



**Departamento de Ciencias de la Computación**  
Universidad de las Fuerzas Armadas - ESPE

# **Cuaderno de clases de la asignatura de Ingeniería de Seguridad de Software**

**Parcial No. 1**

**Nombres:**

Yeshua Amador Chiliquinga Amaya

**Carrera / Asignatura:** Ingeniería de Software / Ingeniería de  
Seguridad de Software

**NRC:** 2540

**Nombre del profesor:** Walter Fuentes, PhD

**Fecha de presentación:** 24 de mayo del 2025

# Índice

<b>1. Clase 1 - 17 de Abril 2025: Introducción a la Ingeniería de Seguridad de Software</b>	<b>4</b>
1.1. Objetivos de Aprendizaje . . . . .	4
1.2. ¿Por qué de la asignatura? . . . . .	4
1.3. Syllabus . . . . .	4
1.3.1. Unidad 1: Introducción a la seguridad de la información . . . . .	4
1.4. Unidad 2: Mecanismos de seguridad . . . . .	5
1.5. Unidad 3: Seguridad de Redes y Aplicaciones . . . . .	5
1.6. Plataforma (topología) experimental . . . . .	6
1.7. Referencias Bibliográficas . . . . .	6
<b>2. Clase 2 - 24 de Abril 2025</b>	<b>6</b>
2.1. Tríada CID . . . . .	6
2.2. Delito Informático (COIP) . . . . .	6
2.3. Ciber-atacantes . . . . .	6
2.4. Ciber-ataques . . . . .	7
2.5. Ciber-defensa . . . . .	7
2.6. Tipos de amenazas . . . . .	8
2.7. Gestión del Tiempo . . . . .	8
<b>3. Clase 3 - Herramientas Tecnológicas para la Seguridad del Software</b>	<b>8</b>
3.1. Objetivos de Aprendizaje . . . . .	8
3.2. Arquitectura de Seguridad en Redes . . . . .	8
3.2.1. Niveles de Seguridad . . . . .	9
3.3. Herramientas de Seguridad por Categoría . . . . .	9
3.3.1. Seguridad Inalámbrica . . . . .	9
3.3.2. Seguridad Perimetral . . . . .	10
3.3.3. Monitoreo y Análisis . . . . .	10
3.4. Clasificación de Herramientas . . . . .	10
3.5. Herramientas Avanzadas . . . . .	10
3.6. Conclusión . . . . .	10
3.7. Próximos Pasos . . . . .	11
<b>4. Clase 4 - 6 de Mayo 2025: Mecanismos de Seguridad y Criptografía</b>	<b>11</b>
4.1. Objetivo Principal . . . . .	11
4.2. Reflexión: El Valor de una Persona . . . . .	11
4.3. Mecanismos de Seguridad . . . . .	11
4.3.1. Firewalls . . . . .	11
4.3.2. UFW (Uncomplicated Firewall) . . . . .	12
4.3.3. IPTTables . . . . .	12
4.3.4. Otras Herramientas . . . . .	13
4.4. Ataques de Fuerza Bruta . . . . .	13
4.4.1. Concepto . . . . .	13
4.4.2. Archivos Críticos en Linux . . . . .	14
4.4.3. Herramientas Comunes . . . . .	14
4.5. Keyloggers . . . . .	14
4.6. Estándares de Seguridad . . . . .	14

4.6.1. ISO/IEC 27000 . . . . .	14
4.6.2. NIST Cybersecurity Framework . . . . .	14
<b>5. Introducción a la Criptografía</b>	<b>14</b>
5.1. Conceptos Básicos . . . . .	14
5.2. Tipos de Criptografía . . . . .	14
5.2.1. Simétrica . . . . .	14
5.2.2. Asimétrica . . . . .	15
5.2.3. Funciones Hash . . . . .	15
5.3. Recomendaciones de Películas . . . . .	15
5.4. Práctica de Laboratorio: Ataques de Fuerza Bruta . . . . .	15
5.5. Conclusión . . . . .	15
<b>6. Clase 5 - 8 de Mayo 2025: Ataques de Fuerza Bruta y Keyloggers</b>	<b>15</b>
6.1. Objetivos de Aprendizaje . . . . .	15
<b>7. Herramientas de Ataque de Fuerza Bruta en Kali Linux</b>	<b>16</b>
7.1. Hydra: Ataques a Protocolos de Red . . . . .	16
7.2. John the Ripper: Descifrado de Hashes . . . . .	16
7.3. Medusa: Ataques Multi-protocolo . . . . .	17
7.4. Burp Suite: Ataques a Formularios Web . . . . .	17
7.5. Hashcat: Descifrado Avanzado de Hashes . . . . .	18
<b>8. Keyloggers</b>	<b>19</b>
8.1. ¿Qué es un Keylogger? . . . . .	19
8.2. Implicaciones de Seguridad . . . . .	19
<b>9. Conclusión</b>	<b>19</b>
<b>10. Próximos Pasos</b>	<b>20</b>
<b>11. Clase 6 - 13 de Mayo 2025: SGSI y Normativa ISO 27000</b>	<b>20</b>
11.1. Requerimientos Funcionales de la Seguridad de la Información . . . . .	20
11.2. Componentes del SGSI . . . . .	20
11.3. Conceptos Clave de ISO 27000 . . . . .	21
11.4. Beneficios de Implementar un SGSI . . . . .	22
11.5. Herramientas Tecnológicas para la Seguridad . . . . .	23
11.6. Recursos Adicionales . . . . .	23
11.7. Conclusión . . . . .	23
11.8. Próximos Pasos . . . . .	24
<b>12. Clase 7 - 15 de Mayo 2025: Especificación de Requisitos de Seguridad</b>	<b>24</b>
12.1. Objetivo de Aprendizaje . . . . .	24
12.2. Reflexión . . . . .	24
12.3. Consejos de Edición . . . . .	25
12.3.1. Escritura de párrafos . . . . .	25

