la calidad de los productos sino también metodologías de desarrollo seguro, las cuales ayudarán a identificar, analizar y evaluar posibles vulnerabilidades, riesgos y amenazas para poder evitar complicaciones en la aplicación web. (Pinilla et al., n.d.)

Las metodologías de código seguro ayudaran a la prevención de defectos. Lo que beneficiará tanto en tiempo de desarrollo como en el coste de este. Debido a que aproximadamente el 60 % de defectos se presentan en la fase de diseño. (Adriana Matei, n.d.)

Pero se debe tener en cuenta que, aunque se use una metodología no se podrá garantizar el 100 % de seguridad, esto se debe que es inevitable que existas, esto también dependerá de la complejidad y del tamaño del software que se desarrolla, la metodología de código seguro ayudará a la reducción de estos errores. Un software creado teniendo en cuenta la seguridad tendrá como características: La confianza de que este funcione como debe, la confiabilidad en donde las se reduce el número de vulnerabilidades a lo mínimo. Y la conformidad en donde se garantiza que tanto los componentes, productos y sistemas cumplan con los requerimientos y estándares solicitados. (Castellaro, Romaniz, Ramos, & Pessolani, n.d.)

## 2.2. Metodologías de Desarrollo Web

La intensificación del uso de Internet con el pasar de los tiempos hasta la actualidad, ha hecho que surjan diferentes metodologías de desarrollo Web, y por medio de estas se ha logrado mejorar la aplicación obteniendo resultados que cumplen con las necesidades del usuario y por ende la mejora de la calidad. (Adriana Matei, n.d.)

Las metodologías que son más usadas son las que contienen una mayor cantidad de elementos que contiene en otras metodologías, lo que le permite tener un rango general de análisis y no concentrarse solo en algo específico, un ejemplo de estas es la metodología OOHDM, la cual cumple con las propiedades ideales para el desarrollo de aplicativos WEB. (De Evaluación & Revistas, 2017)

Existen diferentes tipos de metodologías para desarrollo Web entre las más conocidas:

| | |
|---|---|
| HDM | •HYPERTEXT DESIGN MODEL |
| RMM | • RELATIONSHIP MANAGEMENT METHODOLOGY |
| EORM | •ENHANCED OBJECT RELATIONSHIP METHODOLOGY |
| OOHDM | • OBJECT-ORIENTED HYPERMEDIA DESIGN MODEL |
| SOHDM | • SCENARIO-BASED OBJECT-ORIENTED HIPERMEDIA DESIGN METHODOLOGY |
| WSDM | • WEB SITE DESIGN METHOD |
| WAE | • WEB APPLICATION EXTENSION |

***Tabla 1: Metodologías de Desarrollo Web***

***Fuente: (De Evaluación & Revistas, 2017)***

## 2.3. Metodologías de Desarrollo con Código Seguro

### 2.3.1. Propiedades básicas de Software Seguro

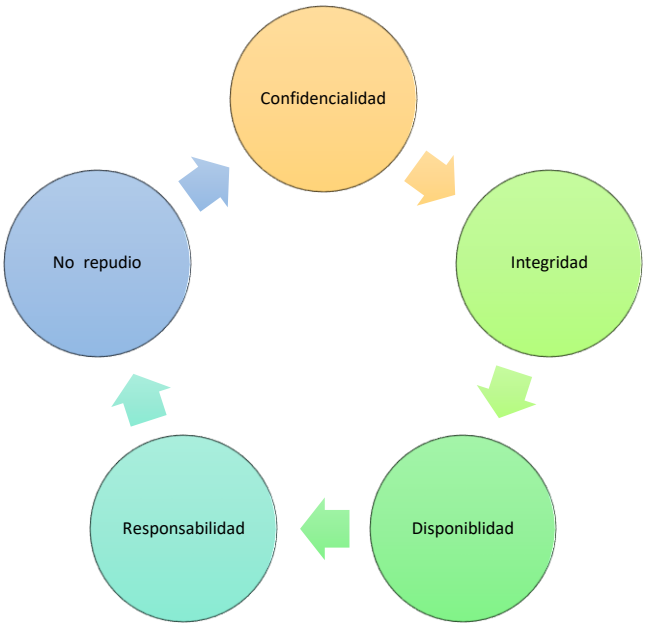



***Ilustración 29 : Propiedades de Software Seguro***

***Fuente:*** (Marulanda & Ceballos, 2012)

### a) Confidencialidad

Se asegurará que el tanto las características, activos y contenido solo estén disponibles y accesibles para entidades autorizadas e inaccesibles para usuarios no autorizados. (Adriana Matei, n.d.)

### b) Integridad

El aplicativo y sus activos deben ser flexibles a subversiones de este, es decir que su integridad se debe preservar en caso de que se realice modificaciones en el desarrollo o en la ejecución del aplicativo. (Castellaro et al., n.d.)

### c) Disponibilidad

El software debe estar operativo para los usuarios autorizados cuando se lo requiera, como también funcionar debidamente para que el usuario pueda realizar las actividades necesarias. (Adriana Matei, n.d.)

Adicionalmente a estas propiedades se encuentran también:

### d) Rendición de cuentas

También llamada Responsabilidad, en esta propiedad toda acción relevante que se encuentre relacionada con la seguridad del software debe ser registrada y también rastreada de esta manera poder decretar las responsabilidades. Esta trazabilidad se debe poder realizar tanto durante como después del registro de las acciones. (Castellaro et al., n.d.)

### e) No repudio

Propiedad en la que se evita de que un usuario niegue la responsabilidad de las acciones realzadas. (Castellaro et al., n.d.)

### 2.3.2. Propiedades conducentes

La propiedad que se presentaran no es de naturaleza segura, pero harán referencia al tamaño, complejidad y trazabilidad de este, es decir, se encuentra orientadas a las propiedades de seguridad base.




***Ilustración 30: Propiedades contundentes***

***Fuente:*** (Adriana Matei, n.d.)

#### a) Confianza

Se refiere a la fiabilidad del software, es decir, de que este opere y esté disponible correctamente sin defectos.

La relación que existe entre fiabilidad y seguridad, se encuentra en que si el software funciona con defectos y debilidades pueden generar vulnerabilidades en el software teniendo un impacto en su seguridad debido al incremento de riesgo de seguridad. (Adriana Matei, n.d.)

#### b) Exactitud

La exactitud es una propiedad importante en calidad, con referencia a la seguridad, muchas veces un software correo no es seguro, por lo que atacantes aprovechan las correcciones realizadas en el aplicativo para poder violarlo. Sin embargo, la Ingeniera de correcciones reduce en número de vulnerabilidades y defectos que pueden ser explotados, se debe tomar en cuenta que esta ingeniera no se encargará de la eliminación

de vulnerabilidades complejas que son causadas por la interacción de componentes que son individuales. (Adriana Matei, n.d.)

### c) Predecible

Tanto funcionalidades y propiedades de software deben funcionar como se espera, de igual forma el ambiente en el que opera. Referente a la seguridad el software debe operar correctamente en caso de que se presenten condiciones imprevistas, es decir que se presenten vulnerabilidades y el software es atacado (Castellaro et al., n.d.), es decir, que para garantizar que un software sea predecible en condiciones impredecibles, se debe tener vulnerabilidades al mínimo.

### d) Confiabilidad

Un software que es confiable es un software que es tolerante a fallos (Adriana Matei, n.d.), es decir, que el correcto funcionamiento del software inclusive cuando existe defectos, debilidades o cambios impredecibles en el ambiente que opera. (Castellaro et al., n.d.)

### e) Seguridad

Depende de la propiedad anterior, en caso de que el aplicativo sea violado, una de las soluciones más optimas es detener su funcionamiento o a su vez quedar parcialmente operativo con un estado seguro. (Castellaro et al., n.d.)

En la seguridad del software también interviene lo que es el tamaño, complejidad y su trazabilidad. Debido a que entre menor número de dependencias que exista en el software, será menos complejo y ayudará a la detección de fallos y reducción de vulnerabilidades. En cuanto a la trazabilidad permitirá asegurar que el diseño del software es un diseño

seguro que cumple con los requerimientos de seguridad. (Adriana Matei, n.d.)

### 2.3.3. Metodologías

Metodologías de desarrollo tradicional como modelo de cascada o de desarrollo ágil como SCRUM no están enfocadas a la seguridad de la aplicación, es por eso que nos enfocaremos en metodologías que tiene como uno de sus principales objetivos el desarrollo seguro, tales como:

| METODOLOGÍA DE DESARROLLO SEGURO | EMPRESA | METODOLOGÍA |
|---|---|---|
| Microsoft Security Development Lifecycle | Microsoft | Tradicional |
| Oracle Software Security Assurance | Oracle | Tradicional |
| Comprehensive Lightweight Application Security Process | OWASP | Tradicional |
| Team Software Process Secure | Software Engineer Institute | Tradicional |
| Software Assurance Maturity Model | OWASP | Tradicional |
| Building Security In Maturity Model | Cigital | Ágil |
| Agile Development Using Microsoft Security Development Lifecycle | Microsoft | Ágil |

***Ilustración 31: Metodologías de Desarrollo seguro***

***Fuente:*** (Caro, n.d.)

De las cuales se tomará para el análisis las metodologías: SDL “Microsoft”, SAMM “OWASP” como metodologías tradicionales.

#### 2.3.3.1. OWASP

OWASP “Open Web Application Security Project”, es una organización que tiene como objetivo en análisis de vulnerabilidades, así como también el de proveer herramientas y documentos relacionados con el campo de seguridad de aplicaciones web (“OWASP,” 2018).

Esta organización se encarga de publicar y actualizar el top 10 de la vulnerabilidad y riesgos en una aplicación web. Entre las cuales tenemos:

Como se había mencionado anteriormente OWASP "Open Web Application Security Project", es una organización que se centra en la mejora de la seguridad de aplicativos("OWASP," 2018). Fue creada el 01 de diciembre, 2001. Esta fundación cuenta con diferentes proyectos y que se actualizan periódicamente, entre uno de ellos es Guías OWASP, las cuales son una base para la seguridad de aplicaciones WEB.

- Referencia de Escritorio en Seguridad de Aplicaciones de OWASP (OWASP Application Security Desk Reference)
- Guía de Desarrollo
- Guía de Pruebas
- Guía de Revisión de Código

La guía que se va a tomar como referencia es la Guía de Desarrollo" Una Guía para Construir Aplicaciones y Servicios Web Seguros", esta guía cuenta con controles de seguridad que s e debe tomar en cuenta para el desarrollo de software. De esta forma poder evitar en su mayoría las vulnerabilidades por medio de controles de seguridad. (GUÍA DE PRUEBAS OWASP 2008 V3.0, 2002)

Esta guía utiliza ejemplos J2EE, ASP.NET y PHO. Proporciona información sobre la Arquitectura y diseño de Seguridad, entre los principales temas que plantea son:

- **Marcos de políticas:** En donde se requiere de: una gestión organizacional que este enfocada en la seguridad, políticas de seguridad documentadas y con estándares nacionales, una metodología de desarrollo y una gestión segura en versiones. (*GUÍA DE PRUEBAS OWASP 2008 V3.0*, 2002)

- **Modelo de riesgos de Amenaza:** OWASP usa el proceso de modelamiento de amenaza realizado por Microsoft, el cual cuenta con 5 pasos (*GUÍA DE PRUEBAS OWASP 2008 V3.0*, 2002):

- Realización del flujo del Modelo de Amenazas

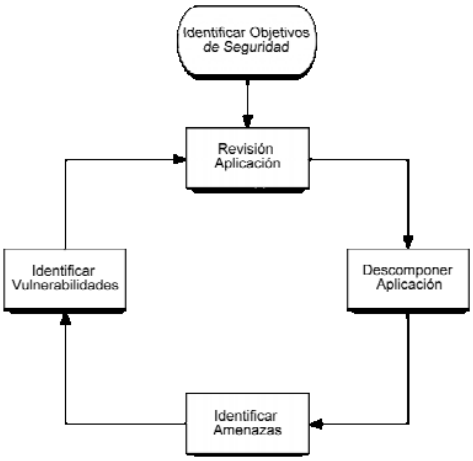



***Ilustración 32: Modelo de Amenazas***

***Fuente:*** *(GUÍA DE PRUEBAS OWASP 2008 V3.0, 2002)*

- Identificación de objetivos de Seguridad
- Visión General de la Aplicación: Análisis de la aplicación
- Descomposición de la aplicación
- Documentación de amenazas

Adicionalmente temas como phishing, servicios web, validación de datos, etc.

De igual forma OWASP provee una guía de buenas prácticas de codificación con el objetivo que se tome en cuenta para el proceso de desarrollo de la aplicación web. (“Modelo de seguridad y Auditoria para aplicaciones web basado en metología OWASP,” 2016)

En esta guía nos da una orientación sobre:

- Validación de entradas
- Codificación de salidas
- Administración de autenticación y contraseñas
- Administración de sesiones

- Control de Acceso
- Prácticas Criptográficas
- Manejo de Errores y Log
- Protección de datos
- Seguridad de Comunicaciones
- Configuración de los sistemas
- Seguridad de Base de Datos
- Manejo de Archivos
- Manejo de Memoria
- Practicas Generales para la Codificación

(Foundation, Commons, & Sharealike, 2010)

#### 2.3.3.1.1. SAMM

##### 2.3.3.1.1.1. Definición

Se lo define como modelo de madurez, ayuda a empresa a establecer estrategias de seguridad para la aplicación Web, es un modelo flexible por lo que puede ser usado tanto en empresas pequeñas y adaptables a los riesgos que presenta la empresa. (Ericka Jamin Robles Gomez, 2011)

Esta metodología provee los siguientes recursos a una organización:

- Evaluación de prácticas de desarrollo de software seguro
- Construcción de un Software seguro que sea equilibrado.
- Mejoras en la aplicación que garantice la seguridad.
- Definición y medición de actividades de seguridad.

(Deleersnyder et al., 2017)

Esta metodología se sujeta a los siguientes principios:

- **El comportamiento de la empresa presenta cambios con el tiempo:** Una aplicación segura deberá ser implementada en iteraciones pequeñas, los cuales tengan resultados tangibles y que trabaje hacia objetivos a largo plazo. (Payne, Watson, Williams, Mcgovern, & Steingruebl, 2016)

- El marco de seguridad en Software debe ser flexible permitiendo la modificación de decisiones basándose en su tolerancia al riesgo y la manera en la que se lo construye. (Payne et al., 2016)

- Los lineamientos, tales como proceso de construcción y medición de la aplicación deben ser específicos, bien definidos y medibles. (Payne et al., 2016)

#### 2.3.3.1.1.2. Modelo

Este modelo presenta la siguiente modelo

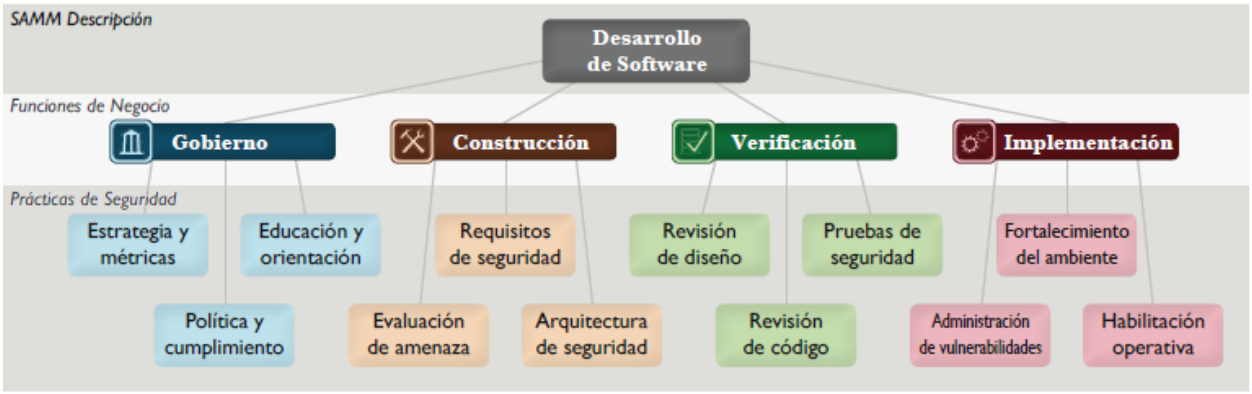


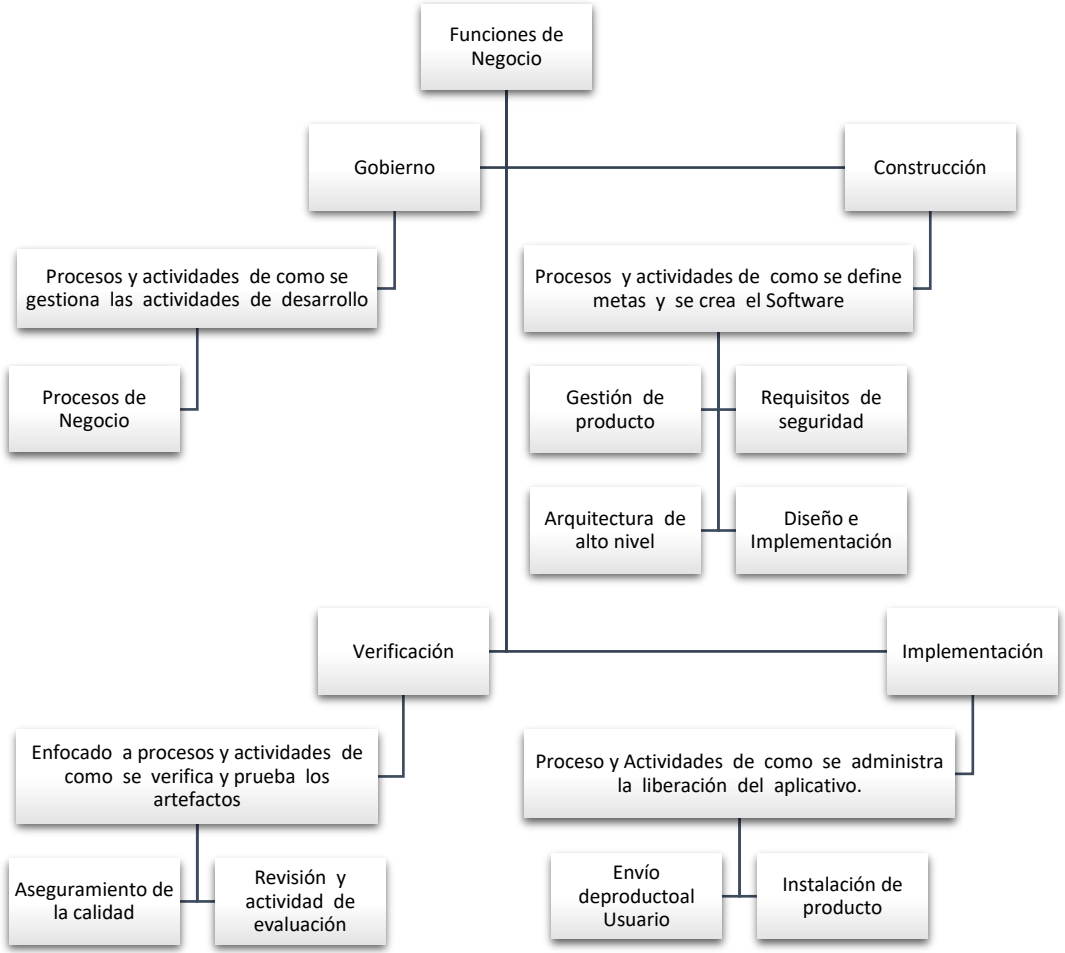

***Ilustración 33: Modelo de desarrollo seguro de SAMM***

***Fuente:*** (Deleersnyder et al., 2017)

| Nivel | Descripción |
|---|---|
| **Funciones de Negocio** | • Nivel más alto<br>• Posee actividades que hacen referencia a las tareas de desarrollo de la aplicación<br>• Consta de 4 actividades |
| **Prácticas de Seguridad** | • Las prácticas de seguridad son áreas conformadas por actividades de seguridad<br>• Construye actividades para funciones de negocio<br>• Está compuesta por 12 prácticas<br>• Tiene como objetivo 3 niveles de madurez |

**Tabla 2: Niveles de Modelo de Desarrollo seguro SAMM**
***Fuente:*** (Deleersnyder et al., 2017)

### a) Funciones de Negocio




***Fuente:*** (Deleersnyder et al., 2017)

#### 2.3.3.2. Microsoft

##### 2.3.3.2.1. SDL

###### 2.3.3.2.1.1. Definición

SDL (Ciclo de Vida de Desarrollo de Software Seguro), metodología desarrollada por Microsoft en 2004, enfocada al desarrollo de software enfocado a la seguridad, teniendo como objetivo la reducción de costos y el aumento de la calidad. (Microsoft, 2015)

Este modelo se base en un conjunto de requisitos de seguridad que ayudaran al desarrollo de aplicaciones más seguras, reduciendo el número de vulnerabilidades y por ende el riesgo. (Microsoft, 2018b)

SDL, aborda temas como la comprensión de elementos que pertenecen al proceso de desarrollo de seguridad, como también al establecimiento de prioridades en el desarrollo. ("Simplified Implementation of the Microsoft SDL," 2016)

Las actividades realizadas en el proceso están organizadas en etapas, las cuales tienen como base la seguridad. (İlgün, n.d.). Al ser un proceso que pertenece a Microsoft no es accesible, por lo cual solo se encuentra documentos en los que se detalla conceptos generales y sus fases. Aunque los ciclos de vida de desarrollo la hacen una metodología flexible, lo que hace que pueda ser implementada a cualquier tipo de empresa, el no ser accesible impide la obtención de herramientas y recursos ya que solo están disponibles internamente para Microsoft. (Alberto, García, Perla, & Velasco Elizondo, 2010)

#### 2.3.3.2.1.2. Modelo

SDL posee un modelo de optimización, para el cual su estructura se basa en cinco áreas, con 4 niveles de Madurez:




**Ilustración 34: Niveles de Madurez SDL**
***Fuente:*** ("Simplified Implementation of the Microsoft SDL," 2016)

Parte desde el nivel de madurez básico, según vaya avanzando en el nivel de madurez hasta el nivel dinámico donde se encontrará con recursos definidos, tales como procesos eficaces, herramientas, etc. Este modelo de optimización tiene como objetivo la mejora en los procesos de desarrollo. (İlgün, n.d.)