<b>13. Clase 8 - 20 de Mayo 2025: Especificación de Requisitos de Seguridad</b>	<b>25</b>
13.1. Introducción . . . . .	25
13.1.1. Estándares Clave . . . . .	25
13.2. Requisitos Esenciales de Seguridad . . . . .	26
13.2.1. 1. Autenticación y Autorización . . . . .	26
13.2.2. 2. Validación de Entrada . . . . .	26
13.2.3. 3. Seguridad en la Comunicación . . . . .	26
13.3. OWASP Top 10: Amenazas Principales . . . . .	26
13.4. Práctica de Laboratorio: OWASP ZAP . . . . .	26
13.5. Checklist de Seguridad . . . . .	27
13.6. Recursos Adicionales . . . . .	27
13.7. Conclusión . . . . .	27
<b>14. Clase 9 - 22 de Mayo 2025: Análisis de Amenazas y Vulnerabilidades</b>	<b>27</b>
14.1. Objetivos de aprendizaje . . . . .	27
14.2. Reflexión: Zona de Confort . . . . .	28
14.3. Mia: Aprendizaje Activo . . . . .	28
14.4. Marco Teórico: Amenazas y Vulnerabilidades . . . . .	28
14.4.1. Ejemplos de vulnerabilidades comunes . . . . .	29
14.5. ESQUEMA GUBERNAMENTAL DE SEGURIDAD DE LA INFORMACIÓN (EGSI) . . . . .	30
<b>15. Práctica de laboratorio 6: Análisis de vulnerabilidades con OWASP-ZAP</b>	<b>30</b>
15.1. Objetivo de aprendizaje . . . . .	30
15.2. Marco teórico . . . . .	30
15.2.1. OWASP . . . . .	30
15.2.2. OWASP-ZAP (Zed Attack Proxy) . . . . .	30

# 1. Clase 1 - 17 de Abril 2025: Introducción a la Ingeniería de Seguridad de Software

## 1.1. Objetivos de Aprendizaje

1. Motivación
2. Presentación
3. Syllabus
4. Unidad 1
  - a) Introducción a la Ciberseguridad

## 1.2. ¿Por qué de la asignatura?

### Ciudadanía digital

Cuando Julian Assange estaba en la embajada recibimos más de 4 millones de ataques.

## 1.3. Syllabus

### 1.3.1. Unidad 1: Introducción a la seguridad de la información

- Seguridad de la información
- Ciberseguridad
- Componentes
  - Amenazas
  - Vulnerabilidad
  - Riesgo
  - Incidentes
- Tríada CID (ISO 27000)
  - Confidencialidad
  - Integridad
  - Disponibilidad
- Herramientas tecnológicas para garantizar la CID
- Análisis de amenazas y vulnerabilidades
- Requerimientos funcionales para implementación de tecnologías de ciberseguridad

## 1.4. Unidad 2: Mecanismos de seguridad

- Criptografía
  - Simétrica
  - Asimétrica
- Funciones Hash aplicadas al software
- Firma digital y certificados digitales
- Autenticación vs Identificación
- Seguridad Física
- Protocolos Criptográficos

## 1.5. Unidad 3: Seguridad de Redes y Aplicaciones

- Seguridad de Redes
  - Perimetral
  - En profundidad
  - Multinivel
- IPS/IDS
- Firewall
- Seguridad de aplicaciones en internet
- OWASP Top 10
- SGSI (Sistema de Gestión de Seguridad de la Información)
  - Lógica
  - Física
  - Legal
  - Procedimental
- Temas adicionales
  - Blockchain
  - Seguridad criptográfica
  - Computación en la nube
- Seguridad en la nube
- Deep web
- IoT (Internet de las Cosas)

## 1.6. Plataforma (topología) experimental

- Entorno controlado
- COIP (Código Orgánico Integral Penal)
- Delitos informáticos tipificados

**VNE:** Virtual Network Environment

## 1.7. Referencias Bibliográficas

- Norma ISO 27000
- Código Orgánico Integral Penal (COIP)
- Material del curso - Ing. Walter Fuertes, PhD

# 2. Clase 2 - 24 de Abril 2025

## 2.1. Tríada CID

La tríada CID (Confidencialidad, Integridad, Disponibilidad) es un modelo fundamental en seguridad de la información que establece los tres pilares principales de la protección de datos.

## 2.2. Delito Informático (COIP)

- Todo acto malicioso en contra de las personas, empresas y estado utilizando herramientas informáticas en el ciberespacio.
- Incluye actividades como acceso no autorizado, interrupción de servicios, fraude electrónico, entre otros.

## 2.3. Ciber-atacantes

- **Insiders:** Atacantes que vulneran desde dentro de la organización.
- **Spammers:** Distribuyen correos no deseados o maliciosos.
- **Piratas telefónicos:** Se introducen en las líneas telefónicas para realizar llamadas sin costo.
- **Geeks:** Personas con amplios conocimientos técnicos.
- **Hackers de sombrero blanco/negro:** Éticos vs. malintencionados.

ATACANTE	DESCRIPCIÓN
<b>Cracker</b>	Personas que rompen o vulneran algún sistema de seguridad de forma ilícita.
<b>Hacker</b>	Su fin es detectar defectos de seguridad para acceder o irrumpir ilegalmente en los equipos y sistemas informáticos.
<b>Sniffer</b>	Captura y analiza los paquetes que se envían y reciben con fines maliciosos.
<b>Phisher</b>	Persona que engaña para obtener contraseñas, números de tarjeta de crédito, etc.
<b>Phreaker</b>	Pirata telefónico, utiliza el teléfono para cometer delitos informáticos
<b>Hactivists</b>	Abuso de la web para promover causas sociales o fines políticos.

Figura 1: Descripción de tipos de ciber-atacantes

## 2.4. Ciber-ataques

- **Ingeniería Social:** Aprovechan la ingenuidad de los usuarios (capa 8 del modelo OSI).
- **DoS/DDoS:** Ataques de denegación de servicio.
- **Aplicaciones Web:** Vulnerabilidades según OWASP.
- **Ciber-terrorismo**
- **Ciber-espionaje**
- **Ciber-sabotaje**

## 2.5. Ciber-defensa

Para contrarrestar las amenazas se necesitan Ciber-soldados o COCICIBER (Comando de Ciberdefensa). La ciberdefensa utiliza la ciberseguridad en tres niveles:

- **Pasiva:** Medidas preventivas.
- **Activa:** Detección y respuesta.
- **Ofensiva:** Contramedidas activas.

## 2.6. Tipos de amenazas

- Ataques de fuerza bruta
- Man in the Middle (MitM)
- Malware:
  - Troyanos
  - Botnets (redes de equipos zombis)
  - Virus
  - Gusanos
  - Insectos
- Ciberarmas

## 2.7. Gestión del Tiempo

- Importancia de la gestión eficiente del tiempo
- Distribución recomendada:
  - Perfección intelectual
  - Salud
  - Actividad Física
  - Entretenimiento
  - Tiempo para la familia
  - Trabajo
  - Perfeccionamiento Espiritual

# 3. Clase 3 - Herramientas Tecnológicas para la Seguridad del Software

## 3.1. Objetivos de Aprendizaje

- Comprender la arquitectura jerárquica de seguridad en redes
- Identificar y clasificar herramientas tecnológicas según su función
- Analizar mecanismos de seguridad en diferentes niveles de red
- Evaluar herramientas de prevención, detección, cifrado y mitigación

## 3.2. Arquitectura de Seguridad en Redes

La seguridad en redes se implementa a través de una arquitectura jerárquica centralizada con diferentes niveles de acceso y control:

### 3.2.1. Niveles de Seguridad

#### 1. Nivel de Cliente/Acceso

- Perfil de usuario
- Claves de acceso
- Software de seguridad local

#### 2. Nivel de Servidor

- Gestión de cuentas
- Firewalls (iptables, ufw)
- Sistemas de detección/prevención de intrusiones (IDS/IPS)

#### 3. Nivel de Red (Core)

- Enrutamiento seguro
- Filtrado de paquetes
- NAT (Traducción de Direcciones de Red)



Figura 2: Arquitectura de seguridad en redes con Netgate

## 3.3. Herramientas de Seguridad por Categoría

### 3.3.1. Seguridad Inalámbrica

#### ■ Protocolos de Seguridad:

- WEP (Wired Equivalent Privacy)
- WPA/WPA2 (Wi-Fi Protected Access)
- EAP (Extensible Authentication Protocol)

#### ■ Dispositivos:

- Puntos de Acceso Inalámbrico (WAP)
- Controladores de Redes Inalámbricas

### 3.3.2. Seguridad Perimetral

- **Firewalls:**
  - pfSense
  - ClaroOS
  - Firewalls de Próxima Generación (NGFW)
- **Sistemas de Gestión de Amenazas Unificadas (UTM)**
- **Servidores Proxy**

### 3.3.3. Monitoreo y Análisis

- **Wireshark:** Análisis de tráfico de red
- **Nmap:** Mapeo de red y escaneo de puertos
- **SIEM:** Gestión de Eventos e Información de Seguridad
- **Metasploit:** Marco de pruebas de penetración

## 3.4. Clasificación de Herramientas

Tipo	Herramientas	Propósito
Prevención	Firewalls, UTM, NGFW	Bloquear amenazas antes de que ingresen
Detección	IDS/IPS, SIEM	Identificar actividades sospechosas
Cifrado	VPN, WPA3, SSL/TLS	Proteger la confidencialidad de los datos
Mitigación	Balanceadores de carga, DDoS Protection	Reducir el impacto de los ataques

Cuadro 1: Clasificación de herramientas de seguridad

## 3.5. Herramientas Avanzadas

- **APT (Amenazas Persistentes Avanzadas):** Tácticas sofisticadas para atacar objetivos específicos
- **Metasploit Framework:** Para pruebas de penetración y desarrollo de exploits
- **Nmap:** Herramienta de descubrimiento de red y auditoría de seguridad

## 3.6. Conclusión

La seguridad del software requiere un enfoque en capas que combine múltiples herramientas tecnológicas. Desde la protección perimetral hasta el monitoreo continuo, cada capa juega un papel crucial en la defensa contra amenazas cibernéticas. La selección e implementación adecuada de estas herramientas, junto con prácticas de seguridad sólidas, son fundamentales para mantener la integridad, confidencialidad y disponibilidad de los sistemas de información.

### 3.7. Próximos Pasos

- Investigar sobre el Congreso Internacional de Ciencia y Tecnología ESPE 2025
- Desarrollar un artículo científico siguiendo el formato Springer
- Profundizar en el estudio de herramientas de seguridad específicas

## 4. Clase 4 - 6 de Mayo 2025: Mecanismos de Seguridad y Criptografía

### 4.1. Objetivo Principal

Diseño e implementación de herramientas tecnológicas para la seguridad de software, enfocadas en la protección contra diversos tipos de ataques y malware.