Esta metodología cuenta con 7 actividades para el aseguramiento de la Seguridad, las cuales son actividades de seguridad que se realizan en un orden especifico y que al ser realizadas como un proceso de desarrollo de software se obtendrá mayores beneficios. (Microsoft, 2018b)




**Ilustración 35: Fases del modelo SDL**
***Fuente:*** (Alberto et al., 2010)

Siendo las 5 fases principales

- Requisitos
- Diseño
- Implementación
- Verificación
- Lanzamiento y Respuesta

(Graham, 2013)

Las cuales se basan en un conjunto de prácticas que abogan por los requisitos de seguridad y su cumplimento, la creación de software seguro permite la reducción en número y gravedad de vulnerabilidades, como también los costos que se generan en el desarrollo. (Microsoft, 2018b)

| Característica | Descripción |
|---|---|
| **Entrenamiento** | • Comprensión de conceptos básicos de seguridad y su integración en el software<br>• Conocer los servicios que ayuden a realizar aplicativos seguros que satisfagan las necesidades comerciales<br>• Capacitaciones que refuercen políticas de seguridad y prácticas de SDL<br>(Microsoft, 2018b) |
| **Requisitos de Seguridad** | • Considera la seguridad y la privacidad en las aplicaciones, en donde los requisitos de seguridad se actualizarán continuamente, los factores que afectan a los requisitos son:<br>o Requisitos legales<br>o Requisitos Industriales<br>o Estándares Internos<br>o Practica de codificación<br>o Incidencias anteriores<br>o Amenazas conocidas<br>(Microsoft, 2018b) |
| **Métricas y reportes de Cumplimiento** | • Definir niveles de calidad de seguridad y responsabilidad<br>• Establecimiento de barra de errores, que permita identificar los umbrales de severidad en las vulnerabilidades. Los cuales se pueden basar en la barra de errores de seguridad que posee SDL |
| **Modelado de amenazas** | • Implementación de seguridad de los diseños de forma estructurada |

| | |
|---|---|
| | • Identificación de vulnerabilidades de forma efectiva reduciendo costos<br>• Identificar riesgos de las amenazas<br>• Definir funciones de seguridad, como también las mitigaciones correspondientes<br>(Microsoft, 2018b) |
| **Requisitos de diseño** | • Definir características seguras como:<br>o Criptografía<br>o Autentificación<br>o Registro<br>o Otros.<br>(Microsoft, 2018b) |
| **Estándares de Criptografía** | • Utilización de cifrado para protección de información<br>• Realización de estándares de cifrado claros especificando los elementos de su implantación<br>• Se recomienda hacer uso de bibliotecas de cifrado que fueron revisadas por la industria<br>• Se recomienda implementarlos de forma que se pueda remplazar fácilmente<br>(Microsoft, 2018b) |
| **Gestión de riesgos de seguridad (Componentes de Terceros)** | • Compresión de impacto que puede causar las vulnerabilidades de componentes de terceros<br>• Inventario de estos componentes<br>• Definir plan de mitigación de riesgos |

| | |
|---|---|
| | • Microsoft cuenta con un plan de gestión de riesgo tanto para componentes terceros y código abierto que se puede utilizar como referencia<br>(Microsoft, 2018b) |
| **Herramientas** | • Establecer lista de herramientas que se encuentren aprobadas<br>• Uso de herramientas que tengan comprobación de seguridad asociada<br>• Uso de herramientas en su última versión<br>(Microsoft, 2018b) |
| **Pruebas de Seguridad (Análisis estático)** | • Análisis de código fuente antes de realizar la compilación<br>o Revisión de código de seguridad<br>o Revisión de políticas de codificación segura<br>• SAST, identifica vulnerabilidades del software<br>• Soluciones para fallas en el desarrollo<br>(Microsoft, 2018b) |
| **Pruebas de Seguridad (Análisis dinámico)** | • Análisis del tiempo de ejecución que tiene un software<br>• Uso de herramientas, ataques, monitoreo que ayude al análisis del comportamiento del aplicativo<br>• Detección de errores como:<br>o Daños de memoria |

| | |
|---|---|
| | ○ Privilegios de usuario<br>• Herramientas:<br>○ Escaneo de la aplicación WEB<br>○ Fucsina (Técnica de Pruebas de Software)<br>(Microsoft, 2018b) |
| **Pruebas de penetración** | • Análisis de seguridad del sistema, este lo realizar personal de seguridad que se encuentran capacitado para simular ataques de piratas informáticos<br>• Detección de vulnerabilidades por:<br>○ Errores de codificación<br>○ Fallas de configuración<br>○ Debilidades operativas<br>• Revisión de código<br>(Microsoft, 2018b) |
| **Proceso estándar de respuesta a incidencias** | • Realización de un plan de respuestas de incidencias (PSIRT)<br>• Contenido del plan:<br>○ Contacto en caso de emergencia de seguridad<br>○ Protocolo de seguridad<br>○ Plan para código heredado o código de terceros |

**Tabla 3: Prácticas del modelo SDL**
***Fuente:*** (Microsoft, 2018b)

# CAPITULO 3: CUADRO COMPARATIVO Y ANÁLISIS DE LA BRECHA QUE EXISTE ENTRE LAS CARACTERÍSTICAS DE LAS METODOLOGÍAS DE DESARROLLO SEGURO.

## 3.1. Descripción

El presente capitulo tratará sobre la metodología que se usará para posteriormente realizar la elaboración de la propuesta del prototipo guía de seguridad enfocado al desarrollo seguro, en este caso la metodología usada es un cuadro de comparación de las metodologías enfocadas a desarrollo seguro, se podrá realizar la identificación de las principales características.

Se realizará un análisis de la brecha que existe en las características de las metodologías de desarrollo seguro, para esto se tomará las metodologías mencionadas en el anterior capítulo, en donde se presenta un resumen de estas y sus modelos.

Estas metodologías se seleccionaron bajo varios criterios, uno de ellos por el grado de información que proporcionan las empresas que las desarrolladoras y debido a que cubren un alto grado de actividades del ciclo de vida del desarrollo, proporcionando aplicativos con mejor calidad, seguridad y menos costo, de igual forma se tomó en cuenta que estas metodologías cumplan en su mayoría con las propiedades básicas y contundentes de un Software Seguro expuestas en anterior capitulo .

Como se mencionó anteriormente, para la realización del cuadro comparativo se tomarán diferentes características o criterios entre las principales.

### 3.1.1. Criterios Principales

- **Flexibilidad:** Establecer si el proceso es adaptable en cuanto a la tolerancia de riesgos, desarrollo y uso. (Alberto et al., 2010)

- **Factibilidad en métricas:** Establecer si por medio del proceso de seguridad se puede realizar la toma de métricas que sean útiles referentes a la seguridad del software. (Alberto et al., 2010)

- **Definición y documentación:** Disponibilidad de documentación respecto a: proceso, soporte, scripts, formatos, guías, etc. Que ayuden a la implementación de un aplicativo seguro. (Alberto et al., 2010)

- **Complejidad:** Identificar el grado de dificultad presenta el proceso de seguridad para poder comprenderlo e implementarlo. (Alberto et al., 2010)

- **Tipos de empresa que la aplica:** Identificar si la metodología se la puede aplicar en empresas pequeñas, medias o grandes. (Alberto et al., 2010)

- **Factibilidad TSPI:** Factibilidad que el proceso se incorpórea un proceso TSPI. (Alberto et al., 2010)

| Criterio\Metodología | SDL | SAMM |
|---|---|---|
| **Flexibilidad** | Alta | Alta |
| **Factibilidad en métricas** | Alta | Alta |
| **Definición y documentación** | Media | Alta |
| **Complejidad** | Alta | Media |
| **Tipos de empresa que la aplica** | Todo tipo de Empresas | Todo tipo de empresa |
| **Factibilidad TSPI** | Media | Media |

**Tabla 4: Cuadro comparativo – Criterios Principales**

#### 3.1.1.1. Análisis

Según la comparativa realizada los dos procesos son medianamente adaptables a TSPI, debido a que la estructura de metodología no es completamente adaptable

a este proceso, en cuanto a seguridad las metodologías son flexibles, ya que se pueden aplicar a cualquier tipo de empresa , como también, el proceso de seguridad que presentan son adaptables a los riesgos y vulnerabilidades que puede presentar el aplicativo, dando soluciones a la mayoría de estas, la mayor diferencia es que SAMM posee documentación libre y disponible para consulta, como también la fundación que la desarrolla OWASP, posee guías y proyectos de desarrollo seguro que se encuentran a disponibilidad a cualquier tipo de persona, a diferencia de SDL que sus información es restringida para uso interno de Microsoft, por lo tal solo se podrá realizar uso de la documentación general que se publica, lo que hace que su grado de complejidad sea mayor a SAMM.

### 3.1.2. Criterios Secundarios

- **Recursos – Persona:** Personas que están involucradas proceso del ciclo de vida de desarrollo seguro. (Ericka Jamin Robles Gomez, 2011)

- **Tiempo:** Incluyen el tiempo de adaptación e implementación, para esto se tomará en cuenta los siguiente de cada método. (Ericka Jamin Robles Gomez, 2011):
    - Prácticas
    - Actividades
    - Funciones
    - Fases
    - Procedimientos

- **Uso en Empresas:** Proporciona información de la popularidad que tiene cada metodología en empresas que utilizan desarrollo seguro. (Ericka Jamin Robles Gomez, 2011)

| Característica\Metodología | SDL | SAMM |
|---|---|---|
| **Recursos - Persona** | • Desarrolladores<br>• Asesor de seguridad/Asesor de privacidad<br>  ○ Auditor<br>  ○ Experto<br>• Líder de equipo<br>  ○ Líder de seguridad<br>  ○ Líder de privacidad | • Desarrolladores<br>• Arquitectos<br>• Administradores<br>• Testeadores de Calidad<br>• Auditores de Seguridad<br>• Dueños de Negocio<br>• Soporte u Operaciones |
| **Tiempo** | 7 etapas (5 Principales)<br>17 Prácticas (12 Principales) | 4 funciones de Negocio<br>12 Prácticas de Seguridad |
| **Uso en Empresas** | 12% | 23% |

**Tabla 5: Cuadro comparativo – Criterios Secundarios**

#### 3.1.2.1. Análisis

En base al cuadro comparativo se puede concluir que SDL es una metodología mucho más popular que SAMM, de igual forma que, aunque realiza uso de menos Recursos - Personas, SAMM posee los recursos necesarios para poder cubrir organizadamente con las fases que esta presenta. Por último, el Tiempo, tomando en cuenta las características expuesta anteriormente, se considera a SAMM como la metodología más flexible, debido a que consta de 4 funciones de Negocio y 12 Practicas, lo que hace que se requiera menos esfuerzo y tiempo.

### 3.1.3. Criterios de Seguridad

En este punto se tomarán en consideración características que debe poseer un aplicativo web para ser seguro, se debe tener en cuenta que una metodología no garantiza un 100% de seguridad en el aplicativo web, pero ayuda a la reducción de seguridad de vulnerabilidades y por ende a la reducción de riesgos o amenazas.

| Característica\Metodología | SDL | SAMM |
|---|---|---|
| Entrenamiento y conciencia de seguridad | Si | Si |
| Establecer Requerimientos de Seguridad | Si | Si |
| Análisis de trazabilidad de Requerimientos | Si | Si |
| Establecer Políticas de Seguridad | No | Si |
| Define métricas de seguridad | No | No |
| Establecer Responsabilidades de seguridad | Si | Si |
| Seguridad física | No | No |
| Seguridad lógica | Si | Si |
| Gestión segura de la configuración | No | No |
| Cumplimiento de la ley y política | Si | Si |
| Modelado de amenazas | Si | Si |

| | | |
|---|---|---|
| Análisis de riesgo | Si | Si |
| Pruebas de seguridad Análisis dinámico | Si | Si |
| Pruebas de seguridad Análisis Estático | Si | Si |
| Definición de requisitos de seguridad | Si | Si |
| Arquitectura de seguridad | Si | Si |
| Análisis de diseño | Si | Si |
| Diseño seguro | Si | Si |
| Análisis de código fuente | Si | Si |
| Análisis de vulnerabilidad | Si | Si |
| Verificación de seguridad | Si | Si |
| Gestión de vulnerabilidad | Si | Si |
| Técnicas y aplicaciones de desarrollo seguro | Si | Si |
| Seguridad del entorno operacional | Si | Si |
| Plan de incidencias | Si | Si |
| Realiza uso de herramientas certificadas | Si | Si |
| Plan de respuesta a incidencias | Si | Si |

***Tabla 6: Cuadro comparativo – Criterios de Seguridad***

#### 3.1.3.1. Análisis

Según la comparativa realizada en la Tabla 6 se tiene lo siguiente

- Características que las dos metodologías no poseen

| Característica\Metodología | SDL | SAMM |
|---|---|---|
| Define métricas de seguridad | No | No |
| Seguridad física | No | No |
| Gestión segura de la configuración | No | No |

**Tabla 7 : Características que no poseen ninguna de las dos metodologías**

Las dos metodologías no definen métricas de seguridad, SDL por ejemplo proporciona una lista de barra de errores de seguridad que se usan como base en la práctica "Definir informes de métricas y cumplimiento". De igual forma SAMM recolecta métricas históricas sobre la postura que tiene la organización sobre la seguridad.

De igual forma estas metodologías no cuentan con una seguridad física que permita controlar amenazas físicas. En cuanto a la gestión segura de la configuración "SCM" estas metodologías no cuentan con herramientas que puedan realizar las gestiones de configuración la cual permite la administración de las configuraciones del aplicativo para poder tener un control en la seguridad de este, de esta formar gestionar los riesgos que pueden provocar

- Características que posee SDL, pero no SAMM

Según el cuadro comparativo la metodología SDL no posee características que no posee SAMM

- Características que posee SAMM, pero no SDL

| Característica\Metodología | SDL | SAMM |
|---|---|---|
| Establecer Políticas de Seguridad | No | Si |

**Tabla 8: Características que solo posee SAMM**

SDL en las fases y prácticas del ciclo de vida de desarrollo no presenta el establecimiento de política de seguridad, solo se menciona el refuerzo de políticas de seguridad en la práctica de entrenamiento.

# CAPITULO 4: Propuesta

## 4.1. PROPUESTA DE UN PROTOTIPO DE GUÍA DE SEGURIDAD ENFOCADO AL DESARROLLO SEGURO, BASADO EN LAS CONTRAMEDIDAS Y METODOLOGÍAS DE SEGURIDAD A NIVEL DE LA CAPA DE APLICACIÓN

Hoy en día la tecnología se ha desarrollado en diferentes áreas de la sociedad, ayudando al desarrollo de la misma. Lamentablemente, no se ve libre de atacantes que pueden provocar daños aprovechando las vulnerabilidades que puede presentar.

Las amenazas y ataques informáticos han incrementado con el avance tecnológico, causando que tanto organizaciones como programadores apliquen contramedidas que permitan evitarlas o prevenirlas. En nuestro campo, poseemos diferentes formas de programar, muchas de ellas no aplican la seguridad pertinente para evitar el mayor grado de vulnerabilidades.

Este capítulo se presentará una propuesta de guía de seguridad, la cual está enfocada al desarrollo seguro. Para la realización de esta guía se realizó la comparación de dos metodologías: SAMM perteneciente a OWASP y de SDL que pertenece a Microsoft, en las cuales se realizó un análisis en cuanto complejidades, tiempo, accesibilidad, acciones de seguridad que presentan tales como: Criptografía, buenas prácticas de codificación enfocadas a seguridad, sistemas de autentificación, etc.

A partir de estas consideraciones, se concluye que tanto la metodología SDL y SAMM, están compuestas con un ciclo de vida que al implementarlo ayudará a la reducción de vulnerabilidades del aplicativo, al ser metodologías flexibles permitirá la implementación en cualquier tipo de empresas.

En cuanto a desarrollo seguro cumplen con la mayoría de características expuestas para la comparación, lo que quiere decir que los modelos de estas metodologías permitirán el desarrollo de una aplicación Web con menor grado de riesgo a ser atacados porque poseen menos vulnerabilidades. Sus procesos de estas metodologías se centran en prácticas como: capacitación de seguridad del personal que va interactuar, de igual forma un análisis de riesgo y amenazas que permite ver las vulnerabilidades que tiene el aplicativo tales como: Pérdida de Sección y Gestión de Sesiones, Configuración defectuosa de seguridad, Exposición de datos sensibles, pérdida de control de acceso, etc. Como también las pruebas de seguridad que se realizan, como, por ejemplo: Pruebas de Autenticación, de Autorización, Pruebas de Gestión de Sesiones, Pruebas de denegación de servicio, etc. Esto cubre las fases de análisis, diseño, implementación, pruebas y verificación del ciclo de vida de desarrollo de software, lo que permitirá desarrollar un software que no solo seguro, sino de calidad y con menos costo.

En conclusión, dado el tiempo de desarrollo, la disponibilidad de acceso a documentación, proceso y guías y complejidad como principales características se propone el uso de la metodología SAMM perteneciente a la fundación OWASP como una guía para desarrollo seguro. Como complemento a esta metodología se debe también realizar la implementación de las Guías “Una Guía para Construir Aplicaciones y Servicios Web Seguros” y Buenas prácticas de codificación que son propuestas por OWASP para de esta forma nuestro aplicativo sea más seguro y con menor grado de riesgo y amenazas.

## 4.2. Caso de Estudio

En el siguiente caso de estudio se basará en una empresa proveedora de Software que tiene un año en el mercado, Dicha empresa se dedica al desarrollo de software local para negocios, como también al desarrollo web, entre otros.

Esta empresa presenta los siguientes recursos para desarrollo web:

| Ambiente | Recursos | Metodologías |
|---|---|---|
| PHP | 1 diseñador Web<br>3 desarrolladores | SCRUM |
| HTML | | |
| MYSQL | | |

**Tabla 9: Recursos de Empresa de Caso de Estudio**

Al ser una empresa pequeña y sin mucho tiempo en el mercado, el grado de seguridad en sus aplicativos webs es bajo, por lo cual se adaptará la metodología propuesta "SAMM". Inicialmente se realizó un análisis de la organización, midiendo el nivel de seguridad que posee la empresa en sus aplicaciones web, mediante una tarjeta de calificaciones propuesta por esta metodología, de esta manera se puede observar en qué punto la organización está fallando, luego de esto se podrá implementar SAMM como una guía que permita construir y mejorar la iniciativa de aseguramiento de Software.

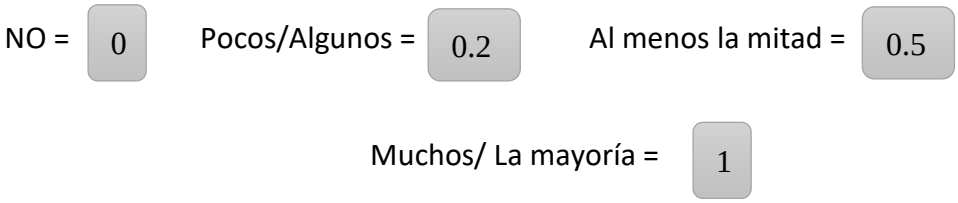



***Ilustración 36: Calificaciones de revisión***

***Fuente:*** (Deleersnyder et al., 2017)

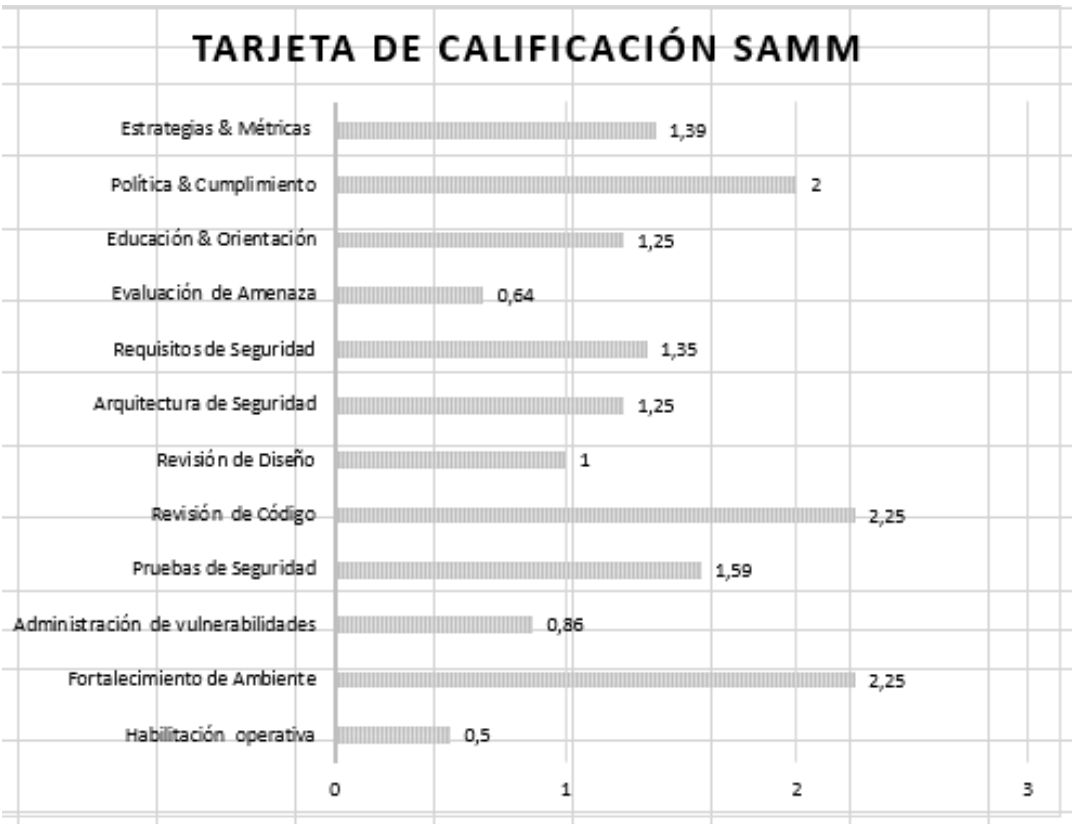



***Ilustración 37: Tarjeta de Calificación***

***Fuente: Anexo 1***

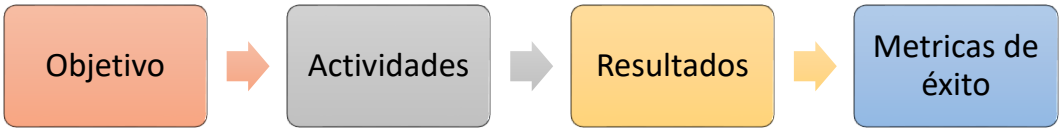
El resultado obtenido de la tarjeta de Calificación, es que la empresa necesita mejorar en varias áreas, se utilizó una escala de tres niveles del 0-3 en donde la mayoría de tareas se encuentran en nivel medio, lo que quiere decir que los aplicativos poseen vulnerabilidades que pueden ser explotadas fácilmente.