### 4.2. Reflexión: El Valor de una Persona

$$V = (H + C) \times A$$

- **V:** Valor de una persona
- **H:** Habilidades (Docker, aptitudes técnicas)
- **C:** Conocimiento
- **A:** Actitud

**Importante:** La actitud es fundamental. Sin una actitud positiva, disposición para socializar, hacer preguntas y participar activamente, incluso las habilidades técnicas más destacadas pueden verse limitadas en el entorno laboral.

### 4.3. Mecanismos de Seguridad

#### 4.3.1. Firewalls

- Primera línea de defensa en redes
- Controla el tráfico entrante y saliente
- Tipos:
  - Firewall de red
  - Firewall de aplicaciones web (WAF)
  - Firewall de próxima generación (NGFW)

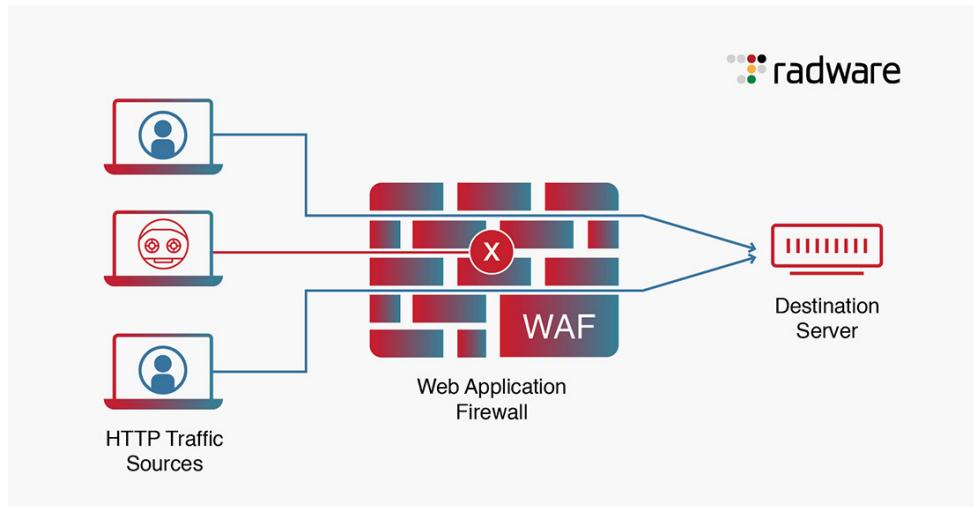
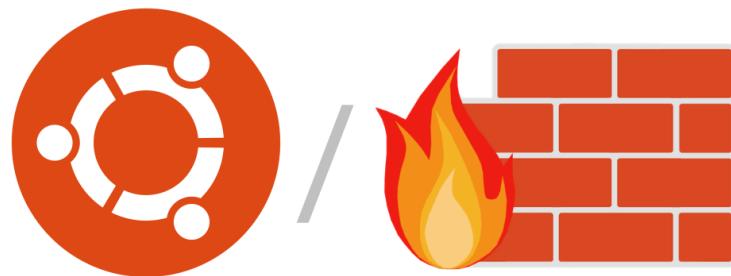


Figura 3: Web Application Firewall (WAF) protegiendo aplicaciones web

#### 4.3.2. UFW (Uncomplicated Firewall)

- Interfaz simplificada para iptables
- Fácil de configurar y usar
- Ideal para servidores y estaciones de trabajo



## Ubuntu UFW

Figura 4: Configuración básica de UFW

#### 4.3.3. IPTables

- Herramienta de firewall basada en línea de comandos
- Permite configurar reglas de filtrado de paquetes
- Base para muchos firewalls modernos



Figura 5: Ejemplo de configuración de IPTables

#### 4.3.4. Otras Herramientas

- **ClaroOS**: Solución de seguridad unificada
- **pfSense**: Firewall de código abierto
- **Snort**: Sistema de detección/prevención de intrusiones
- **IDS/IPS**:
  - HIDS/HIPS: Basado en host
  - NIDS/NIPS: Basado en red

### 4.4. Ataques de Fuerza Bruta

#### 4.4.1. Concepto

Ataque contra sistemas de autenticación que prueba múltiples combinaciones de credenciales hasta encontrar la correcta.

#### 4.4.2. Archivos Críticos en Linux

- `/etc/passwd`: Información de usuarios
- `/etc/shadow`: Contraseñas encriptadas

#### 4.4.3. Herramientas Comunes

- **John the Ripper**
- **Hydra**
- **Medusa**

### 4.5. Keyloggers

- Software o hardware que registra las pulsaciones del teclado
- Usado para robo de credenciales
- Ejemplo de fraude informático

### 4.6. Estándares de Seguridad

#### 4.6.1. ISO/IEC 27000

- 27001: Requisitos para SGSI
- 27005: Gestión de Riesgos
- 27032: Ciberseguridad

#### 4.6.2. NIST Cybersecurity Framework

Marco de trabajo para gestionar riesgos de ciberseguridad.

## 5. Introducción a la Criptografía

### 5.1. Conceptos Básicos

- **Criptología**: Estudio de las comunicaciones seguras
- **Criptoanálisis**: Estudio de métodos para descifrar información sin autorización
- **Cifrado/Descifrado**: Procesos de codificación y decodificación de información

### 5.2. Tipos de Criptografía

#### 5.2.1. Simétrica

- Usa una sola clave secreta
- Ejemplo: AES, DES, 3DES

### 5.2.2. Asimétrica

- Usa par de claves (pública/privada)
- Ejemplo: RSA, ECC

### 5.2.3. Funciones Hash

- Transformación irreversible de datos
- Ejemplo: SHA-256, MD5
- Usado en integridad de datos y almacenamiento de contraseñas

## 5.3. Recomendaciones de Películas

- *The Imitation Game* (Sobre Alan Turing)
- *A Beautiful Mind*

## 5.4. Práctica de Laboratorio: Ataques de Fuerza Bruta

1. Configurar entorno de prueba
2. Generar diccionario de contraseñas
3. Ejecutar ataque usando herramientas como Hydra
4. Analizar resultados y contramedidas

## 5.5. Conclusión

La seguridad informática requiere un enfoque integral que combine herramientas técnicas, estándares de seguridad y concientización del usuario. La criptografía juega un papel fundamental en la protección de la información, mientras que los firewalls y sistemas de detección de intrusiones protegen los sistemas de amenazas externas e internas.

## 6. Clase 5 - 8 de Mayo 2025: Ataques de Fuerza Bruta y Keyloggers

### 6.1. Objetivos de Aprendizaje

1. Comprender los ataques de fuerza bruta y sus herramientas
2. Analizar el funcionamiento de los keyloggers
3. Realizar prácticas de laboratorio con herramientas de seguridad

## 7. Herramientas de Ataque de Fuerza Bruta en Kali Linux

Los ataques de fuerza bruta son intentos sistemáticos de adivinar credenciales de autenticación probando múltiples combinaciones.

## 7.1. Hydra: Ataques a Protocolos de Red

- Realiza ataques de fuerza bruta contra múltiples protocolos (SSH, HTTP, FTP, etc.)
  - Ideal para probar credenciales en servicios de red
  - Fácil de usar y altamente configurable



Figura 6: Ejemplo de uso de Hydra para ataques de fuerza bruta

## 7.2. John the Ripper: Descifrado de Hashes

- Especializado en descifrar contraseñas a partir de hashes
  - Soporta múltiples algoritmos de hash
  - Incluye modo de diccionario y ataque por fuerza bruta



Figura 7: Interfaz de John the Ripper para descifrado de contraseñas

### 7.3. Medusa: Ataques Multi-protocolo

- Similar a Hydra, con soporte para múltiples protocolos
- Eficiente en el uso de recursos
- Permite ataques paralelos



Figura 8: Medusa en acción realizando ataques multi-protocolo

### 7.4. Burp Suite: Ataques a Formularios Web

- Herramienta integral para pruebas de seguridad web

- Permite interceptar y modificar peticiones HTTP/HTTPS
- Incluye funcionalidad para ataques de fuerza bruta



Figura 9: Burp Suite para pruebas de seguridad web

### 7.5. Hashcat: Descifrado Avanzado de Hashes

- Herramienta avanzada para descifrado de hashes
- Soporta múltiples algoritmos de hash
- Utiliza la GPU para acelerar el proceso de descifrado



Figura 10: Hashcat para descifrado avanzado de hashes

## 8. Keyloggers

### 8.1. ¿Qué es un Keylogger?

- Software o hardware que registra las pulsaciones del teclado
- Puede capturar contraseñas, mensajes y otra información confidencial
- Usado tanto para monitoreo legítimo como para actividades maliciosas



Figura 11: Ejemplo de keylogger capturando pulsaciones

### 8.2. Implicaciones de Seguridad

- **Amenaza a la privacidad:** Captura de información confidencial
- **Robo de identidad:** Obtención de credenciales de acceso
- **Impacto financiero:** Pérdidas económicas por fraude
- **Aspectos legales:** Uso no autorizado es penado por la ley

## 9. Conclusión

- Las herramientas de fuerza bruta son efectivas contra configuraciones débiles
- Hydra y Medusa son versátiles para pruebas de red
- Los keyloggers representan una amenaza significativa a la privacidad
- Es esencial implementar medidas de seguridad adecuadas:
  - Contraseñas fuertes
  - Autenticación de dos factores
  - Monitoreo de actividad sospechosa
  - Actualizaciones de seguridad regulares

## 10. Próximos Pasos

- Practicar con herramientas de seguridad en entornos controlados
- Aprender sobre contramedidas y detección de ataques
- Explorar técnicas de hardening de sistemas

## 11. Clase 6 - 13 de Mayo 2025: SGSI y Normativa ISO 27000

### 11.1. Requerimientos Funcionales de la Seguridad de la Información

- Marco normativo ISO/IEC 27000
- Sistema de Gestión de Seguridad de la Información (SGSI)
- Estándares clave:
  - ISO/IEC 27001: Requisitos para SGSI
  - ISO/IEC 27002: Código de buenas prácticas
    - Control de acceso
    - Criptografía
    - Controles de seguridad

### 11.2. Componentes del SGSI

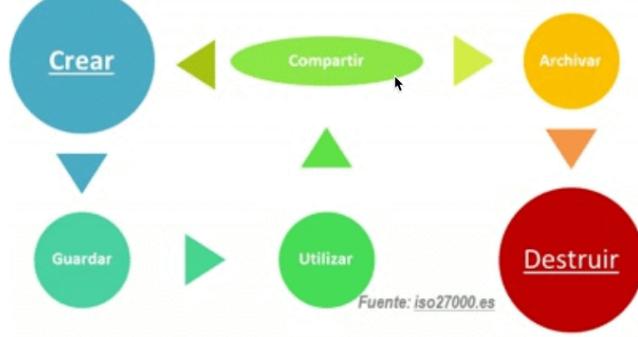
- **Física:** Protección de instalaciones y equipos
- **Lógica:** Controles de software y acceso lógico
- **Procedimental:** Políticas y procedimientos
- **Legal:** Marco normativo aplicable
  - COIP (Código Orgánico Integral Penal)
  - Ley de Protección de Datos
  - Ley de Comercio Electrónico

## ¿Qué es un SGSI?

**Confidencialidad:** la información no se pone a disposición ni se revela a individuos, entidades o procesos no autorizados.