En base a esta evaluación se deberá realizar un plan de implementación por fases adecuando a las necesidades de la organización, que permitan la mejora en aseguramiento de software, reduciendo las vulnerabilidades de las aplicaciones y de esta forma reduciendo el riesgo de ataques.

### 4.2.1. Niveles de Madurez

El caso de estudio se desarrollará en base a los niveles de madurez propuestos por SAMM, en donde cada una de las 12 prácticas consta de tres niveles de madurez, cada nivel nos ayudará con diferentes compones que nos permitirá alcanzar el nivel establecido.

Para esto se deberá definir para cada fase lo siguiente:




- **Objetivo:** Objetivo de seguridad propuestos en cada etapa.
- **Actividad:** Requisitos básicos propuestos en cada nivel, se definirá que actividades son necesarias a llevar a cabo para el aseguramiento del software, estas serán ejecutadas por un rol especifico.
- **Resultados:** Resultados a alcanzar en cada nivel
- **Métricas de Éxito:** Se definirá las métricas iniciales que permitan analizar como la organización se desempeña.

(OWASP, 2017)

### 4.2.2. Pruebas

En este caso de estudio, adicional a las pruebas de rendimiento, se realizarán pruebas de seguridad, las cuales permitirán el aseguramiento de la integridad y la confiabilidad del sistema. Se tiene como objetivo principal para estas pruebas: Captar las vulnerabilidades que puede tener el sistema antes de que salga a producción.

Las pruebas principales que se manejaran son:

- Autentificación
- Gestión de Sesiones
- Pruebas de Autorización
- Pruebas de Validación de Datos

### 4.2.3. Fases y Paquetes de Trabajo

En base a la revisión inicial, en este caso la organización no realiza alguna activad de aseguramiento, se iniciará realizando un plan de implementación que se adapte a las necesidades de la empresa en cuestión.

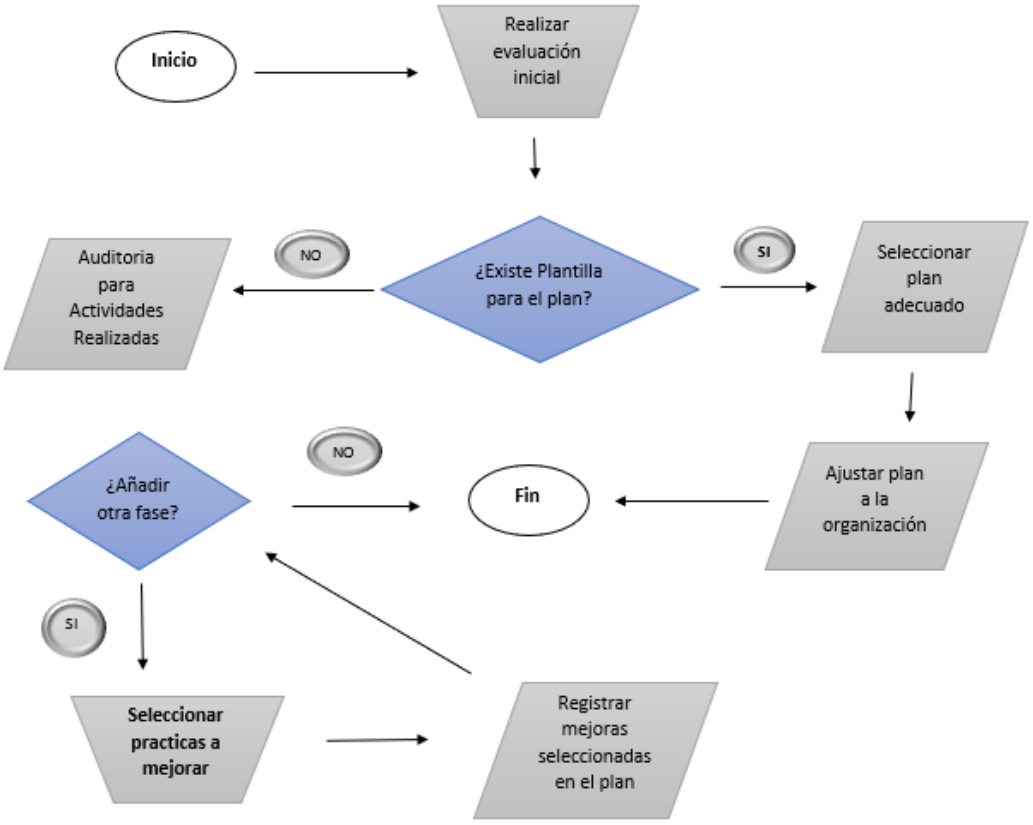



***Ilustración 38: Plan de implementación***

***Fuente:*** (Guide, 2017)

Se propone 5 fases en las cuales se definirán las prácticas más esenciales.

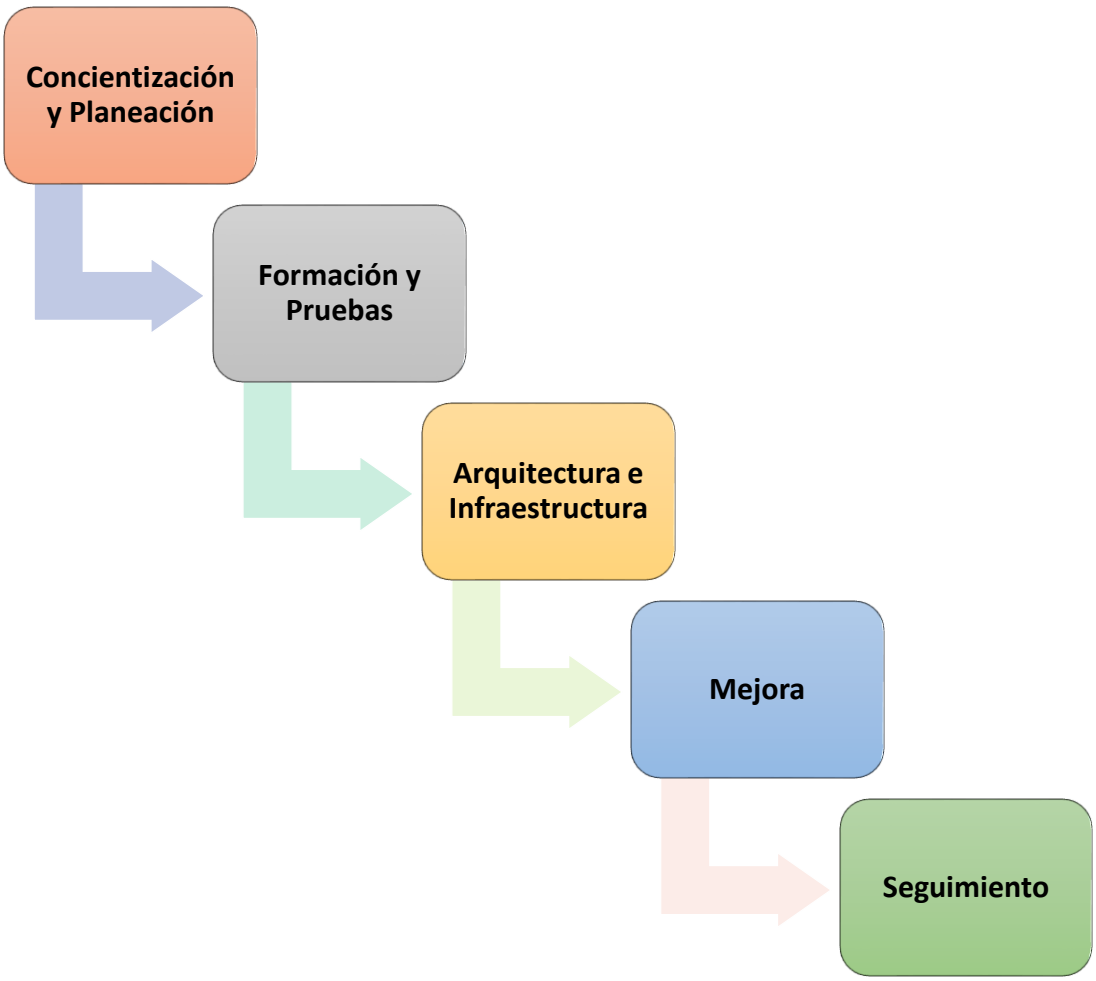



***Ilustración 39: Fases***

***Fuente: (Deleersnyder et al., 2017)***

Como principal objetivo al aplicar esta metodología propuesta por OWASP, se tiene la evolución de la organización respecto a seguridad en el tiempo que dura su implementación, para lo cual se enfocará principalmente en la educación y concientización por parte del equipo de desarrollo.

Basándonos en la plantilla para Plan de Implementación de Proveedores independientes de Software, se propone el siguiente Plan de implementación, el cual constará de 4 fases.

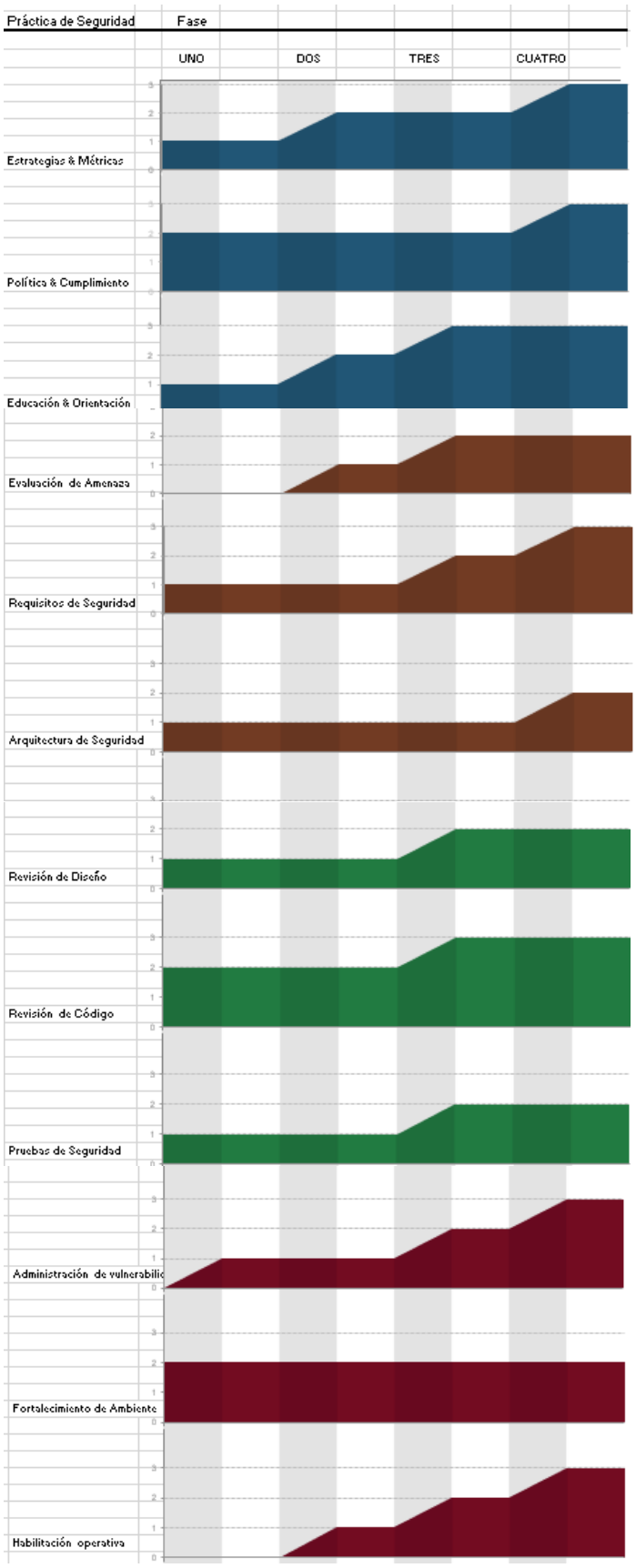



**Ilustración 40: Plantilla para Plan de Implementación de Proveedores independientes de Software**
***Fuente:*** (Payne et al., 2016)

En base a las fases expuestas anteriormente se proponen los siguientes paquetes de trabajo, en las cuales como Participantes se tendrá a todo el equipo de trabajo, debido a que como se había mencionado con anterioridad cada uno desempeñará más de un Rol.

Al completar las cuatro fases luego de los 12 meses que se tiene plantificado, se realizará un seguimiento y control, de esta forma se validará que la empresa funcione en base a los parámetros estipulados.

| Nombre de la Fase: | Concientización y Planeación |
|---|---|
| Participantes | Todo el equipo de trabajo |
| Duración de la fase | 0 a 3 meses |

**Objetivos**

- Capacitación del personal de la organización
- Fomentar el conocimiento en temas de seguridad web
- Identificar principales vulnerabilidades de seguridad
- Análisis de riesgos

**Actividades**

**SM1**

- Estimar el perfil general de riesgo del negocio
- Construir y mantener la hoja de ruta del programa de aseguramiento

**EG1**

- Capacitación técnica de sensibilización en seguridad
- Construir y mantener lineamientos técnicos

**SR1**

- Derivar los requisitos de seguridad de la funcionalidad empresarial

**IR1**

- Crear listas de verificación de revisión de los requisitos de seguridad conocidos

**Resultados**

**SM1**

- Lista de Riesgos más Críticos causados por Software a nivel del Negocio
- Plan de Implementación adaptado, que soluciones necesidades de seguridad

**EG1**

- Concientización de parte de desarrolladores sobre los problemas comunes que se dan a nivel de código, manteniendo un software con mejores prácticas de seguridad

**SR1**

| |
|---|
| • Capturar ad hoc de mejores prácticas de seguridad como requisitos<br>**IR1**<br>Inspección de vulnerabilidades comunes que se presentan en el código |
| **Entregables** |
| • **Anexo 1:** Tarjeta de Calificaciones<br>• **Anexo 2:** Análisis de Riesgo<br>**b)** Identificación de Riesgos<br>• **Anexo 3:** Plan de Capacitación |

**Fuente:** (OWASP, 2017)

| **Nombre de la Fase:** | Formación y Pruebas |
|---|---|
| Participantes | Todo el equipo de trabajo |
| Duración de la fase | 3 a 6 meses |

**Objetivos**

- Capacitación del personal en cuanto a revisión, identificación y corrección de vulnerabilidades en código
- Capacitación en el uso de herramientas automatizadas
- Fijar niveles de riesgos de negocio

**Actividades**

**SM2**

- Clasificar datos y aplicaciones en función de riesgo empresarial

**PC2**

- Construir políticas y estándares de Seguridad y cumplimiento
- Establecer la práctica de auditoria de proyectos

**EG2**

- Capacitaciones de seguridad especificadas de rol
- Entrenadores en temas de seguridad

**TA1**

- Construir modelos de amenaza para la aplicación
- Desarrollar perfil del atacante

**ST2**

- Uso de herramientas de prueba
- Integrar pruebas de seguridad

**Resultados**

**SM2**

- Planes de aseguramiento basados en el valor central del negocio

**PC2**

- Sensibilización por parte de los integrantes de la organización a lo que se refiere a expectativas de seguridad y cumplimiento

**EG2**

| |
|---|
| • Concientización por parte del equipo de las vulnerabilidades de seguridad en el producto, diseño o código<br>• Planes de acción en caso de que se presente fallas de diseño<br>**ST2**<br>• Validación de la funcionalidad de seguridad del SW<br>• Equipo con la capacidad de evaluar y corregir problemas antes de la liberación del producto |
| **Entregables** |
| **Anexo 2:** Análisis de Riesgos<br>a) Metodología de Riesgo<br>**Anexo 3:** Plan de Capacitación<br>**Anexo 4:** Modelo de Amenazas |

**Fuente:** (OWASP, 2017)

| **Nombre de la Fase:** | **Arquitectura e Infraestructura** |
|---|---|
| Participantes | Todo el equipo de trabajo |
| Duración de la fase | 6 a 9 meses |

Objetivos

- Construcción de casos de abuso
- Establecer proceso de respuesta de incidencias
- Establecer políticas y lineamientos técnicos respecto a codificación seguro

**Actividades**

**TA2**

- Construcción de modelos de Caso de abuso

**SR2**

- Construcción de matriz de Control de Acceso
- Especificar requisitos de seguridad basados en riesgos conocidos

**VM2**

- Establecer Proceso de respuestas a incidencia
- Adoptar un Plan de Divulgación de problemas de seguridad

**Resultados**

**TA2**

- Marco que permita la toma de decisiones en la presentación de amenazas

**SR2**

- Conocimiento de los diferentes escenarios de ataque que se pueden presentar

**VM2**

- Plan de comunicaciones para el manejo de reportes de vulnerabilidades
- Plan de respuestas de incidentes que permita el control y seguimiento del incidente presentado.

- **Entregables**

**Anexo 5:** Modelo de Caso de Abuso

**Anexo 6:** Matriz de Control de Acceso

| |
|---|
| **Anexo 7:** Proceso de respuesta a incidentes |
| **Anexo 8:** Plan de divulgación |

**Fuente:** (OWASP, 2017)

| **Nombre de la Fase:** | **Mejora** |
|---|---|
| Participantes | Todo el equipo de trabajo |
| Duración de la fase | 9 a 12 meses |
| Objetivos | |
| • Mejorar las funciones de seguridad que posee la organización<br>• Análisis de la aplicación antes de su liberación<br>• Mejora en la seguridad de la aplicación | |
| **Actividades** | |
| **PC3**<br>• Crear puertas de cumplimiento para proyectos<br>**EG3**<br>• Establecer un examen / certificación basado en roles.<br>**DR3**<br>• Establecer puertas de liberación para la revisión de diseño<br>**CR3**<br>• Análisis de Código<br>• Establecer puertas de liberación para revisión de código<br>**ST3**<br>• Establecer puertas de liberación para pruebas de seguridad<br>**VM3**<br>• Realizar análisis de causa raíz para incidentes<br>• Recopilar métricas por incidente<br>**EH3**<br>• Identificación e implementación de herramientas de protección de operaciones<br>**OE3**<br>• Implementar firma de código para componentes en una aplicación | |
| Resultados | |
| • Garantía del cumplimiento a nivel del proyecto<br>• Mitigación de nuevas amenazas y vulnerabilidades | |

- Análisis de puntos débiles en diseño, arquitectura y código del sistema
- Incrementación de confianza en la precisión del análisis de código
- Retroalimentación de los incidentes ocurridos, que permiten la mejora de la organización
- Reducción en probabilidad de que un ataque tenga éxito

**Fuente:** (OWASP, 2017)

# CAPITULO 5: CONCLUSIONES Y RECOMENDACIONES

## 5.1. Conclusiones

- La mayoría de ataques cibernéticos son dirigidos a servicios web, debido al desarrollo de herramientas que puedan explotar las vulnerabilidades de estos sitios, en la capa de aplicación el 37% de los ataques realizando son dirigidos al protocolo HTTP, en su mayoría son ataques de fuerza bruta, siendo el 40% de estos dirigidos a la obtención de la información.

- La implementación de una metodología no hará que el aplicativo sea 100% seguro, sino que reducirá en el mayor grado los errores que presenta el aplicativo, los cuales dependerán del grado de complejidad y tamaño del software

- Existe diferentes metodologías que son enfocadas a seguridad, sin embargo, la metodología más factible de las metodologías analizadas basando en el tiempo de desarrollo, disponibilidad de información y complejidad es SAMM de la fundación OWASP

- SAMM es una metodología flexible, adaptable a riesgos y a vulnerabilidades que se pueden presentar en el aplicativo, dando soluciones que permitan mitigar la mayoría de estos, lo que hace que sea una metodología factible para la aplicación en cualquier organización.