**Integridad:** mantenimiento de la exactitud y completitud de la información y sus métodos de proceso.

**Disponibilidad:** acceso y utilización de la información y los sistemas de tratamiento de la misma por parte de los individuos, entidades o procesos autorizados cuando lo requieran.

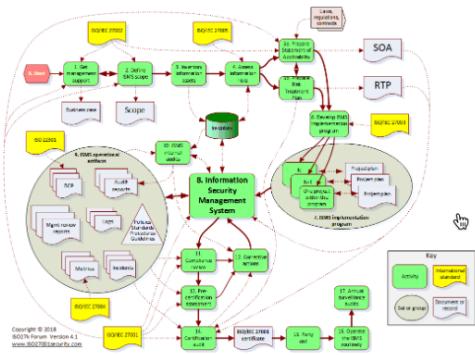


Fuente: [iso27000.es](http://iso27000.es)

Figura 12: Componentes de un Sistema de Gestión de Seguridad de la Información

### 11.3. Conceptos Clave de ISO 27000

- **Organización:** Entidad que gestiona la información
- **Sistema:** Conjunto de elementos interrelacionados para un fin común
- **Activo:** Recurso con valor que requiere protección
- **Alcance:** Límites y cobertura del SGSI



### 1 Alcance SGSI (4.3)

El alcance del SGSI aclara los límites del SGSI en función del contexto y/o importancia y ubicación de los activos críticos de información de la organización (por ejemplo, unidades, ubicaciones o departamentos) y los riesgos propios o externos asociados (p.ej. leyes y reglamentos, obligaciones contractuales, estrategias y políticas impuestas por organismos centrales).

Se debe tener en cuenta los flujos de información que cruzan los límites del alcance.

Una estrategia de alto nivel impulsada por la organización o una declaración de visión (ya sea hecha o al menos formalmente respaldada por la alta gerencia) es una forma de cristalizar tanto el alcance como el propósito de aplicación del SGSI, y puede ser útil para fines de concientización así como de promoción.

### 2 Política del SGSI (5.2)

Establece y confirma el compromiso de la alta dirección con los objetivos de seguridad de la información de la organización y la mejora continua del SGSI, entre otros posibles aspectos relevantes.

La alta gerencia puede preferir una **política de tipo de gobierno único, sucinta, amplia / general** (que satisface formalmente el requisito de ISO), completada con otro conjunto adicional de políticas complementarias de riesgo, seguridad,

Figura 13: Alcance de un SGSI en una organización

## 11.4. Beneficios de Implementar un SGSI

- Protección de la información crítica
- Cumplimiento normativo
- Reducción de riesgos de seguridad
- Mejora continua de los procesos
- Ventaja competitiva

# Beneficios

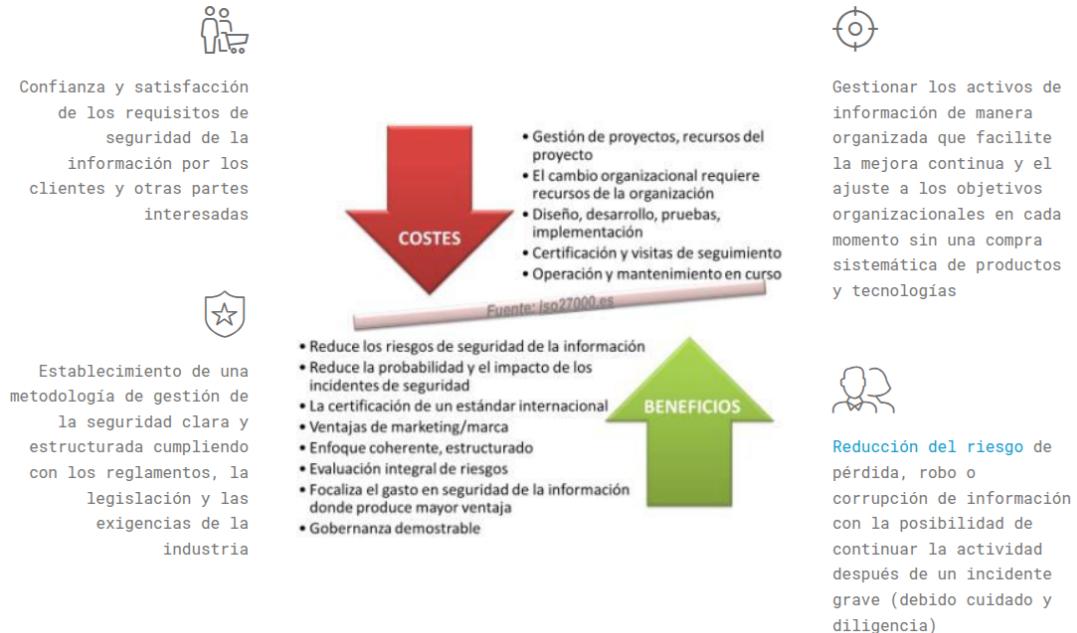


Figura 14: Beneficios de implementar un SGSI basado en ISO 27001

## 11.5. Herramientas Tecnológicas para la Seguridad

- **Mapas de Ciberataques:** Visualización de amenazas en tiempo real
- **Herramientas de Pruebas de Seguridad:**
  - SQLMap para pruebas de inyección SQL
  - Escáneres de vulnerabilidades
  - Herramientas de análisis forense

## 11.6. Recursos Adicionales

- Sitio web oficial de ISO 27000: <https://www.iso27000.es/>
- Documentación oficial de ISO/IEC 27001 e ISO/IEC 27002
- Guías de implementación de SGSI

## 11.7. Conclusión

- La implementación de un SGSI basado en ISO 27001 es fundamental para la gestión efectiva de la seguridad de la información
- La norma proporciona un enfoque sistemático para gestionar la información confidencial

- La adopción de estas normas ayuda a las organizaciones a proteger sus activos de información

## 11.8. Próximos Pasos

- Estudiar los requisitos específicos de ISO 27001
- Analizar casos de estudio de implementación de SGSI
- Explorar herramientas para la gestión de seguridad de la información

# 12. Clase 7 - 15 de Mayo 2025: Especificación de Requisitos de Seguridad

## 12.1. Objetivo de Aprendizaje

- Especificación de Requerimientos de seguridad del Software
  - NIST
  - CYBERSECURITY
  - Framework
- Ensayo argumentativo
  - Cyber Threats Word Maps
  - Mejorar la escritura
- Práctica de laboratorio SQLMap

## 12.2. Reflexión

$F = M + V + CA$  (Alberto Meraní)

Felicidad = F

Metas = M

Vínculos = V

Cualidades Afectivas = CA

→ Es una decisión

Metas → Soñarlas, dibujarlas acciones

Para ello podemos hacer un metagrama con los siguientes campos:



Figura 15: Metagrama para la planificación de metas

### 12.3. Consejos de Edición

#### 12.3.1. Escritura de párrafos

- Coherencia y secuencialidad en los párrafos (utilizar conectores gramaticales)
- Evitar el uso de adjetivos calificativos
- Evitar errores ortográficos
- Evitar el uso de adverbios

## 13. Clase 8 - 20 de Mayo 2025: Especificación de Requisitos de Seguridad

### 13.1. Introducción

En el desarrollo de software seguro, los requisitos de seguridad son la base para construir aplicaciones confiables. Vamos a explorar los estándares clave y cómo implementar estos requisitos de manera efectiva.

#### 13.1.1. Estándares Clave

Estándar	Descripción
ISO 27000	Familia de normas para la gestión de seguridad de la información
NIST	Marco de ciberseguridad del Instituto Nacional de Estándares y Tecnología
OWASP	Proyecto de seguridad de aplicaciones web abierto

## 13.2. Requisitos Esenciales de Seguridad

### 13.2.1. 1. Autenticación y Autorización

**Ejemplo Práctico:** Un sistema bancario requiere autenticación de dos factores para transacciones superiores a \$1,000.

- **MFA:** Implementación de autenticación multifactor
- **RBAC:** Control de acceso basado en roles
- **JWT:** Uso de tokens seguros para sesiones

### 13.2.2. 2. Validación de Entrada

```
def validar_entrada(usuario):  
    # Validación por lista blanca  
    caracteres_permitidos = set("abcdefghijklmnopqrstuvwxyz0123456789_-")  
    if not all(c in caracteres_permitidos for c in usuario):  
        raise ValueError("Caracteres no permitidos")  
    return usuario
```

### 13.2.3. 3. Seguridad en la Comunicación

Protocolo	Uso Seguro
HTTP	No recomendado
HTTPS	Obligatorio para todo tipo de comunicación segura
SFTP	Para transferencia segura de archivos
WPA3	Para redes inalámbricas seguras

## 13.3. OWASP Top 10: Amenazas Principales

1. Inyección (SQL, NoSQL, OS, LDAP)
2. Autenticación Rota
3. Exposición de Datos Sensibles
4. Entidades Externas XML (XXE)
5. Control de Acceso Roto
6. Configuración de Seguridad Incorrecta
7. Cross-Site Scripting (XSS)
8. Deserialización Insegura
9. Componentes Vulnerables
10. Registro y Monitoreo Insuficientes

## 13.4. Práctica de Laboratorio: OWASP ZAP

1. Instalar OWASP ZAP
2. Configurar el navegador para usar ZAP como proxy
3. Analizar una aplicación web
4. Identificar vulnerabilidades
5. Generar reporte de hallazgos

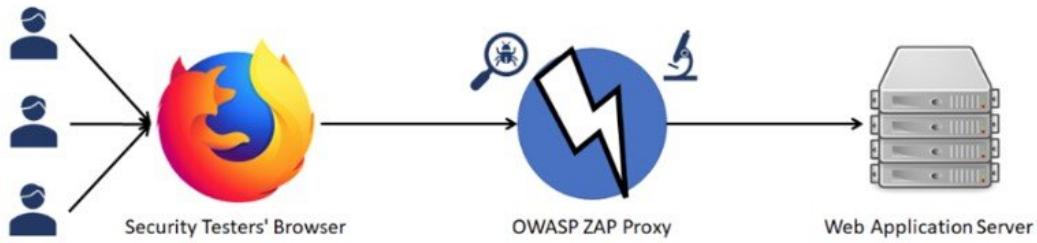


Figura 16: Interfaz de OWASP ZAP para pruebas de seguridad

### 13.5. Checklist de Seguridad

- ¿Se validan todas las entradas del usuario?
- ¿Se implementa el principio de mínimo privilegio?
- ¿Los datos sensibles están cifrados?
- ¿Existen registros de auditoría suficientes?
- ¿Se actualizan regularmente las dependencias?