- SAMM es una metodología que involucra el uso de más Recursos - Personas que SDL, debido a todos los roles que se maneja, pero en la cual se requiere menos esfuerzo y tiempo ya que solo consta de 4 funciones de Negocio y 12 Prácticas.

- SDL es una metodología que cuenta casi con las mismas características de SAMM, pero debido a que el acceso a su documentación es muy restringido no

la hace una metodología flexible y que se pueda aplicar a cualquier tipo de organización.

## 5.2. Recomendaciones

- Se recomienda que antes de la implementación de una metodología se realice un análisis y estudio de esta, de esta forma nos permitirá saber si la metodología sería adecuada para la organización, debido a que su mala implementación puede causar la generación de errores, aumentando el riesgo y las amenazas en la aplicación.

- Se recomienda la realización de un modelo de amenazas y un plan de riesgos. De esta forma la organización puede estar preparada y mitigar los riegos que pueden presentarse. No toda aplicación es 100% segura, por lo cual se recomienda la constante investigación en cuanto amenazas y ataques cibernéticos que se realizan a aplicativos webs.

- Se recomienda realizar medidas continuas al aplicar una metodología, en este caso SAMM, es decir se deberá registrar las calificaciones o el nivel de mejora en periodos de tiempos estipulados y consistentes, que permitan ver que la mejora de aseguramiento del software existe, de igual forma realizar una comparativa con un antes y después de la aplicación, de esta forma podremos observar si se aumentó el nivel de madurez de las aplicaciones en la empresa.

# ANEXOS

## Anexo 1 - Tarjeta de Calificación SAMM

| Governance | | SCORE | | | |
|---|---|---|---|---|---|
| | | 0,0 | 0,2 | 0,5 | 1,0 |
| **Strategy & Metrics** | | | | | |
| ***SM1*** | Is there a software security assurance program in place? | X | | | |
| | Are development staff aware of future plans for the assurance program? | | | X | |
| | Do the business stakeholders understand your organization's risk profile? | | | | X |
| ***SM2*** | Are many of your applications and resources categorized by risk? | | | X | |
| | Are risk ratings used to tailor the required assurance activities? | | | X | |
| | Does the organization know about what's required based on risk ratings? | | | X | |
| ***SM3*** | Is per-project data for the cost of assurance activities collected? | X | | | |
| | Does your organization regularly compare your security spend with that of other organizations? | | | | X |
| **Policy & Compliance** | | | | | |
| ***PC1*** | Do project stakeholders know their project's compliance status? | | | | X |
| | Are compliance requirements specifically considered by project teams? | | | | X |
| ***PC2*** | Does the organization utilize a set of policies and standards to control software development? | | | | X |
| | Are project teams able to request an audit for compliance with policies and standards? | | | X | |

| | | | | | |
|---|---|---|---|---|---|
| ***PC3*** | Are projects periodically audited to ensure a baseline of compliance with policies and standards? | | | X | |
| | Does the organization systematically use audits to collect and control compliance evidence? | X | | | |
| **Education & Guidance** | | | | | |
| ***EG1*** | Have developers been given high-level security awareness training? | X | | | |
| | Does each project team understand where to find secure development best-practices and guidance? | | | | X |
| ***EG2*** | Are those involved in the development process given role-specific security training and guidance? | | | X | |
| | Are stakeholders able to pull in security coaches for use on projects? | X | | | |
| ***EG3*** | Is security-related guidance centrally controlled and consistently distributed throughout the organization? | | | | X |
| | Are developers tested to ensure a baseline skillset for secure development practices? | X | | | |

**Fuente: (Deleersnyder et al., n.d.)**

| Construction | | SCORE | | | |
|---|---|---|---|---|---|
| | | 0,0 | 0,2 | 0,5 | 1,0 |
| **Threat Assessment** | | | | | |
| ***TA1*** | Do projects in your organization consider and document likely threats? | X | | | |
| | Does your organization understand and document the types of attackers it faces? | X | | | |
| ***TA2*** | Do project teams regularly analyze functional requirements for likely abuses? | X | | | |
| | Do project teams use a method of rating threats for relative comparison? | X | | | |
| | Are stakeholders aware of relevant threats and ratings? | | | X | |
| ***TA3*** | Do project teams specifically consider risk from external software? | | | X | |
| | Are the majority of the protection mechanisms and controls captured and mapped back to threats? | | | X | |
| **Security Requirements** | | | | | |
| ***SR1*** | Do project teams specify security requirements during development? | | | X | |
| | Do project teams pull requirements from best practices and compliance guidance? | X | | | |
| ***SR2*** | Do stakeholders review access control matrices for relevant projects? | X | | | |
| | Do project teams specify requirements based on feedback from other security activities? | | | | X |
| ***SR3*** | Do stakeholders review vendor agreements for security requirements? | | | | X |
| | Are audits performed against the security requirements specified by project teams? | | X | | |
| **Secure Architecture** | | | | | |

| | | | | | |
|---|---|---|---|---|---|
| ***SA1*** | Are project teams provided with a list of recommended third-party components? | X | | | |
| | Are project teams aware of secure design principles and do they apply them consistently? | | | | X |
| ***SA2*** | Do you advertise shared security services with guidance for project teams? | X | | | |
| | Are project teams provided with prescriptive design patterns based on their application architecture? | | | | X |
| ***SA3*** | Do project teams build software from centrallycontrolled platforms and frameworks? | | | X | |
| | Are project teams audited for the use of secure architecture components? | X | | | |

**Fuente: (Deleersnyder et al., n.d.)**

| Verification | | SCORE | | | |
|---|---|---|---|---|---|
| | | 0,0 | 0,2 | 0,5 | 1,0 |
| **Design Review** | | | | | |
| ***DR1*** | Do project teams document the attack perimeter of software designs? | X | | | |
| | Do project teams check software designs against known security risks? | X | | | |
| ***DR2*** | Do project teams specifically analyze design elements for security mechanisms? | X | | | |
| | Are project stakeholders aware of how to obtain a formal secure design review? | | | X | |
| ***DR3*** | Does the secure design review process incorporate detailed data-level analysis? | | | | X |
| | Does a minimum-security baseline exist for secure design review results? | | | X | |
| **Implementation Review** | | | | | |
| ***IR1*** | Do project teams have review checklists based on common security related problems? | | | | X |
| | Do project teams review selected high-risk code? | | | | X |
| ***IR2*** | Can project teams access automated code analysis tools to find security problems? | | | | X |
| | Do stakeholders consistently review results from code reviews? | | | | X |
| ***IR3*** | Do project teams utilize automation to check code against application-specific coding standards? | X | | | |
| | Does a minimum-security baseline exist for code review results? | | | X | |
| **Security Testing** | | | | | |
| ***ST1*** | Do projects specify security testing based on defined security requirements? | | | | X |

| | | | | | |
|---|---|---|---|---|---|
| | Is penetration testing performed on high risk projects prior to release? | | | | X |
| | Are stakeholders aware of the security test status prior to release? | | | X | |
| ***ST2*** | Do projects use automation to evaluate security test cases? | | | X | |
| | Do projects follow a consistent process to evaluate and report on security tests to stakeholders? | X | | | |
| ***ST3*** | Are security test cases comprehensively generated for application-specific logic? | | X | | |
| | Does a minimum-security baseline exist for security testing? | | | X | |

**Fuente: (Deleersnyder et al., n.d.)**

| Operations | | SCORE | | | |
|---|---|---|---|---|---|
| | | 0,0 | 0,2 | 0,5 | 1,0 |
| **Issue Management** | | | | | |
| ***IM1*** | Do projects have a point of contact for security issues or incidents? | | | X | |
| | Does your organization have an assigned security response team? | X | | | |
| | Are project teams aware of their security point(s) of contact and response team(s)? | | | X | |
| ***IM2*** | Does the organization utilize a consistent process for incident reporting and handling? | X | | | |
| | Are project stakeholders aware of relevant security disclosures related to their software projects? | X | | | |
| ***IM3*** | Are incidents inspected for root causes to generate further recommendations? | | | | X |
| | Do projects consistently collect and report data and metrics related to incidents? | X | | | |
| **Environment Hardening** | | | | | |
| ***EH1*** | Do projects document operational environment security requirements? | | | | X |
| | Do projects check for security updates to third-party software components? | | | | X |
| ***EH2*** | Is a consistent process used to apply upgrades and patches to critical dependencies? | | | | X |
| | Do projects leverage automation to check application and environment health? | | | X | |
| ***EH3*** | Are stakeholders aware of options for additional tools to protect software while running in operations? | | | X | |
| | Does a minimum-security baseline exist for environment health (versioning, patching, etc)? | | | X | |

| Operational Enablement | | | | | |
|---|---|---|---|---|---|
| ***OE1*** | Are security notes delivered with each software release? | X | | | |
| | Are security-related alerts and error conditions documented on a per-project basis? | | | | X |
| ***OE2*** | Do projects utilize a change management process that's well understood? | X | | | |
| | Do project teams deliver an operational security guide with each product release? | X | | | |
| ***OE3*** | Are project releases audited for appropriate operational security information? | X | | | |
| | Is code signing routinely performed on software components using a consistent process? | X | | | |

**Fuente: (Deleersnyder et al., n.d.)**

## Anexo 2 - Análisis de Riesgos

### a) Metodología de Riesgo

El enfoque que tiene OWASP para el análisis de Riesgos de negocio se basa en un modelo de riesgos estándar:

***Riesgo = Probabilidad * Impacto***

OWASP plantea la metodología de categorización de riesgos:

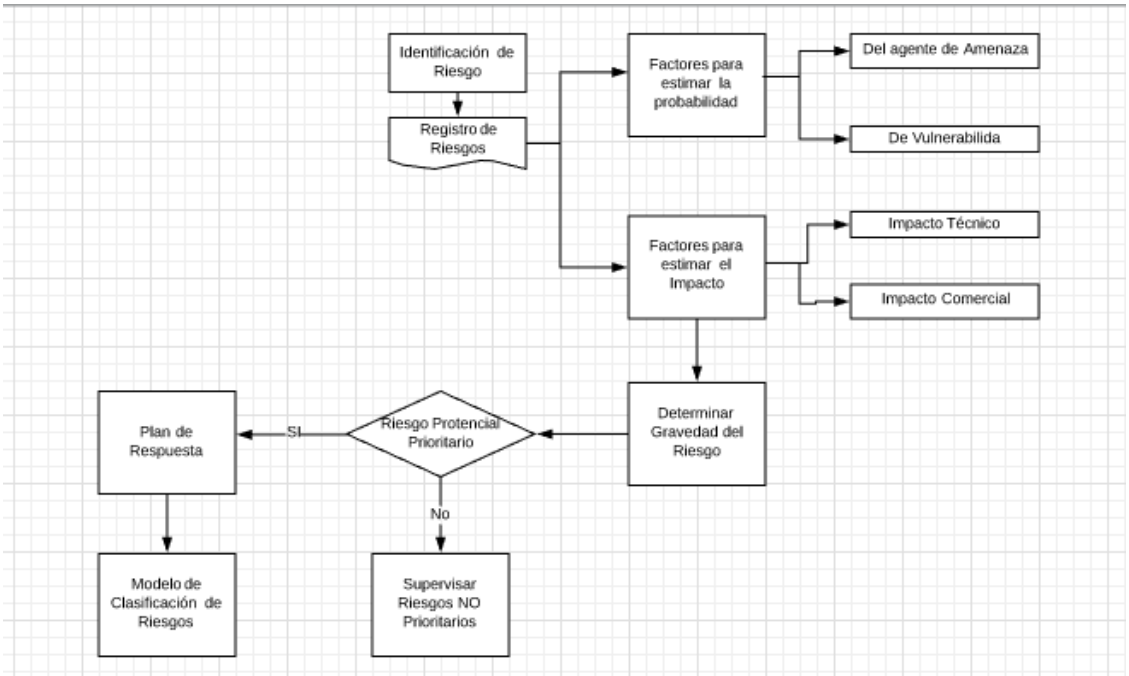



***Ilustración 41: Flujo de Análisis de Riesgos***
***Fuente:*** (OWASP, 2019a)

### b) Responsabilidades por Proceso

| PROCESOS | RESPONSABILIDADES |
|---|---|
| **Planificación - Gestión de Riesgos** | • Realización de Plan de Gestión de Riesgos<br>• Aprobar el Plan de gestión de Riesgos<br>• Ejecutar plan de riesgos |
| **Identificación de Riesgos** | • Identificar los riesgos o amenazas presentes y futuras. |
| **Análisis de Riesgos** | • Realizar el análisis de categorización de riesgos, priorizándolos mediante la matriz de Probabilidad e Impacto. |
| **Plan de Respuesta a los Riesgos** | • Realizar un plan de respuesta a los riesgos identificados. |
| **Monitoreo y Control** | • Ejecutar Plan de Respuesta.<br>• Identificación y Registro de Nuevos Riesgos |

**Tabla 10: Proceso y Responsabilidades**

### c) Identificación de Riesgos

Un ataque puede usar diferentes rutas en una aplicación, las cuales se convierten en riesgos, los cuales fueron o no atendidos en su debido tiempo y que pueden afectar a la organización.

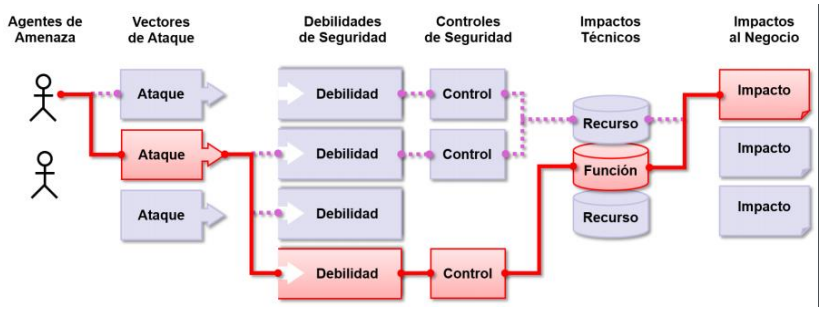



**Ilustración 42: Flujo de ataques**
**Fuente:** (OWASP Foundation, 2017)

Considerando los factores expuestos anteriormente se realiza la estimación general de riesgos de negocio de la empresa en la que se realiza el caso de estudio, nos colocaremos en el primer punto que es la "Identificación de Riesgos" realizando una lista de los peores escenarios, en los cuales se recomienda que sea una lista de entre 3 a 7 escenarios.

En este caso los cinco primeros escenarios describirán amenazas que explotan las vulnerabilidades de la aplicación, los 2 restantes se consideran a dos escenarios uno que puede causar que una aplicación tenga vulnerabilidades y el otro escenario que es la consecuencia de que en una aplicación se explote sus vulnerabilidades.

| ID | Escenario | PLAN DE ACCIÓN |
|---|---|---|
| **R1** | Manipulación de Datos | Implementación de métodos de control de acceso que permitan evitar la explotación de vulnerabilidades previniendo ataques tipo Inyección SQL |
| **R2** | Robo y alteración de datos sensibles | Implementación de métodos de protección para evitar la explotación de vulnerabilidades y prevenir la exposición de datos sensibles como métodos criptográficos |
| **R3** | Suplantación de Identidad | Implementación de métodos de autentificación y gestión de sesiones permitirán evitar comprometer la información de los usuarios como usuario y contraseña. |

| | | |
|---|---|---|
| **R4** | Pérdida de control de Acceso | Establecer políticas de acceso |
| **R5** | Errores de Programación | Implementación de políticas, estándares y técnicas de desarrollo en la empresa. |
| **R6** | Falta de conocimiento por parte de TI | Capacitación al personal de la organización |
| | Perdida monetaria | Evaluación del sistema antes de salir a producción, validando las vulnerabilidades existentes. |

**Tabla 11: Lista de Escenarios**

### d) Nivel de Riesgo

- **Gravedad de Riesgo**

Se determinará la Gravedad de riesgos identificados. Determinando el impacto y la probabilidad de estos mediante la siguiente escala.

| Probabilidad y niveles de impacto | |
|---|---|
| 0 a <3 | BAJO |
| 3 a <6 | MEDIO |
| 6 a 9 | ALTO |

**Ilustración 43: Gravedad de riesgo (Probabilidad e Impacto)**

**Fuente:** (OWASP, 2019a)

Se Considerará:

| Impacto | Descripción | Valor |
|---|---|---|
| **Alta** | Vulnerabilidad que si se la explota comprometa la seguridad de la información de la organización causando un daño negativo en esta por lo cual debe solucionarse inmediatamente. | 1 |
| **Media** | Vulnerabilidad que si se la explota tendría un impacto leve sobre el negocio, por lo que se puede solucionar en un tiempo prudente. | 2 |
| **Baja** | Vulnerabilidad que, si se la explota no causar mayor inconveniente, por lo cual su solución no es considerada inmediatamente. | 3 |

***Tabla 12: Probabilidad de Ocurrencia***

- **Severidad**

Se combinará las medidas de Probabilidad e impacto para poder conseguir la calificación de severidad de riesgo

| Gravedad de riesgo general | | | | |
|---|---|---|---|---|
| **Impacto** | ALTO | Medio | Alto | Crítico |
| | MEDIO | Bajo | Medio | Alto |
| | BAJO | Nota | Bajo | Medio |
| | | BAJO | MEDIO | ALTO |
| | **Probabilidad** | | | |

***Ilustración 44: Severidad de riesgo***

**Fuente:** (OWASP, 2019a)

| Nivel | Descripción |
|---|---|
| **Crítico** | Riesgo extremo, se necesita mitigación inmediata de la amenaza. Compromete información de la empresa. |
| **Alto** | |
| **Medio** | Riesgo moderado, se inicia proceso de análisis de riesgos y se evalúa acciones para la mitigación de este. |
| **Bajo** | Riesgo Bajo, no existe ninguna afectación, se consideran “Riesgos no Prioritarios" |
| **Ninguno** | Se considera no ejecutar ningún proceso ya que no existe afectación. |

**Tabla 13: Descripción de Niveles de Riesgo**

En este caso se realizar la captura de respuestas por medio de un método informal el cual consiste en el análisis de los diferentes factores para la estimación del nivel de impacto.

| ID | Escenario | Nivel de Riesgo |
|---|---|---|
| **R001** | Manipulación de Datos | Alto |
| **R002** | Robo y alteración de datos sensibles | Alto |
| **R003** | Suplantación de Identidad | Alto |
| **R004** | Pérdida de Control de Acceso | Alto |
| **R005** | Errores de Programación | Medio |
| **R006** | Falta de conocimiento por parte de TI | Alto |
| **R007** | Perdida monetaria | Alto |

**Tabla 14: Estimación informal de nivel de impacto**

## Anexo 3 - Plan de Capacitación

### a) Alcance

El presente plan de Capacitación tiene como objetivo la concientización del personal de la organización sobre las amenazas y vulnerabilidades de una aplicación. Así como también los métodos que permitan la reducción de riesgos en estas.

### b) Plan de Capacitación

| Fase 1 - Concientización y Planeación | SEMANA | | | |
|---|---|---|---|---|
| **Actividades a Desarrollar** | **1** | **2** | **3** | **4** |
| Mejores Prácticas - Validación de entrada | X | X | | |
| Codificación de Salida | X | X | | |
| Manejo de Errores | X | X | | |
| Registro | | | X | X |
| Autentificación | | | X | X |
| Autorización | | | X | X |
| Vulnerabilidades Comunes en el SW | | | X | X |
| Realización de Ad Hoc de Mejores Prácticas | X | X | | |
| Realización de Lista de verificación de Código | | | X | X |

**Tabla 15: Capacitación Fase 1**

En base a la capacitación realizada al personal de la organización, se realizar un ad hoc de las mejores prácticas de seguridad que se implementaran en el desarrollo de aplicaciones en la organización.

| Fase 2 - Formación y Pruebas | Semana | | | |
|---|---|---|---|---|
| **Actividades a Desarrollar** | **1** | **2** | **3** | **4** |

| | | | | |
|---|---|---|---|---|
| Seguridad en función del rol | X | | | |

**Tabla 16: Capacitación fase 2**

Los temas recomendados dependiendo del Rol son los siguientes:

| Rol | Actividad |
|---|---|
| Gerentes | Planificación de Requisitos de Seguridad |
| | Gestión de Vulnerabilidades |
| | Modelo de Amenazas |
| | Diseño de Casos de Uso |
| Auditores y evaluadores | Técnicas de Revisión de Código |
| | Técnicas de Revisión de Arquitectura |
| | Análisis de diseño |
| | Análisis de tiempo ejecución |
| | Planificación Pruebas de Seguridad |
| Desarrollador y Arquitecto | Patrones de Diseño de Seguridad |
| | Herramientas de seguridad |
| | Modelo de Amenazas |
| | Técnicas de evaluación de SW |

**Tabla 17: Roles Generales**

En base a las capacitaciones realizadas se procederá a realizar las políticas y estándares de seguridad que se establecerán en el desarrollo de la organización.

| **Fase 3 – Fase 4** | Semana | | | |
|---|---|---|---|---|
| **Actividades a Desarrollar** | **1** | **2** | **3** | **4** |
| Consultoría de seguridad – Temas adicionales | X | X | | |
| Mejoras | | | X | |

**Tabla 18: Capacitación Fase 3 – Fase 4**

Debido a que la empresa es pequeña y cuenta con poco personal, se considerará un tiempo total de 2 meses para la capacitación de los diferentes temas propuestos, con un tiempo de 1 hora y media al día, dando un total de 60 horas.

### c) Presupuesto

Para la implementación de esta metodología es necesario realizar una inversión económica debido a que el equipó necesita adquirir conocimiento de seguridad lo cual se realizará por medio de consultores.

Como gastos personales se considera los siguientes puntos:

- La implementación de la metodología tiene la duración de un año
- No se considera el pago de horas extras para el personal de la organización
- No se trabará fines de semana ni feriados

**Costos de Implementación:**

| RECURSOS | CANTIDAD | TIEMPO | VALOR | TOTAL |
|---|---|---|---|---|
| Recurso Externo (Consultores) | 1 | 2 meses | $ 10.000 | $ 20.000 |
| Recurso Interno | 4 | 12 meses | $ 0.00 | $ 0.00 |
| Herramientas de seguridad | 1 | - | $ 0.00 | $ 0.00 |
| **TOTAL** | | | | **$ 20.000** |

**Tabla 19: Presupuesto**

**Recurso Externo:** Consultores con conocimientos de seguridad, se encargarán de impartir la capacitación al personal de la organización.

**Recurso Interno:** Personal de la organización, no se considerará un costo adicional para Recursos Internos.

**Herramientas de Seguridad:** Herramientas para escaneo de vulnerabilidades y pruebas de seguridad, se adquiere productos sin ningún costo y que se encuentran disponibles, como, por ejemplo: ZAP OWASP 2.7.0.

(OWASP, 2017)

A la finalización de la capacitación, se debe presentar los siguientes entregables:

| N° Producto | Tipo | Descripción |
|---|---|---|
| Producto 0 | Documento | Lineamientos Técnicos |
| Producto 1 | Documento | Políticas y estándares de Seguridad |
| Producto 2 | Documento | Plan de auditoria |
| Producto 3 | Documento | Manuales de uso de herramientas de seguridad |
| Producto 4 | Documento | Matriz de indicadores de seguridad |
| Producto 5 | Documento | Ad Hoc de Mejores Prácticas |
| Producto 6 | Documento | Lista de Verificación de Código |
| Producto 7 | Documento | Listado de Escenarios y vulnerabilidades comunes |
| Producto 8 | Documento | Informe Técnico de resumen de los productos entregados |
| Producto 9 | Otros | Documentos presentados en la capacitación y se crea necesario su implementación en la organización |
| **Entregables** | | |
| Los documentos que se generen durante el tiempo de capacitación, deberán ser entregados en formatos electrónicos (EXCEL, PDF, DOC), adicionalmente se requiere una copia impresa.<br>Los documentos deberán tener el siguiente formato: control de versiones, autor, fecha, revisor y aprobador de consultoría. | | |

**Tabla 20: Entregables luego de Capacitación**

## Anexo 4: Modelo de Amenazas

SAMM utiliza el modelo de amenazas desarrollado por Microsoft, el cual nos ayudará a identificar las vulnerabilidades, establecer contramedidas para su prevención o mitigar los efectos que causa la amenaza. Un modelo de amenazas se puede descomponer en tres pasos generales:

| No | Pasos | Descripción |
|---|---|---|
| **1** | Descomponer la aplicación | Creación de Caso de Uso<br>Identificación de Puntos de Entrada<br>Identificación de Activos<br>Niveles de Confianza |
| **2** | Determinar y Jerarquizar Amenazas | Identificar Amenazas Críticas |
| **3** | Determinar Contramedidas y Mitigaciones | Mapeo de Amenazas - Contramedidas |

**Tabla 21: Modelo de Amenazas**
**Fuente:** (OWASP, 2019b)

En este caso de estudio se escogió la metodología "**Threat Analysis and Modeling**", la cual se puede esquematizar de la siguiente manera:

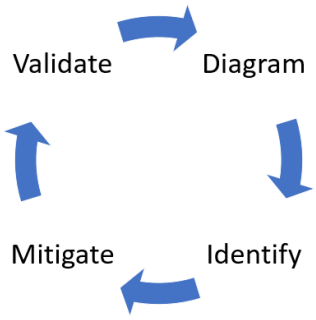



**Ilustración 45: Flujo de "Threat Analysis and Modeling"**

**Fuente:**(Microsoft, 2018a)

### a) Etapas del modelo de Amenazas

Este modelo se basa en las siguientes etapas:

| ID | Etapa | Descripción |
|---|---|---|
| **E1** | Grupo de Análisis de Riesgo | Identificar objetos de seguridad.<br>Realizar una descripción general de la aplicación. |
| **E2** | Descomponer Aplicación | Conocer la arquitectura de la aplicación<br>Identificar principales funcionalidades<br>Elaboración de casos de uso o diagrama de flujo datos que permita observar los componentes de la aplicación |
| **E3** | Determinar y Categorizar Amenazas | Realización de Plantillas de categorización de Amenazas |
| **E4** | Asignación de Valor a Amenazas | Calcular el Riesgo Asociado a Cada Amenaza |
| **E5** | Respuesta a amenazas | Mitigación de Amenazas según el Riesgo |

**Tabla 22: Etapas del Modelo de Amenazas**

**Fuente:** (Olivares & Eduardo, n.d.)

Estas etapas se implementarán para cada uno de los proyectos o aplicaciones que la empresa se encuentre desarrollando.

**b) Categorización de Amenazas**

Se realiza plantillas de categorización de las amenazas que se identificó.

| **Categoría** | **Manipulación de datos** |
|---|---|
| **Descripción** | Inyección de comandos SQL |
| **Objetivo de la Amenaza** | Componentes de Acceso a base de datos |
| **Nivel de Riesgo** | Alto |
| **Técnicas de Ataque** | Comando SQL |
| **Contramedidas** | Implementación de métodos de control de acceso |

| **Categoría** | **Robo y alteración de datos sensibles** |
|---|---|
| **Descripción** | Divulgación de la información |
| **Objetivo de la Amenaza** | Información sensible |
| **Nivel de Riesgo** | Alto |
| **Técnicas de Ataque** | - |
| **Contramedidas** | Implementación de métodos de protección como métodos criptográficos o de cifrado |

| **Categoría** | **Suplantación de Identidad** |
|---|---|
| **Descripción** | Identificación falsa |
| **Objetivo de la Amenaza** | Usar información de la autenticación de otro usuario |
| **Nivel de Riesgo** | Alto |
| **Técnicas de Ataque** | - |

| | |
|---|---|
| **Contramedidas** | Implementación de métodos de autentificación Gestión de sesiones |

| **Categoría** | **Pérdida de control de Acceso** |
|---|---|
| **Descripción** | Elevación de privilegios |
| **Objetivo de la Amenaza** | Ingreso no autorizado |
| **Nivel de Riesgo** | Alto |
| **Técnicas de Ataque** | Elevación de privilegios |
| **Contramedidas** | Establecer políticas de acceso |

**Tabla 23 : Plantillas Categorización de Amenazas**

### c) Perfil del atacante

Se plantea un perfil inicial de los posibles atacantes, se establece para diferentes tipos de Industrias a las que como proveedores de software se puede implementar una aplicación. Con el paso de tiempo y recibiendo la capacitación adecuada se podrá realizar una actualización la plantilla.

| **Industria** | **Financieras Seguros** |
|---|---|
| **Tipo** | Persona motivados por dinero |
| **¿Qué Busca?** | Información sensible<br>Datos o credenciales de cuentas |
| **¿Cuál es su objetivo?** | Fraude<br>Beneficio Personal |
| **Técnica** | Ataques a nivel de Aplicación Ingeniería Social |

| | |
|---|---|
| **Industria** | **Manufacturera, Transporte, Tecnología, Utilidades, Infraestructura, Defensa** |

| | |
|---|---|
| **Tipo** | Personas dentro de la organización<br>Personas fuera de la organización<br>Combinación de Ambas |
| **¿Qué Busca?** | Información Sensible<br>Acceso a propiedad intelectual |
| **¿Cuál es su objetivo?** | Obtener ventajas estratégicas o tecnológicas<br>Fraude<br>Robo de Identidad<br>Pagos Fraudulentos |
| **Técnica** | Ingeniería Social |

| **Industria** | **Salud** |
|---|---|
| **Tipo** | - |
| **¿Qué Busca?** | Información Sensible (Información Médica, Identificación persona o datos de pago) |
| **¿Cuál es su objetivo?** | Robo o Modificación de Información<br>Fraude<br>Robo de Identidad<br>Pagos Fraudulentos |
| **Técnica** | Ingeniería Social |

| **Industria** | **Venta por menor, Alimento** |
|---|---|
| **Tipo** | - |
| **¿Qué Busca?** | Información Sensible |
| **¿Cuál es su objetivo?** | Fraude<br>Robo de Identidad<br>Pagos Fraudulentos<br>Robo de Propiedad Intelectual<br>Obtener inteligencia Corporativa |
| **Técnica** | Varias |

**Tabla 24: Plantillas Estimación Perfil de Atacante**

## Anexo 5: Modelo de Caso de abuso

El modelo de caso de abuso nos permitirá describir los escenarios que se consideran como mal uso, sean intencionales o no, en los cuales se planteará como un atacante podría aprovechar la vulnerabilidad y atacar la aplicación, se usará representación gráfica de los casos de abuso, lo cual nos permitirán establecer contramedidas o acciones que permitan la mitigación a la amenaza presentada.

Para esto se plante el siguiente proceso:

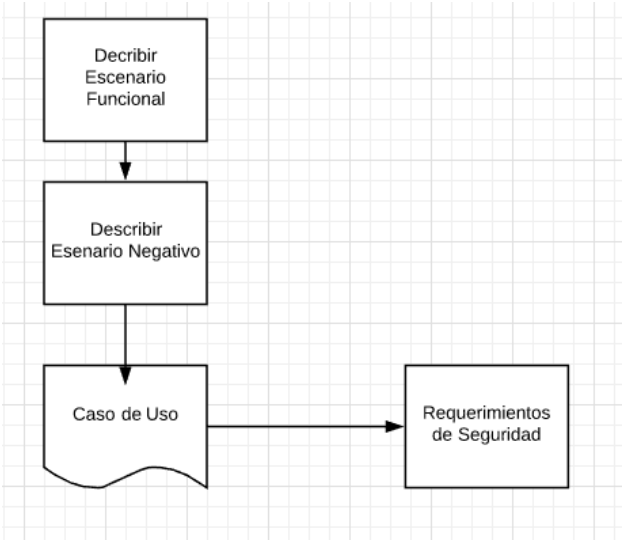



**Ilustración 46 : Flujo de Casos de Abuso**
**Fuente:**(*GUÍA DE PRUEBAS OWASP 2008 V3.0*, n.d.)

**Escenario Funcional:** Se valida acciones que realiza el usuario en el sistema, en el caso de Autentificación, el usuario ingresa credenciales para inicio de sección, el sistema valida si credenciales y genera mensaje de error si esta es fallida. (*GUÍA DE PRUEBAS OWASP 2008 V3.0*, n.d.)

**Escenario Negativo:** Descripción de acciones que realiza el atacante. Se rompe proceso de autentificación por medio de un ataque de fuerza bruta o de diccionario de contraseña, obteniendo la información del usuario desde los mensajes de error. (*GUÍA DE PRUEBAS OWASP 2008 V3.0*, n.d.)

**Caso de Uso:** Representación Gráfica flujo del escenario funcional incluyendo el ataque realizado al sistema (Escenario Negativo).

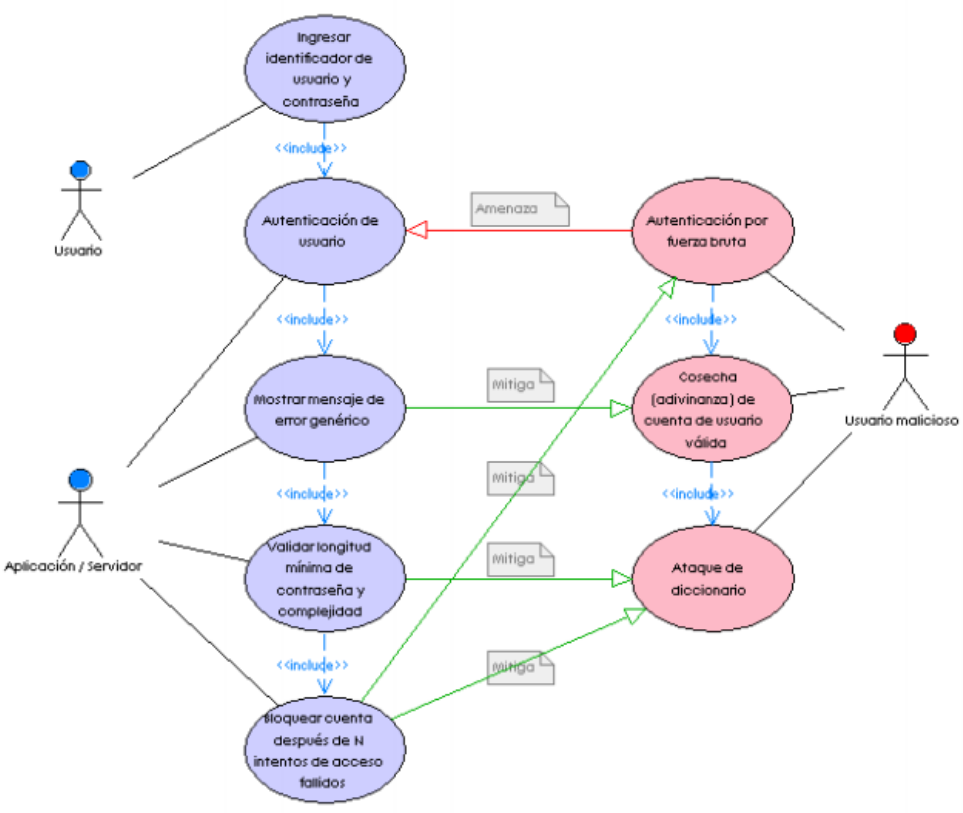



**Ilustración 47 : Caso de Abuso**
**Fuente:** (*GUÍA DE PRUEBAS OWASP 2008 V3.0*, n.d.)

**Obtener Requerimientos de Seguridad:** Se planteará los requerimientos que sean necesarios para la mitigación de la amenaza. Para la autentificación se plantearán métodos de autentificación en Gestión de Sesiones los cuales necesitan ser documentados. (*GUÍA DE PRUEBAS OWASP 2008 V3.0*, n.d.)

## Anexo 6: Matriz de Control de Acceso

La implementación de autentificación permitirá la restricción y control de accesos a recurso y archivos, la asignación de privilegios se llevará a cabo por medio de una lista de control de Acceso, en este caso se utiliza dos tipos de control de Acceso:

**Control de Acceso Basados en Roles (RBAC):** Se basa en los roles que desempeña en la organización, así como también las funciones a las que fue asignado. Se deberá tomar en cuenta factores como: Acción a realizar, cuando se realiza, donde, etc. (OWASP, n.d.) Para lo cual se define los siguientes perfiles de Trabajo en base a los Roles propuestos por SAMM:

| Rol | Responsabilidades |
|---|---|
| Desarrollador | Diseño e implementa SW |
| Arquitecto | Diseños de alto nivel |
| Gerente | Gestiona actividades del desarrollo del proyecto. |
| Personal de Control de Calidad | Realizar pruebas de calidad<br>Verificación previa al lanzamiento del aplicativo |
| Seguridad | Proveen conocimientos de seguridad<br>Pruebas de Seguridad |
| Dueños de Negocio | Toma de decisiones respecto al SW y sus requisitos comerciales: “Cliente que solicita el SW”. |

***Tabla 25: Perfiles de Trabajo en la Organización***

Debido a que la empresa es pequeña y se cuenta con mucho personal para que se desempeñe un rol por persona, por lo que una persona deberá desempeñar más de un rol a la vez. Es importante mencionar que para el desarrollo de un Software seguro es esencial de una persona que cumpla con el rol de Seguridad, para de esta forma pueda proveer a la organización con conocimientos esenciales para un software seguro.

**Control de Acceso discrecional:** En este caso se restringe el acceso a la información en base a la identidad del usuario, la decisión de acceso se realizará en función a la validación de credenciales que fueron entregadas al usuario y con las que realiza la autentificación. (OWASP, n.d.)

## Anexo 7: Proceso de respuesta a incidentes

Se propone un plan de respuesta de incidentes, que permita la detección, mitigación y recuperación de los incidentes que se puedan presentar.

La propuesta del plan de incidencias consta en 4 fases:

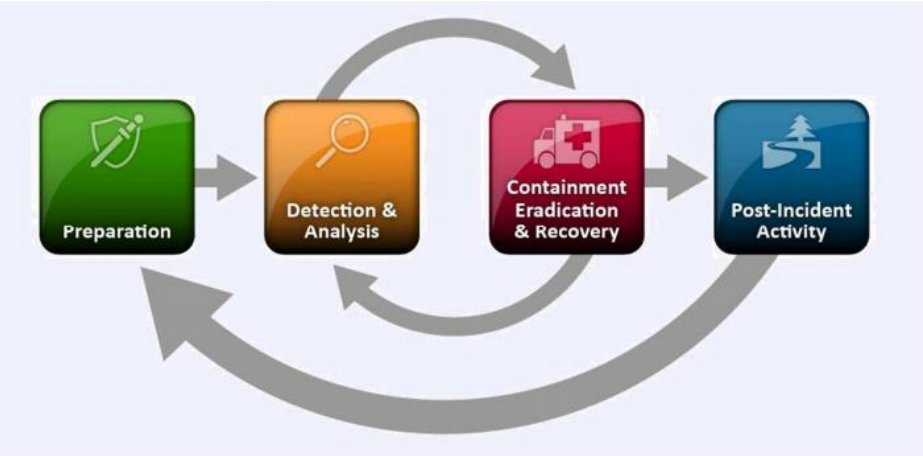



**Ilustración 48: Respuesta a Incidentes**
**Fuente:** (*COLECCIÓN DE SEGURIDAD FÍSICA NUCLEAR DEL OIEA Y PUBLICACIONES CONEXAS*, n.d.)

### Preparación

- Armar grupo de respuesta a incidentes el cual se encuentre capacitado
- Tener un centro de respuesta a incidentes externos (CERT/CSIRT)
- Identificación de Activos
- Implementación de políticas y procedimientos de seguridad
- Implementar uso de herramientas de seguridad, para las cuales también se deberá establecer políticas sobre su uso.
- En base a los casos de abuso realizados se deberá tener avisos, por ejemplo, en los puntos de acceso del sistema

**Detección y Análisis**

- Identificación de incidente
    - Signos Indicadores: Indica que un incidente ocurrió o está ocurriendo. Por ejemplo: Alerta de antivirus, accesos lentos, caída de servidor, etc.
    - Signos Precursores: Indica que un incidente puede ocurrir a futuro, por ejemplo: Escaneo de vulnerabilidades, análisis de riesgos.
- Identificar si el incidente encontrado es un falso positivo o no
- Categorizar incidente dependiendo del Nivel de Riesgo
- Implementar medidas provisionales hasta que se entre la solución que permita mitigar el incidente
- Identificar repercusiones y daños que tendrá el incidente tanto en la seguridad de la aplicación como en la información resguardada
- Identificar la causa raíz del incidente, como también la fuente del atacante, de esta manera que nos permita armar posibles perfiles.
- Notificación de Incidente de seguridad
- Registrar Incidente
- Determinar medidas de compensación
- Elaborar estrategia de mitigación

**Mitigación (Contención, Erradicación y recuperación)**

- Ya identificado el incidente, se deberá aplicar la solución para su mitigación, dependiendo del tipo de Incidente.
- Neutralizar componentes asociados al incidentes o nuevas incidenticas presentadas que se puedan presentar

- Implementar medidas de reconstrucción del sistema, para que el incidente presentado no vuelva a ocurrir y tener un sistema seguro
- Restablecer información y componentes afectados
- Evaluar daños coleteras.
- Monitorear proceso de mitigación

**Actividad posterior al incidente**

- Elaborar lista de acciones tomadas en la mitigación del incidente
- Presentar informes del incidente
- Evaluar recursos y activos, en caso de daño
- Evaluar Daños y costo
- Actualización de matriz de amenazas
- Revisión de plan de respuesta
- Mejora de solución a la mitigación en caso de haberla

## Anexo 8: Plan de Divulgación

Para la comunicación en la duración del o los proyectos se plantea 3 frentes:

- **Comunicación Interna:** La realizara los miembros que conforman la empresa (Gerente del Proyecto, Desarrolladores, Arquitecto, etc.).

- **Comunicación Externa:** Beneficiarios Directos y potenciales beneficiaros en base al resultado del proyecto.

Los medios de comunicación que se utilizaran son: Correo o celular para comunicación Interna o Externa.

## Anexo 9: Métricas

Para las métricas de seguridad se tomar en cuenta los siguientes factores:

- Significativa: Simple y fácil de entender
- Precisa: Proporcione información útil
- Rentable: No sea costa en adquirir y mantener
- Indicativa de resultados.

Se establece las siguientes métricas como métricas iniciales:

| Métrica | Indicador |
|---|---|
| Políticas de Seguridad | Nivel de cumplimiento de Políticas Aprobadas |
| Activos Críticos de la Organización | Nivel de afectación del activo de la organización en un ataque |
| Número de ataques detectados | Eficacia de medidas de seguridad implementadas |
| Coste de brechas de seguridad | Pérdidas económicas debido a fallos de seguridad |
| Recursos asignados a funciones de Seguridad | Coste económico utilizado en el programa de seguridad |
| Tiempo de respuesta a Incidencia | Eficacia con la que se mitiga una incidencia |

**Tabla 26 : Métricas generales**

## Anexo 10: Plantilla Hoja de Vida de Indicador

| Hoja de Vida de Indicador | | | |
|---|---|---|---|
| **ID** | | **Fecha Creación** | |
| **Fecha Actualización** | | **Versión** | |
| **Nombre** | | | |
| **Objetivo** | | | |
| | | | |
| **Tipo de Indicador** | | | |
| | | | |
| **Descripción de Variables** | **Fórmula de Cálculo** | **Fuente de Datos** | |
| | | | |
| **Metas** | | | |
| **No cumple** | 0% – 24 % | **Cumple mínimamente** | 25% – 49 % |
| **Cumple parcialmente** | 50% – 74% | **Cumple mayoritariamente** | 75% – 99% |
| **Cumple totalmente** | | 100% | |
| **Observaciones** | | | |
| | | | |

| Hoja de Vida de Indicador | | | |
|---|---|---|---|
| **ID** | | **Fecha creación** | |
| **Fecha Actualización** | | **Versión** | |
| **Nombre** | | | |
| **Objetivo** | | | |
| | | | |
| **Tipo de Indicador** | | | |
| | | | |
| **Descripción de Variables** | **Fórmula de Cálculo** | **Fuente de Datos** | |
| | | | |
| **Metas** | | | |
| **Minimiza** | **Satisfactoria** | **Excelente** | |
| 0%-79% | 80% - 99% | 100% | |
| **Observaciones** | | | |
| | | | |

**Tabla 27: Hoja de Vida de Indicador**

## Anexo 11: Formato de Reporte de Incidentes

| Reporte de Incidentes de Seguridad | | | | | |
|---|---|---|---|---|---|
| **Fecha del Incidente** | | | | | |
| **N° Incidente** | | | | | |
| **Descripción General del Incidente** | | | | | |
| **¿Qué ocurrió?** | | | | | |
| | | | | | |
| **¿Por qué ocurrió?** | | | | | |
| | | | | | |
| **¿Cómo ocurrió?** | | | | | |
| | | | | | |
| **Existe Compontes Afectados** | **Si** | | **No** | | |
| | **¿Cuál?** | | | | |
| **Se identifica Vulnerabilidad** | **Si** | | **No** | | |
| | **¿Cuál?** | | | | |
| **Estado del Incidente** | | | | | |
| **Sucediendo** | | **Sucedió** | | **Sucede Nuevamente** | |
| **Detalle del Incidente** | | | | | |
| **¿Fecha y Hora en la que sucedió el incidente?** | | | | | |
| **¿Fecha y Hora en la que se descubrió el incidente?** | | | | | |
| **¿Fecha y Hora en la que se reportó el incidente?** | | | | | |
| **¿La respuesta a este incidente se finalizó?** | SI | | NO | | |
| **¿En caso afirmativo, especifique cuánto tiempo duró el incidente (Días/Horas/Minutos)?** | | | | | |
| **Persona Responsable** | | | | | |

*Tabla 28: Reporte de Incidentes*

| Categorización de Incidente | | | |
|---|---|---|---|
| **Disponibilidad** | | | |
| Falla del hardware | | Falla del software | |
| Error de usuario | | Falta de personal | |
| Error de diseño | | | |
| ¿Otro? | | | |
| Especifique | | | |
| **Uso Inapropiado de Recursos** | | | |
| Violación de una política de uso aceptable de recursos | | Violación de medidas de seguridad | |
| ¿Otro? | | | |
| Especifique | | | |
| **Acceso no Autorizado** | | | |
| Acceso lógico o físico sin permiso a una red | | Acceso lógico o físico sin permiso a una aplicación | |
| Acceso lógico o físico sin permiso a los datos | | Acceso lógico o físico sin permiso a un Sistema | |
| Acceso lógico o físico sin permiso a otro recurso | | | |
| ¿Otro? | | | |
| Especifique | | | |
| **Ataques técnicos** | | | |
| Escaneo de Redes | | Intentos de acceso | |
| Aprovechamiento de Puertas traseras | | Denegación de Servicio | |
| ¿Otro? | | | |
| Especifique | | | |
| **Código Malicioso** | | | |
| Virus | | Gusano de Red | |
| Troyano | | Botnet | |
| Ataques combinados | | | |
| ¿Otro? | | | |
| Especifique | | | |
| **Desastres naturales** | | | |
| Terremoto | | Inundación | |
| Descarga Electromagnética | | Erupciones volcánicas | |
| ¿Otro? | | | |
| Especifique | | | |
| **Conflicto social** | | | |
| Disturbios | | Ataques Terroristas | |

| | | | |
|---|---|---|---|
| Guerras | | Corrupción | |
| ¿Otro? | | | |
| Especifique | | | |
| **Daño físico** | | | |
| Incendio | | Agua | |
| Electrostática | | Destrucción de Equipo | |
| Robo de Equipos | | Alteración de Equipo | |
| ¿Otro? | | | |
| Especifique | | | |
| **Falla en la infraestructura** | | | |
| Fallas en la Alimentación Eléctrica | | Falles en las Redes | |
| Fallas en el Aire Acondicionado | | Fallas en el suministro de Agua | |
| ¿Otro? | | | |
| Especifique | | | |
| **Fallas Eléctricas** | | | |
| Radiación Electromagnética | | Pulsos Electromagnéticos | |
| Interferencia Electrónica | | Fluctuación de Tensión | |
| Radicación Térmica | | | |
| ¿Otro? | | | |
| Especifique | | | |

**Tabla 29: Categorización del Incidente**

| Detalle de la Solución del Incidente | |
|---|---|
| **¿Fecha y Hora de revisión del incidente?** | |
| **¿Responsable/s revisión del incidente?** | |
| **¿Fecha y Hora de la Finalización del incidente?** | |
| **¿Fecha y Hora de la Finalización del impacto?** | |
| **En caso de Resolución del Incidente** | |
| **Descripción de las acciones planeadas para resolver el incidente** | |
| | |
| **Descripción de las acciones tomadas para resolver el incidente** | |
| | |
| **Acciones pendientes para resolver el incidente** | |
| | |
| **Conclusiones** | |
| | |
| **Observaciones (Lesiones Aprendidas)** | |
| | |

**Tabla 30: Detalle Solución del Incidente**

## Anexo 12: Plantilla de Casos de Abuso

| Identificador | CA -1 |
|---|---|
| **Nombre del Caso de Abuso** | Nombre del caso de Uso - Abuso claro y preciso<br>Debe comenzar con un verbo |
| **Fecha de Creación** | Fecha de Creación del Caso de Abuso |
| **Descripción** | Se dará una descripción del caso, la cual debe ser corta y concreta. |
| **Fase** | Fase de desarrollo de casos de abuso - Fase 3 (Arquitectura e Infraestructura) |
| **Actores** | Personas que interactúan directamente con el sistema |
| **Escenario Funcional** | Acciones que realiza el usuario en el sistema |
| **Escenario Negativo** | Acciones que realiza el atacante |

**Tabla 31: Plantilla de Casos de Abuso**

## Anexo 13: Plantillas de Gestión de Riesgos

| Identificación del riesgo | | | | | |
|---|---|---|---|---|---|
| **N.º DE RIESGO** | RI -1 | | | | |
| **RIESGO** | | | | | |
| **CAUSAS** | | | | | |
| **CONSECUENCIAS** | | | | | |
| **PROBABILIDAD** | | | | | |
| **Calificación del riesgo** | | | | | |
| **PROBABILIDAD** | 0 -2 | | 3-5 | | 6-9 |
| **IMPACTO** | Bajo | | Medio | | Alto |
| **EVALUACION DEL RIESGO** | Riesgo = Impacto * Probabilidad | | | | |
| **SEVERIDAD** | Ninguno | Bajo | Medio | Alto | Crítico |
| **Observaciones** | | | | | |
| | | | | | |

**Tabla 32: Plantilla de Gestión de Riesgo**

## Anexo 14: Escaneo de Vulnerabilidades

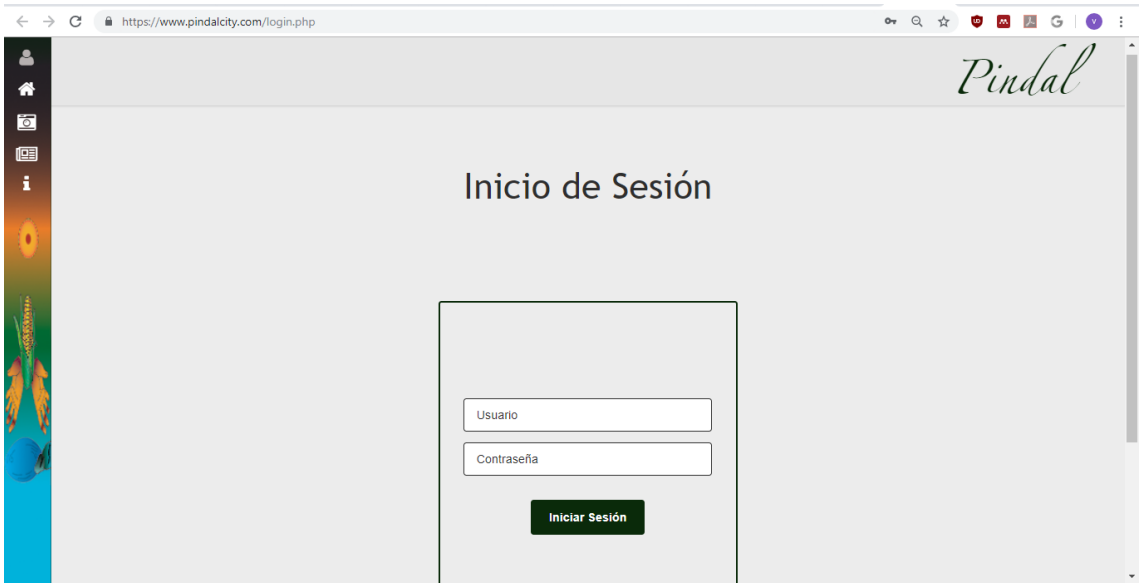
Un sistema debe poseer una arquitectura de seguridad adecuada, volviéndose un pilar fundamental de la aplicación. Esta debe contar con características, controles, procesos de seguridad que nos permitan reducir el riesgo de ataques. Como se observó en los resultados presentados en los anteriores capítulos, entre los mayores ataques se tiene SQL Inyección, perdida de Autenticación y gestión de sesiones, como también ataques de fuerza bruta y phishing. Ataques que aprovechan las vulnerabilidades de las aplicaciones.

A continuación, se presenta el escaneo de la aplicación web por medio de la herramienta de ZAP OWASP 2.7.0. La cual nos permite identificar las vulnerabilidades que posee la aplicación, presentando los siguientes niveles de alertas

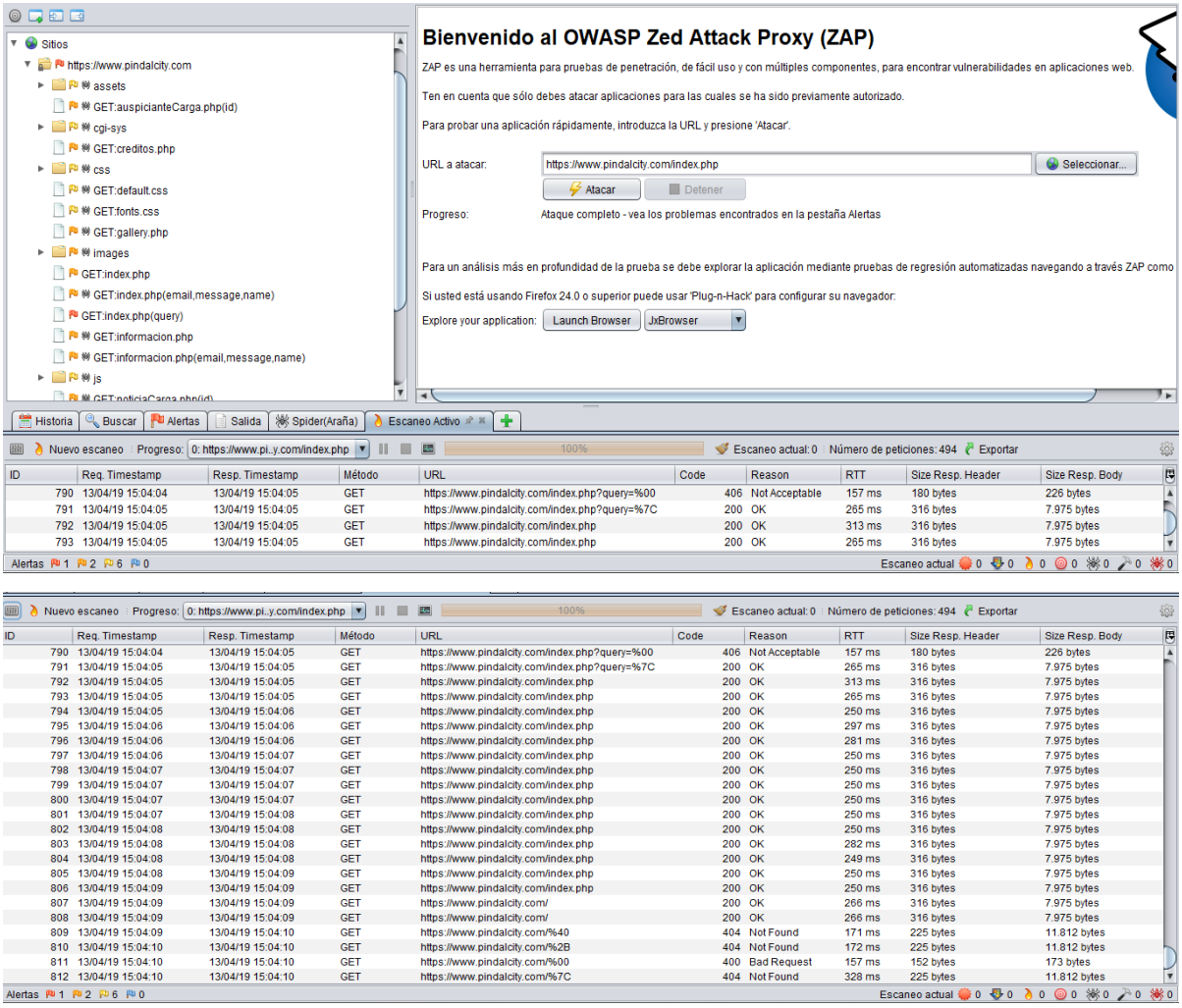



**Ilustración 49: Niveles de Alertas Herramienta ZAP OWASP**

**Página a Escanear**

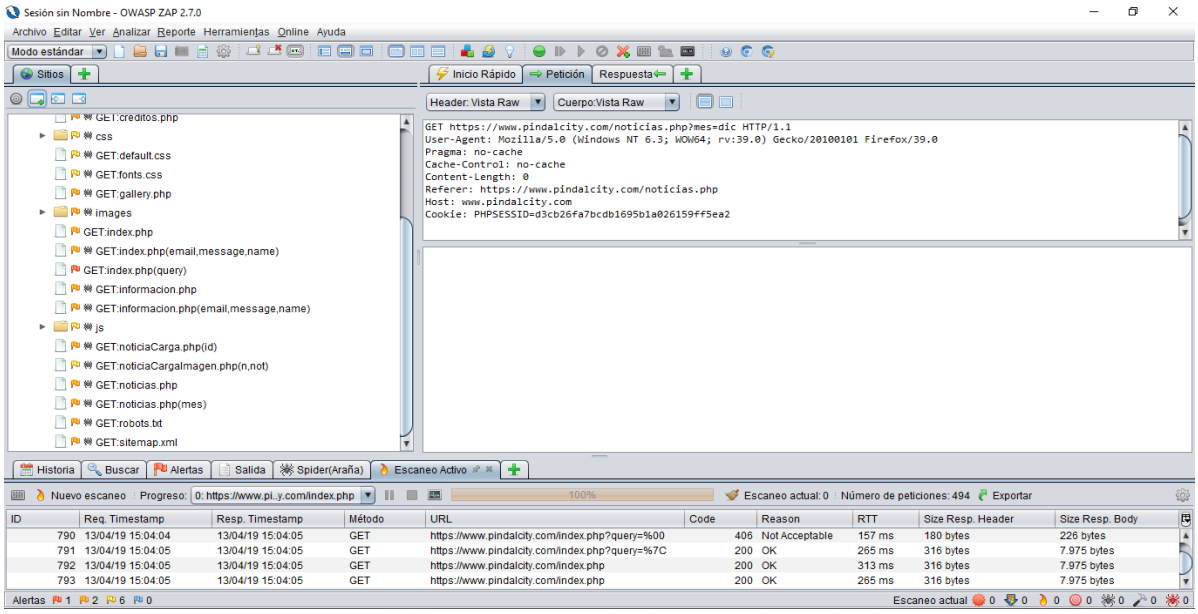


## Inicio

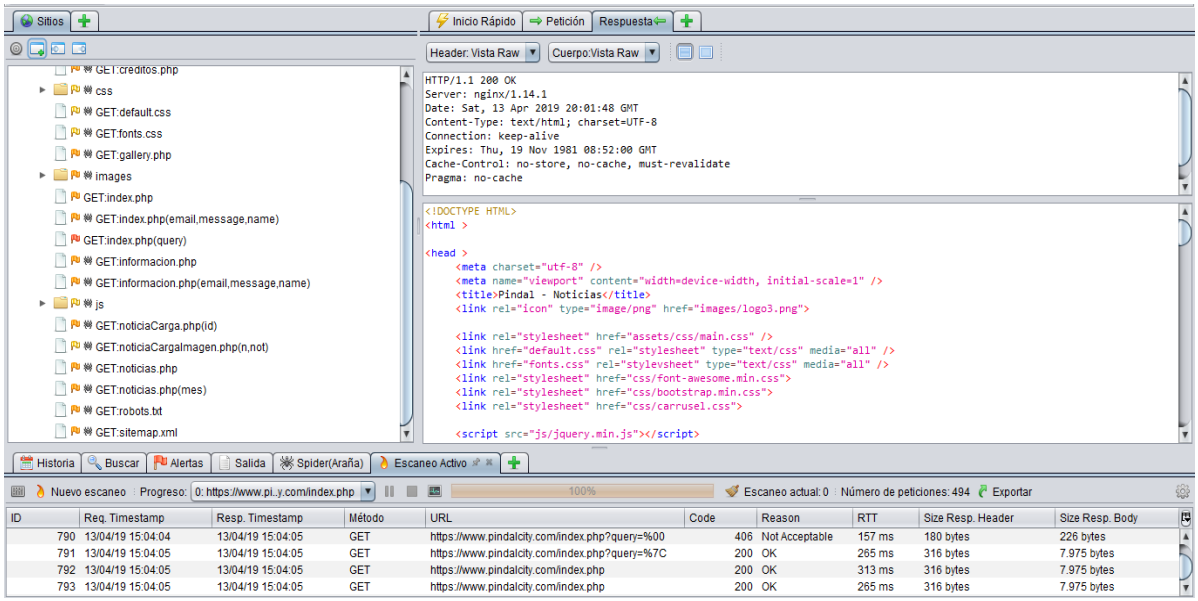

Sitios
https://www.pindalcity.com
assets
GET:auspicianteCarga.php(id)
cgi-sys
GET:creditos.php
css
GET:default.css
GET:fonts.css
GET:gallery.php
images
GET:index.php
GET:index.php(email,message,name)
GET:index.php(query)
GET:informacion.php
GET:informacion.php(email,message,name)
js
Bienvenido al OWASP Zed Attack Proxy (ZAP)
ZAP es una herramienta para pruebas de penetración, de fácil uso y con múltiples componentes, para encontrar vulnerabilidades en aplicaciones web.
Ten en cuenta que sólo debes atacar aplicaciones para las cuales se ha sido previamente autorizado.
Para probar una aplicación rápidamente, introduzca la URL y presione 'Atacar'.
URL a atacar:
https://www.pindalcity.com/index.php
Seleccionar...
Atacar
Detener
Progreso:
Ataque completo - vea los problemas encontrados en la pestaña Alertas
Para un análisis más en profundidad de la prueba se debe explorar la aplicación mediante pruebas de regresión automatizadas navegando a través ZAP como
Si usted está usando Firefox 24.0 o superior puede usar 'Plug-n-Hack' para configurar su navegador:
Explore your application:
Launch Browser
JxBrowser
Historia
Buscar
Alertas
Salida
Spider(Araña)
Escaneo Activo
Nuevo escaneo
Progreso:
0: https://www.pi..y.com/index.php
100%
Escaneo actual: 0
Número de peticiones: 494
Exportar
ID
Req. Timestamp
Resp. Timestamp
Método
URL
Code
Reason
RTT
Size Resp. Header
Size Resp. Body
790 13/04/19 15:04:04 13/04/19 15:04:05 GET https://www.pindalcity.com/index.php?query=%00 406 Not Acceptable 157 ms 180 bytes 226 bytes
791 13/04/19 15:04:05 13/04/19 15:04:05 GET https://www.pindalcity.com/index.php?query=%7C 200 OK 265 ms 316 bytes 7.975 bytes
792 13/04/19 15:04:05 13/04/19 15:04:05 GET https://www.pindalcity.com/index.php 200 OK 313 ms 316 bytes 7.975 bytes
793 13/04/19 15:04:05 13/04/19 15:04:05 GET https://www.pindalcity.com/index.php 200 OK 265 ms 316 bytes 7.975 bytes
Alertas 1 2 6 0
Escaneo actual 0 0 0 0 0 0 0

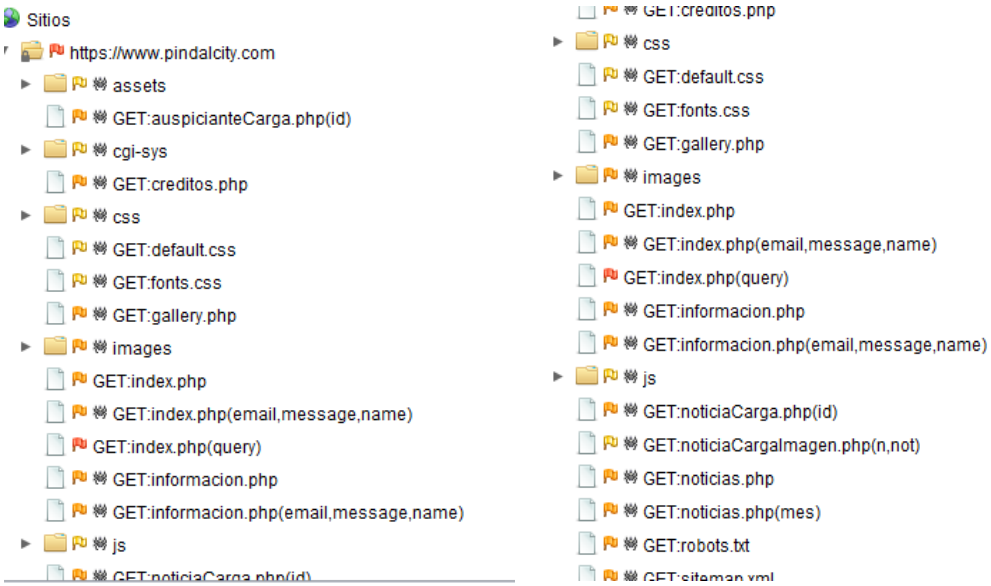

Nuevo escaneo
Progreso:
0: https://www.pi..y.com/index.php
100%
Escaneo actual: 0
Número de peticiones: 494
Exportar
ID
Req. Timestamp
Resp. Timestamp
Método
URL
Code
Reason
RTT
Size Resp. Header
Size Resp. Body
790 13/04/19 15:04:04 13/04/19 15:04:05 GET https://www.pindalcity.com/index.php?query=%00 406 Not Acceptable 157 ms 180 bytes 226 bytes
791 13/04/19 15:04:05 13/04/19 15:04:05 GET https://www.pindalcity.com/index.php?query=%7C 200 OK 265 ms 316 bytes 7.975 bytes
792 13/04/19 15:04:05 13/04/19 15:04:05 GET https://www.pindalcity.com/index.php 200 OK 313 ms 316 bytes 7.975 bytes
793 13/04/19 15:04:05 13/04/19 15:04:05 GET https://www.pindalcity.com/index.php 200 OK 265 ms 316 bytes 7.975 bytes
794 13/04/19 15:04:05 13/04/19 15:04:06 GET https://www.pindalcity.com/index.php 200 OK 250 ms 316 bytes 7.975 bytes
795 13/04/19 15:04:06 13/04/19 15:04:06 GET https://www.pindalcity.com/index.php 200 OK 297 ms 316 bytes 7.975 bytes
796 13/04/19 15:04:06 13/04/19 15:04:06 GET https://www.pindalcity.com/index.php 200 OK 281 ms 316 bytes 7.975 bytes
797 13/04/19 15:04:06 13/04/19 15:04:07 GET https://www.pindalcity.com/index.php 200 OK 250 ms 316 bytes 7.975 bytes
798 13/04/19 15:04:07 13/04/19 15:04:07 GET https://www.pindalcity.com/index.php 200 OK 250 ms 316 bytes 7.975 bytes
799 13/04/19 15:04:07 13/04/19 15:04:07 GET https://www.pindalcity.com/index.php 200 OK 250 ms 316 bytes 7.975 bytes
800 13/04/19 15:04:07 13/04/19 15:04:07 GET https://www.pindalcity.com/index.php 200 OK 250 ms 316 bytes 7.975 bytes
801 13/04/19 15:04:07 13/04/19 15:04:08 GET https://www.pindalcity.com/index.php 200 OK 250 ms 316 bytes 7.975 bytes
802 13/04/19 15:04:08 13/04/19 15:04:08 GET https://www.pindalcity.com/index.php 200 OK 250 ms 316 bytes 7.975 bytes
803 13/04/19 15:04:08 13/04/19 15:04:08 GET https://www.pindalcity.com/index.php 200 OK 282 ms 316 bytes 7.975 bytes
804 13/04/19 15:04:08 13/04/19 15:04:08 GET https://www.pindalcity.com/index.php 200 OK 249 ms 316 bytes 7.975 bytes
805 13/04/19 15:04:08 13/04/19 15:04:09 GET https://www.pindalcity.com/index.php 200 OK 250 ms 316 bytes 7.975 bytes
806 13/04/19 15:04:09 13/04/19 15:04:09 GET https://www.pindalcity.com/index.php 200 OK 250 ms 316 bytes 7.975 bytes
807 13/04/19 15:04:09 13/04/19 15:04:09 GET https://www.pindalcity.com/ 200 OK 266 ms 316 bytes 7.975 bytes
808 13/04/19 15:04:09 13/04/19 15:04:09 GET https://www.pindalcity.com/ 200 OK 266 ms 316 bytes 7.975 bytes
809 13/04/19 15:04:09 13/04/19 15:04:10 GET https://www.pindalcity.com/%40 404 Not Found 171 ms 225 bytes 11.812 bytes
810 13/04/19 15:04:10 13/04/19 15:04:10 GET https://www.pindalcity.com/%2B 404 Not Found 172 ms 225 bytes 11.812 bytes
811 13/04/19 15:04:10 13/04/19 15:04:10 GET https://www.pindalcity.com/%00 400 Bad Request 157 ms 152 bytes 173 bytes
812 13/04/19 15:04:10 13/04/19 15:04:10 GET https://www.pindalcity.com/%7C 404 Not Found 328 ms 225 bytes 11.812 bytes
Alertas 1 2 6 0
Escaneo actual 0 0 0 0 0 0 0

## Petición

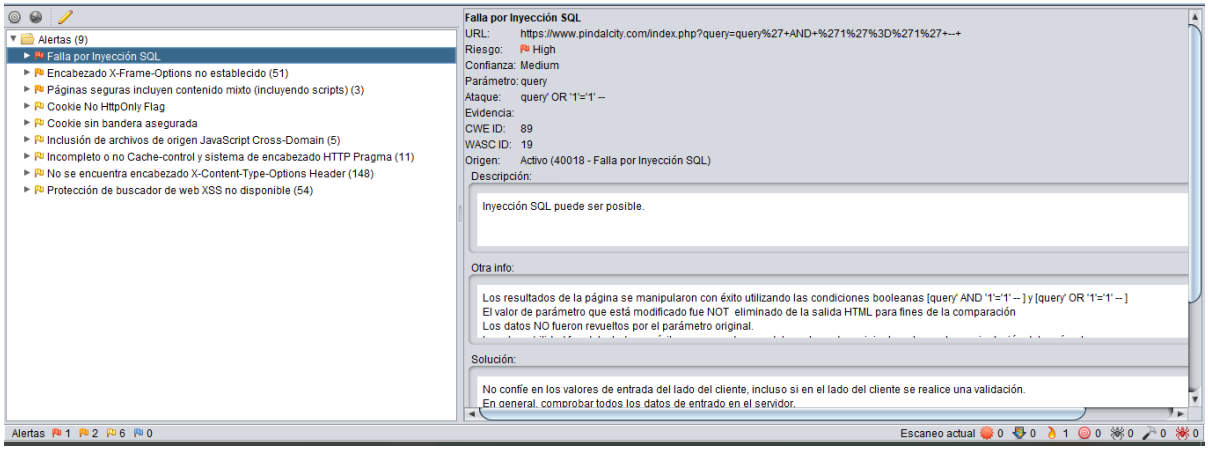


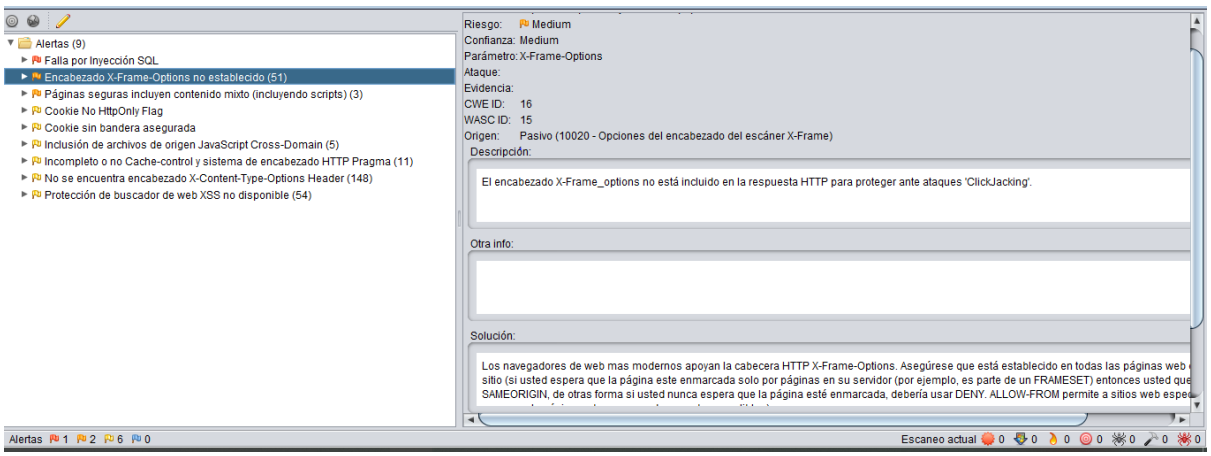

## Respuesta

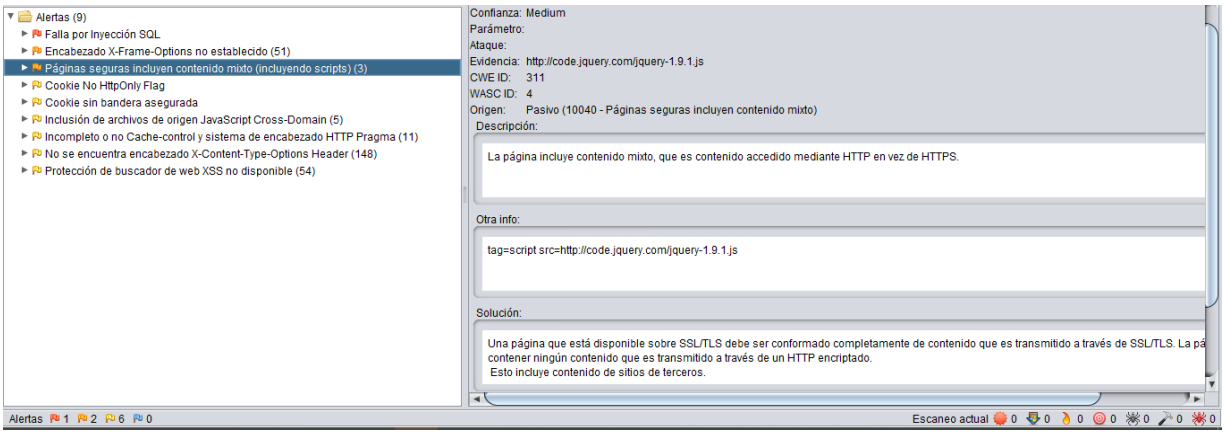


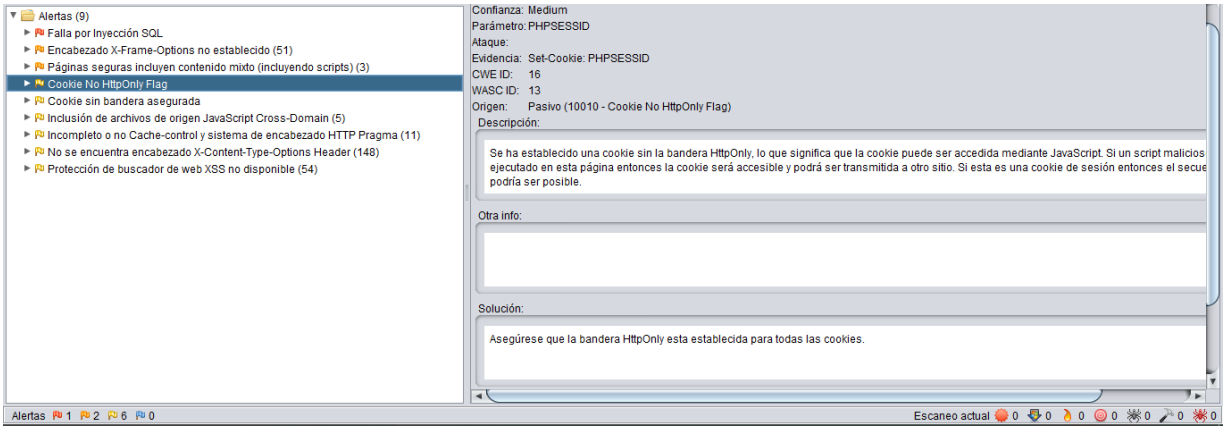

Sitios
https://www.pindalcity.com
assets
GET:auspicianteCarga.php(id)
cgi-sys
GET:creditos.php
css
GET:default.css
GET:fonts.css
GET:gallery.php
images
GET:index.php
GET:index.php(email,message,name)
GET:index.php(query)
GET:informacion.php
GET:informacion.php(email,message,name)
js
GET:noticiaCarga.php(id)
GET:noticiaCargaImagen.php(n,not)
GET:noticias.php
GET:noticias.php(mes)
GET:robots.txt

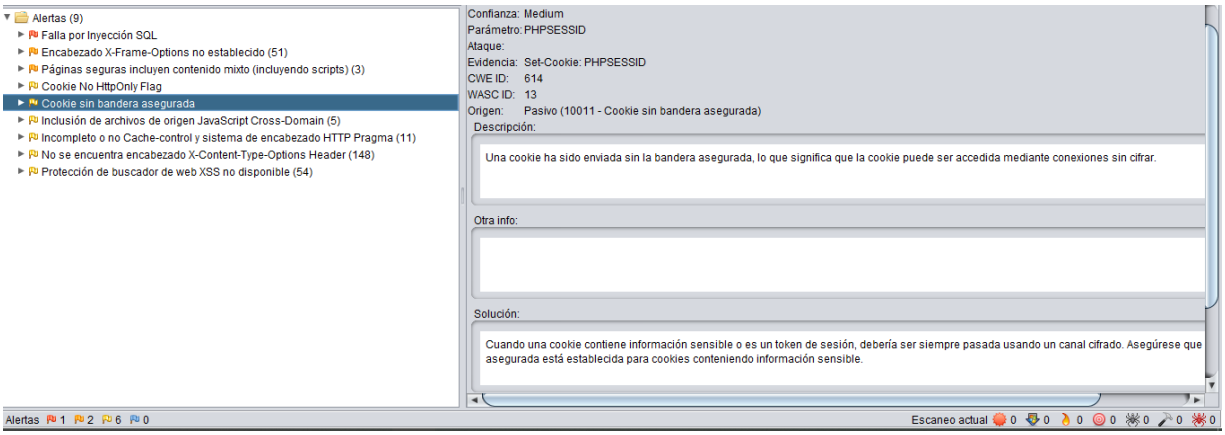
## Alertas

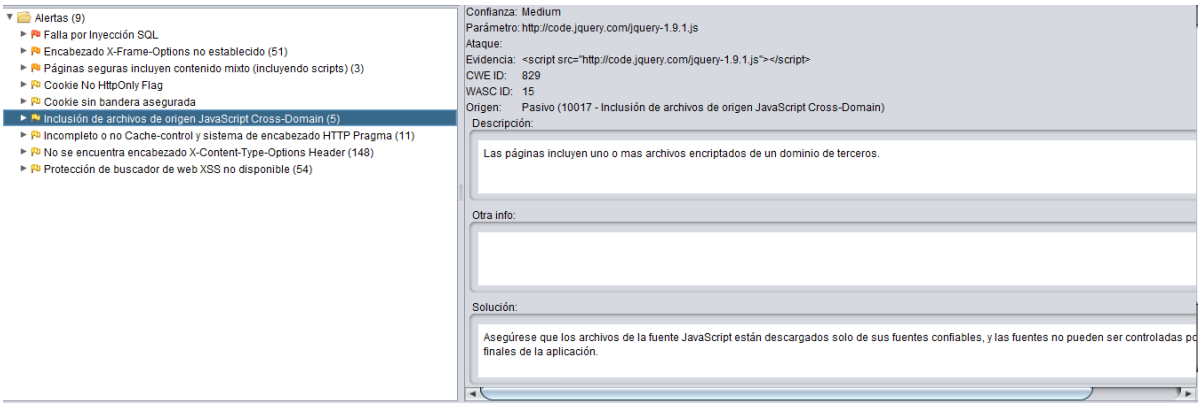

Alertas (9)
Falla por Inyección SQL
Encabezado X-Frame-Options no establecido (51)
Páginas seguras incluyen contenido mixto (incluyendo scripts) (3)
Cookie No HttpOnly Flag
Cookie sin bandera asegurada
Inclusión de archivos de origen JavaScript Cross-Domain (5)
Incompleto o no Cache-control y sistema de encabezado HTTP Pragma (11)
No se encuentra encabezado X-Content-Type-Options Header (148)
Protección de buscador de web XSS no disponible (54)
Falla por Inyección SQL
URL: https://www.pindalcity.com/index.php?query=query%27+AND+%271%27%3D%271%27+--+
Riesgo: High
Confianza: Medium
Parámetro: query
Ataque: query' OR '1'='1' --
Evidencia:
CWE ID: 89
WASC ID: 19
Origen: Activo (40018 - Falla por Inyección SQL)
Descripción:
Inyección SQL puede ser posible.
Otra info:
Los resultados de la página se manipularon con éxito utilizando las condiciones booleanas [query' AND '1'='1' -- ] y [query' OR '1'='1' -- ]
El valor de parámetro que está modificado fue NOT eliminado de la salida HTML para fines de la comparación
Los datos NO fueron revueltos por el parámetro original.
Solución:
No confíe en los valores de entrada del lado del cliente, incluso si en el lado del cliente se realice una validación.
Alertas 1 2 6 0
Escaneo actual 0 0 0 1 0 0 0 0 0

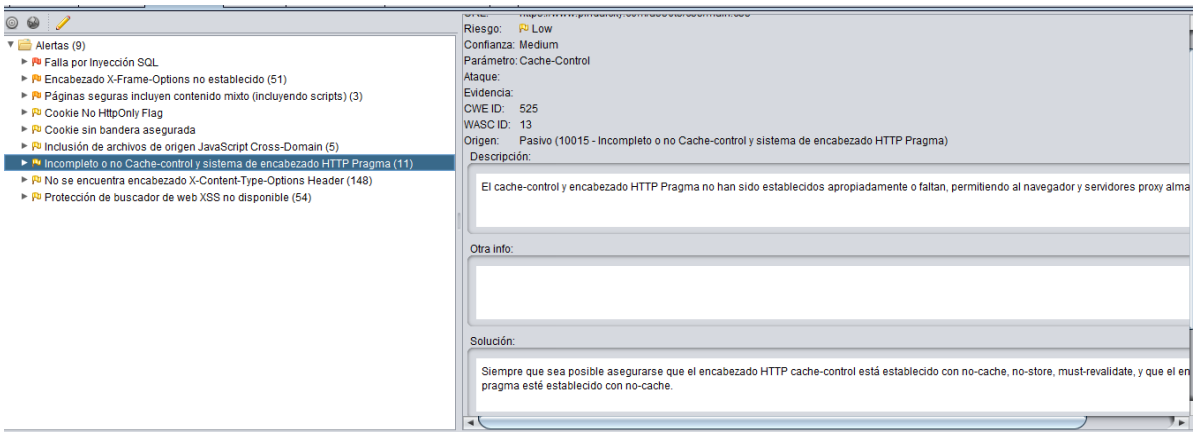

Alertas (9)
Falla por Inyección SQL
Encabezado X-Frame-Options no establecido (51)
Páginas seguras incluyen contenido mixto (incluyendo scripts) (3)
Cookie No HttpOnly Flag
Cookie sin bandera asegurada
Inclusión de archivos de origen JavaScript Cross-Domain (5)
Incompleto o no Cache-control y sistema de encabezado HTTP Pragma (11)
No se encuentra encabezado X-Content-Type-Options Header (148)
Protección de buscador de web XSS no disponible (54)
Riesgo: Medium
Confianza: Medium
Parámetro: X-Frame-Options
Ataque:
Evidencia:
CWE ID: 16
WASC ID: 15
Origen: Pasivo (10020 - Opciones del encabezado del escáner X-Frame)
Descripción:
El encabezado X-Frame_options no está incluido en la respuesta HTTP para proteger ante ataques 'ClickJacking'.
Otra info:
Solución:
Los navegadores de web mas modernos apoyan la cabecera HTTP X-Frame-Options. Asegúrese que está establecido en todas las páginas web
sitio (si usted espera que la página este enmarcada solo por páginas en su servidor (por ejemplo, es parte de un FRAMESET) entonces usted que
SAMEORIGIN, de otras forma si usted nunca espera que la página esté enmarcada, debería usar DENY. ALLOW-FROM permite a sitios web espe
Alertas 1 2 6 0
Escaneo actual 0 0 0 0 0 0 0 0 0

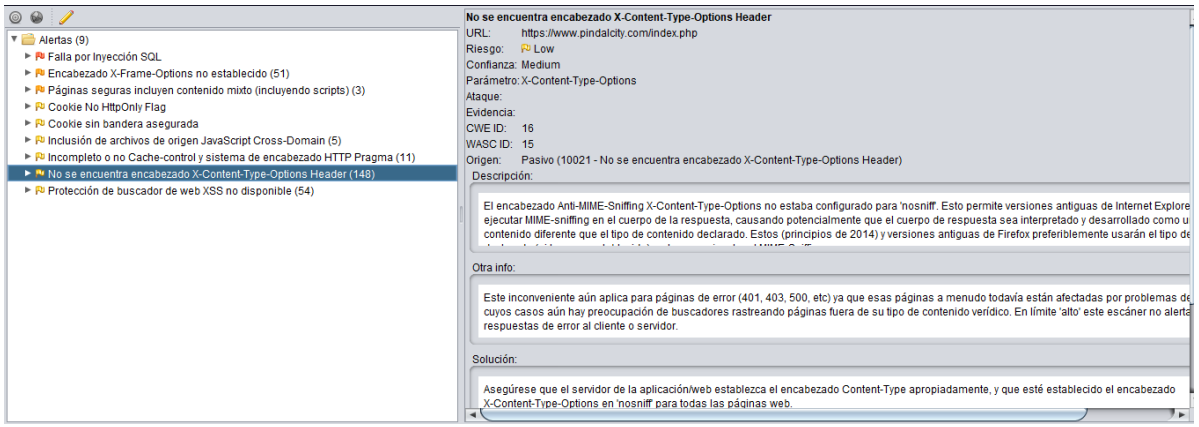

Alertas (9)
Falla por Inyección SQL
Encabezado X-Frame-Options no establecido (51)
Páginas seguras incluyen contenido mixto (incluyendo scripts) (3)
Cookie No HttpOnly Flag
Cookie sin bandera asegurada
Inclusión de archivos de origen JavaScript Cross-Domain (5)
Incompleto o no Cache-control y sistema de encabezado HTTP Pragma (11)
No se encuentra encabezado X-Content-Type-Options Header (148)
Protección de buscador de web XSS no disponible (54)
Confianza: Medium
Parámetro:
Ataque:
Evidencia: http://code.jquery.com/jquery-1.9.1.js
CWE ID: 311
WASC ID: 4
Origen: Pasivo (10040 - Páginas seguras incluyen contenido mixto)
Descripción:
La página incluye contenido mixto, que es contenido accedido mediante HTTP en vez de HTTPS.
Otra info:
tag=script src=http://code.jquery.com/jquery-1.9.1.js
Solución:
Una página que está disponible sobre SSL/TLS debe ser conformado completamente de contenido que es transmitido a través de SSL/TLS. La pá
contener ningún contenido que es transmitido a través de un HTTP encriptado.
Esto incluye contenido de sitios de terceros.
Alertas 1 2 6 0
Escaneo actual 0 0 0 0 0 0 0

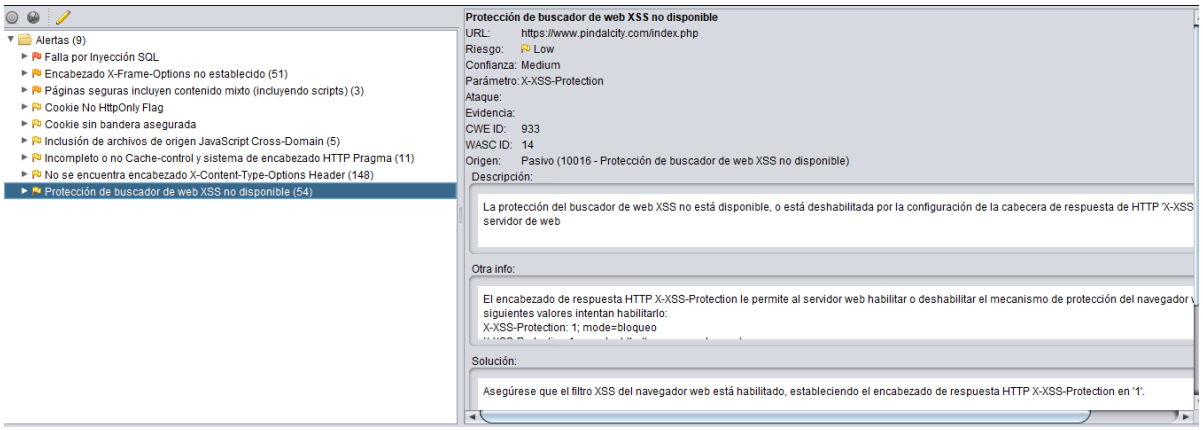

Alertas (9)
Falla por Inyección SQL
Encabezado X-Frame-Options no establecido (51)
Páginas seguras incluyen contenido mixto (incluyendo scripts) (3)
Cookie No HttpOnly Flag
Cookie sin bandera asegurada
Inclusión de archivos de origen JavaScript Cross-Domain (5)
Incompleto o no Cache-control y sistema de encabezado HTTP Pragma (11)
No se encuentra encabezado X-Content-Type-Options Header (148)
Protección de buscador de web XSS no disponible (54)
Confianza: Medium
Parámetro: PHPSESSID
Ataque:
Evidencia: Set-Cookie: PHPSESSID
CWE ID: 16
WASC ID: 13
Origen: Pasivo (10010 - Cookie No HttpOnly Flag)
Descripción:
Se ha establecido una cookie sin la bandera HttpOnly, lo que significa que la cookie puede ser accedida mediante JavaScript. Si un script malicios
ejecutado en esta página entonces la cookie será accesible y podrá ser transmitida a otro sitio. Si esta es una cookie de sesión entonces el secue
podría ser posible.
Otra info:
Solución:
Asegúrese que la bandera HttpOnly esta establecida para todas las cookies.
Alertas 1 2 6 0
Escaneo actual 0 0 0 0 0 0 0


Alertas (9)
Falla por Inyección SQL
Encabezado X-Frame-Options no establecido (51)
Páginas seguras incluyen contenido mixto (incluyendo scripts) (3)
Cookie No HttpOnly Flag
Cookie sin bandera asegurada
Inclusión de archivos de origen JavaScript Cross-Domain (5)
Incompleto o no Cache-control y sistema de encabezado HTTP Pragma (11)
No se encuentra encabezado X-Content-Type-Options Header (148)
Protección de buscador de web XSS no disponible (54)
Confianza: Medium
Parámetro: PHPSESSID
Ataque:
Evidencia: Set-Cookie: PHPSESSID
CWE ID: 614
WASC ID: 13
Origen: Pasivo (10011 - Cookie sin bandera asegurada)
Descripción:
Una cookie ha sido enviada sin la bandera asegurada, lo que significa que la cookie puede ser accedida mediante conexiones sin cifrar.
Otra info:
Solución:
Cuando una cookie contiene información sensible o es un token de sesión, debería ser siempre pasada usando un canal cifrado. Asegúrese que
asegurada está establecida para cookies conteniendo información sensible.
Alertas 1 2 6 0
Escaneo actual 0 0 0 0 0 0 0


Alertas (9)
Falla por Inyección SQL
Encabezado X-Frame-Options no establecido (51)
Páginas seguras incluyen contenido mixto (incluyendo scripts) (3)
Cookie No HttpOnly Flag
Cookie sin bandera asegurada
Inclusión de archivos de origen JavaScript Cross-Domain (5)
Incompleto o no Cache-control y sistema de encabezado HTTP Pragma (11)
No se encuentra encabezado X-Content-Type-Options Header (148)
Protección de buscador de web XSS no disponible (54)
Confianza: Medium
Parámetro: http://code.jquery.com/jquery-1.9.1.js
Ataque:
Evidencia: <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
CWE ID: 829
WASC ID: 15
Origen: Pasivo (10017 - Inclusión de archivos de origen JavaScript Cross-Domain)
Descripción:
Las páginas incluyen uno o mas archivos encriptados de un dominio de terceros.
Otra info:
Solución:
Asegúrese que los archivos de la fuente JavaScript están descargados solo de sus fuentes confiables, y las fuentes no pueden ser controladas p
finales de la aplicación.




## Análisis

En conclusión, al escaneo se observó que la aplicación es vulnerable a ataques debido a que no se realizó una implementación de seguridad adecuada, lo que le permitirá al atacante ingresar a la información fácilmente, en alertas de nivel alto posee 1, de nivel medio 2 y de nivel bajo 6.

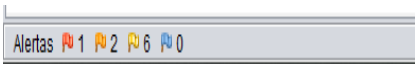



***Ilustración 50: Alertas de la aplicación "Pindal"***

## BIBLIOGRAFÍA


Adriana Matei. (n.d.). *Guía para el desarrollo de Software Seguro*. Retrieved from http://oa.upm.es/34770/1/PFC_ADRIANA_MATEI.pdf

Alberto, J., García, B., Perla, D., & Velasco Elizondo, I. (2010). *Incorporación de prácticas de seguridad de software en TSPi Que para obtener el grado de*. Retrieved from https://cimat.repositorioinstitucional.mx/jspui/bitstream/1008/406/1/ZACTE9.pdf

Aldana, L. A. F. (2009). *Transmisión y comunicación de datos*. (E. C. Edi, Ed.).

Caro, A. (n.d.). *MODELOS DE DESARROLLO SEGURO DE SOFTWARE*. Retrieved from http://map.norsecorp.com

Castañeda, P. (n.d.). *Introducción a las Metodologías de Desarrollo de Software: 1-¿Qué es una metodología de desarrollo de software?* Retrieved from https://pid.dsic.upv.es/C1/Material/Documentos Disponibles/Introducción Proceso

Castellaro, M., Romaniz, S., Ramos, J., & Pessolani, P. (n.d.). *Hacia la Ingeniería de Software Seguro*. Retrieved from http://sedici.unlp.edu.ar/bitstream/handle/10915/21332/Documento_completo.pdf?sequence=1&isAllowed=y

*COLECCIÓN DE SEGURIDAD FÍSICA NUCLEAR DEL OIEA Y PUBLICACIONES CONEXAS*. (n.d.). Retrieved from https://www-pub.iaea.org/MTCD/Publications/PDF/TDL005_S_web.pdf

Congreso, de las Naciones Unidas, sobre Prevención del Delito, & y Justicia Penal. (2010). Novedades recientes en el uso de la ciencia y la tecnología por los delincuentes y por las autoridades competentes en la lucha contra la delincuencia, incluido el delito cibernético. *Computerkriminalität Und Strafrecht*. Retrieved from https://www.unodc.org/documents/crime-congress/12th-Crime-Congress/Documents/A_CONF.213_9/V1050385s.pdf

De, D. E. C. (2015). ¿Qué entendemos por ciberseguridad? 1.

De Evaluación, P., & Revistas, D. E. (2017). Tecnología. https://doi.org/10.17993/3ctecno.2017.v6n3e23

Deleersnyder, S., Win, B. De, Glas, B., Arciniegas, F., Bartoldus, M., Carter, J., … Glas, B. (n.d.). Software Assurance Maturity Model.

Deleersnyder, S., Win, B. De, Glas, B., Arciniegas, F., Bartoldus, M., Carter, J., … Glas, B. (2017). Software Assurance Maturity Model.

Enrique Jaziel Ulloa Castillón. (n.d.). *Tesis para obtener el grado de Maestro en Tecnologías de la Información*. Retrieved from http://mti.cucea.udg.mx/sites/default/files/Investigacion-Enrique-Ulloa.pdf

Ericka Jamin Robles Gomez. (2011). *Ciclo de vida de desarrollo de Software Seguro en Metodologías Ágiles*. Retrieved from https://cimat.repositorioinstitucional.mx/jspui/bitstream/1008/411/1/ZACTE14.pdf

Estadísticas | UNAM-CERT. (2017). Retrieved May 29, 2018, from https://www.cert.org.mx/estadisticas

*Estándar de Verificación de Seguridad en*. (2017). Retrieved from https://www.owasp.org/images/a/aa/Estándar_de_Verificación_de_Seguridad_en_Aplicaciones_3.0.1.pdf

FireEye. (2018). Endpoint Forensics: investigue los puntos finales en tiempo real | FireEye. Retrieved May 24, 2018, from https://www.fireeye.com/solutions/mir-endpoint-forensics.html

Foundation, T. O., Commons, C., & Sharealike, A. (2010). OWASP Secure Coding Practices Quick Reference Guide, (November), 1–17.

Gerónimo, A. F. (2009). *Modelo OSI*. (E. C. Edi, Ed.).

Giant, N. (2016). *Ciberseguridad para la i-Generación*. (N. Ediciones, Ed.).

Graham, G. (2013). White Paper Microsoft Security Development Lifecycle Adoption : Why and How, (September).

Gregg, M., Watkins, S., Mays, G., Ries, C., Bandes, R., & Franklin, B. (2006). *Hack the Stack*. *Hack the Stack*. https://doi.org/10.1016/B978-1-59749-109-9.X5000-X

Guamán Quinche, R. (n.d.). *Seguridad en Entornos Web para Sistemas de Gestión Académica*. Retrieved from http://www.owasp.org

*GUÍA DE PRUEBAS OWASP 2008 V3.0*. (n.d.). Retrieved from https://www.owasp.org/images/8/80/Guía_de_pruebas_de_OWASP_ver_3.0.pdf

*GUÍA DE PRUEBAS OWASP 2008 V3.0*. (2002). Retrieved from https://www.owasp.org/images/8/80/Guía_de_pruebas_de_OWASP_ver_3.0.pdf

Guide, H. O. W. T. O. (2017). SOFTWARE ASSURANCE MATURITY MODEL, (Cc).

Hackers, L. O. S. (2014). Seguridad informática, 301.

Hillar, G. C. (2004). *Redes*. (E. H. A. HASA, Ed.).

IBM. (n.d.). IBM developerWorks en español : Introducción a SOA y servicios web. Retrieved September 25, 2017, from https://www.ibm.com/developerworks/ssa/webservices/newto/service.html

Ii, P., & Cisco, A. (2018). Annual Cybersecurity Report.

İlgün, İ. (n.d.). *Comparison of SDLC and OWASP*. Retrieved from https://courses.cs.ut.ee/MTAT.03.246/2013_spring/uploads/Main/essay10.pdf

Incibe. (2015). Las 30 mayores vulnerabilidades explotadas por los ciberatacantes | INCIBE. Retrieved May 9, 2018, from https://www.incibe.es/protege-tu-empresa/blog/30-mayores-vulnerabilidades

Incibe. (2017). Amenaza vs Vulnerabilidad, ¿sabes en qué se diferencian? | INCIBE. Retrieved May 9, 2018, from https://www.incibe.es/protege-tu-empresa/blog/amenaza-vs-vulnerabilidad-sabes-se-diferencian

INEC. (2016). Retrieved March 13, 2018, from http://www.ecuadorencifras.gob.ec/estadisticas/

Inteco. (2014). Estudio sobre la seguridad de la información y e-confianza en los hogares españoles. Retrieved from http://www.ontsi.red.es/ontsi/sites/ontsi/files/ciberseguridad_y_confianza_en_los_hogares.pdf

Internet World Stats. (2017). World Internet Users Statistics and 2018 World Population Stats. Retrieved March 13, 2018, from https://www.internetworldstats.com/stats.htm

ISACA. (2015). State of Cybersecurity : Implications for 2015. *CyberSecurity Nexus*, 22.

Isaca, A., & Survey, R. S. A. C. (2016). State of Cybersecurity Implications for 2016 The State of Cybersecurity.

Jordá Capitán, E., Priego Fernández, V. de, & Sádaba, C. (2014). *La protección y seguridad de la persona en Internet: aspectos sociales y jurídicos*. (Editorial Reus, Ed.).

Lopéz, A. A. (2005). *La red Internet. El modelo TCP/IP*. (G. A. F. y Consultoría, Ed.).

Manzuik, S., Pfeil, K., & Gold, A. (2007). *Network Security Assessment*.

Marañón, G. Á., & García, P. P. P. (2004). *Seguridad informática para empresas y particulares*. (M.-H. España, Ed.).

Marulanda, C. L., & Ceballos, J. H. (2012). *UNA REVISIÓN DE METODOLOGÍAS SEGURAS EN CADA FASE DEL CICLO DE VIDA DEL DESARROLLO DE SOFTWARE A REVIEW OF SECURE METHODOLOGIES FOR ALL THE STAGES OF LIFE CYCLE OF SOFTWARE DEVELOPMENT*. *Ing. USBMed* (Vol. 3). Enero-Junio. Retrieved from http://web.usbmed.edu.co/usbmed/fing/v3n1/v3n1a2.pdf

Merinero, J. M. (2010). *TRABAJO FIN DE CARRERA DISEÑO DE INFRAESTRUCTURA DE RED Y SOPORTE INFORMÁTICO PARA UN CENTRO PÚBLICO DE EDUCACIÓN INFANTIL Y PRIMARIA*. Retrieved from http://oa.upm.es/4976/3/PFC_JUAN_MARUGAN_MERINEROx.pdf

Microsoft. (2015). Ciclo de Vida de Desarrollo Seguro de Software (es-ES) - TechNet Articles - United States (English) - TechNet Wiki. Retrieved January 8, 2019, from https://social.technet.microsoft.com/wiki/contents/articles/36676.ciclo-de-vida-de-desarrollo-seguro-de-software-es-es.aspx

Microsoft. (2018a). Introducción: Microsoft Azure Threat Modeling Tool | Microsoft Docs. Retrieved July 16, 2019, from https://docs.microsoft.com/es-es/azure/security/azure-security-threat-modeling-tool-getting-started

Microsoft. (2018b). Microsoft Security Development Lifecycle Practices. Retrieved January 8, 2019, from https://www.microsoft.com/en-us/securityengineering/sdl/practices

Miríada. (2016). Miríadax -. Retrieved January 9, 2019, from https://miriadax.net/web/ciberseguridad-entender-los-ataques-para-desplegar-contramedidas/reto?p_p_auth=nrDHDe5K&p_p_id=resourceExternalActivity_WAR_liferaylmsportlet&p_p_lifecycle=0&p_p_state=normal&p_p_mode=view&p_p_col_id=column-1&p_p_col_count=1&p_r_p

*Modelo de seguridad y Auditoria para aplicaciones web basado en metología OWASP*. (2016). Retrieved from https://tesis.ipn.mx/bitstream/handle/123456789/17973/Tesina C7.1675.pdf?Sequence=1

Nacional, C. C., & Nacional, C. (2018). Ciberamenzas y tendencias.

Nancy G. Leveson, C. S. T. (1993). An Investigation of the Therac-25 Acidents. Retrieved from https://www.cs.umd.edu/class/spring2003/cmsc838p/Misc/therac.pdf

Nohemy, L., Velandia, M., Mesías, W., & López, L. (2015). ESCOGER UNA METODOLOGÍA PARA DESARROLLAR SOFTWARE, DIFÍCIL DECISIÓN CHOOSING A SOFTWARE DEVELOPMENT METHODOLOGY, HARD DECISION, *10*, 98–109. Retrieved from

http://www.educacioneningenieria.org

Observatorio TIC. (2017). Estadistica. Retrieved March 13, 2018, from https://observatoriotic.mintel.gob.ec/estadistica/

Olivares, B., & Eduardo, G. (n.d.). *MODELADO DE AMENAZAS, UNA TÉCNICA DE ANÁLISIS Y GESTIÓN DE RIESGO ASOCIADO A SOFTWARE Y APLICACIONES*. Retrieved from http://polux.unipiloto.edu.co:8080/00004067.pdf

OWASP. (n.d.). CheatSheetSeries/Access_Control_Cheat_Sheet.md at master · OWASP/CheatSheetSeries · GitHub. Retrieved July 16, 2019, from https://github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/Access_Control_Cheat_Sheet.md

OWASP. (2018). Retrieved May 9, 2018, from https://www.owasp.org/index.php/Main_Page

OWASP. (2019a). Metodología de calificación de riesgo de OWASP - OWASP. Retrieved July 16, 2019, from https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology#The_OWASP_Risk_Rating_Methodology

OWASP. (2019b). Modelado de Amenazas - OWASP. Retrieved July 16, 2019, from https://www.owasp.org/index.php/Modelado_de_Amenazas#Pasos_del_modelado_de_amenazas

OWASP Foundation. (2017). OWASP Top 10 -2017, 25. Retrieved from https://www.owasp.org/images/5/5e/OWASP-Top-10-2017-es.pdf

Payne, J., Watson, C., Williams, J., Mcgovern, J., & Steingruebl, A. (2016). Software Assurance Maturity Model.

Pinilla, C., Julian, A., Buenas, P., Para El Desarrollo De Código, S., Castañeda, P., & Alejandro, J. (n.d.). *Universidad Piloto de Colombia Buenas Prácticas para el Desarrollo de Código Seguro*. Retrieved from http://polux.unipiloto.edu.co:8080/00001881.pdf

Rideout, V. J., Foehr, U. G., & Roberts, D. F. (2011). Generation M2: Media in the Lives of 8 to 18 year Olds: A Kaiser Family Foundation Study, January 2010. Retrieved from https://kaiserfamilyfoundation.files.wordpress.com/2013/04/8010.pdf

Riffo, M. A. (2008). Vulnerabilidades de las Redes TCP/IP y Principales Mecanismos de Seguridad. *In Vitro*, *3*(2), 1–23. Retrieved from http://www.ncbi.nlm.nih.gov.myaccess.library.utoronto.ca/pubmed/11720961

Servitja Roca, X. (2013). Opinión Documento de Opinión. Retrieved from http://www.ieee.es/Galerias/fichero/docs_opinion/2013/DIEEEO42-2013_Inteligencia_Iran_XSertvija.pdf

Simplified Implementation of the Microsoft SDL. (2016).

*SSL*. (n.d.). Retrieved from http://www.juntadeandalucia.es/empleo/recursos/material_didactico/especialidades/materialdidactico_administrador_servidores/Content/4-seguridad/8-SSL.pdf

Vieites, A. G. (2007). Enciclopedia de la seguridad informática. Retrieved from http://www.sidalc.net/cgi-bin/wxis.exe/?IsisScript=CAMOA.xis&method=post&formato=2&cantidad=1&expresion=mfn=001653