### 13.6. Recursos Adicionales

- OWASP: <https://owasp.org/>
- NIST: <https://csrc.nist.gov/>
- ISO 27001: <https://www.iso.org/isoiec-27001>

### 13.7. Conclusión

La seguridad del software no es un producto, sino un proceso continuo. La implementación de estos requisitos debe ser parte integral del ciclo de vida del desarrollo de software, no una ocurrencia tardía. *“La seguridad siempre es excesiva hasta que ya no es suficiente.”*

## 14. Clase 9 - 22 de Mayo 2025: Análisis de Amenazas y Vulnerabilidades

### 14.1. Objetivos de aprendizaje

1. Análisis de las amenazas y vulnerabilidades más comunes
  - a) Magerit
  - b) OWASP
  - c) EGSI
2. Práctica de laboratorio de análisis de vulnerabilidades OWASP-ZAP

## 14.2. Reflexión: Zona de Confort

La zona que usted conoce aquí no requiere más esfuerzo, pero ¿a cambio de qué?  
Al salir de esa zona de confort, se entra a una zona de pánico.

Le llaman la zona de pánico porque se desconocen las cosas, pero si usted sigue avanzando, poco a poco esa zona de pánico se convierte en zona de aprendizaje. Se va saliendo de la zona de pánico, hasta que se llega a una tercera zona, que ahora tiene dos pisos, y que se llama la zona de descubrimiento, la zona de crecimiento.



Figura 17: Modelo de zonas de desarrollo personal

## 14.3. Mia: Aprendizaje Activo

Para aprender realmente, es mejor hacer las cosas por uno mismo, equivocarse y ahí aprender.

El verdadero aprendizaje no es copiar y pegar, sino cuestionarse lo que se hace, lo que uno cree y la manera en que uno aprende. Esa curiosidad y el darle espacio a la experimentación es la mejor manera de aprender. Con la práctica se aprende, y a través de la práctica es como realmente se consolida el conocimiento.

## 14.4. Marco Teórico: Amenazas y Vulnerabilidades

- **Amenaza** (si no tiene backups o respaldos) + **vulnerabilidades** = **riesgo**
- **Amenazas Naturales:**
  - Fuego
  - Erupción volcánica
  - Movimientos telúricos
- **Amenazas:** Causa potencial de un incidente que puede causar daños en un sistema personal, institución o país
  - Humanas

- Intencionales (ej. incendios provocados)
- No intencionales (por omisión o negligencia)

■ **Procedimientos:**

- Técnicas (bugs, troyanos, spyware, gusanos)
- Tecnologías (cortocircuitos, obsolescencia, incumplimiento de normas)

■ **Vulnerabilidades:** Debilidad que puede ser explotada por un ciberataque interno o externo

**14.4.1. Ejemplos de vulnerabilidades comunes**

1. Mal funcionamiento y herramientas sin licencia
2. Gestión inadecuada de red o falla en los enlaces de comunicación
3. Destrucción de registros y respaldo irregular
4. Acceso a información restringida y compartir credenciales
5. Falta de monitoreo de red y ataques a páginas publicadas
6. Suspensión del servicio y actualización de parches del software base
7. Falta de monitoreo y ciberataques

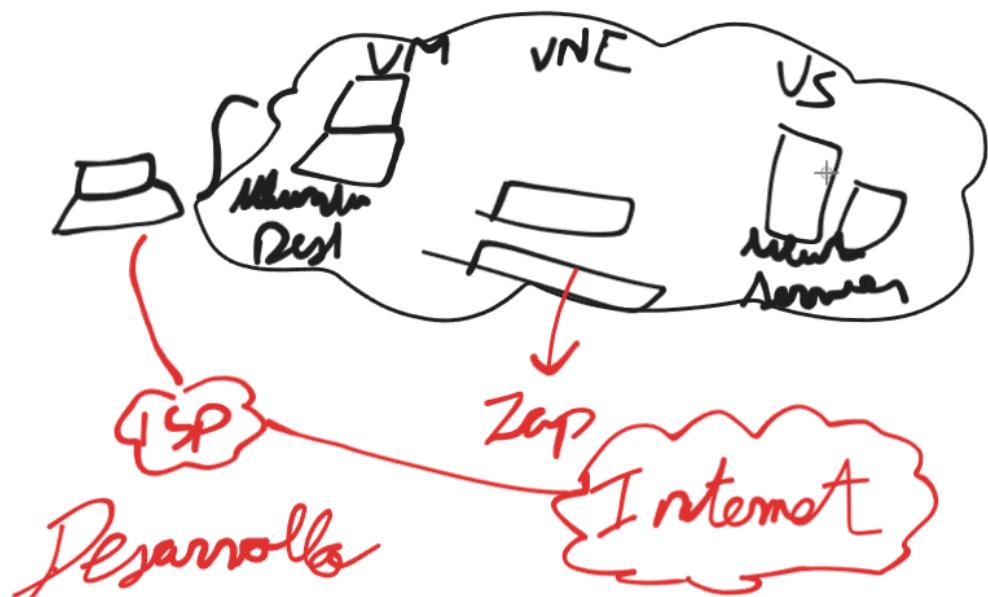


Figura 18: Diagrama de análisis de riesgos

## **14.5. ESQUEMA GUBERNAMENTAL DE SEGURIDAD DE LA INFORMACIÓN (EGSI)**

Puede encontrar más información en: <https://www.gobiernoelectronico.gob.ec/egsi/>

## **15. Práctica de laboratorio 6: Análisis de vulnerabilidades con OWASP-ZAP**

### **15.1. Objetivo de aprendizaje**

1. Comprender el funcionamiento de las herramientas de análisis de vulnerabilidades OWASP
2. Aprender a utilizar OWASP-ZAP para identificar amenazas y vulnerabilidades
3. Realizar escaneos activos y pasivos para detectar vulnerabilidades comunes en aplicaciones web

### **15.2. Marco teórico**

#### **15.2.1. OWASP**

Open Web Application Security Project (OWASP) es una comunidad abierta dedicada a permitir que las organizaciones desarrollen, adquieran y mantengan aplicaciones confiables.

#### **15.2.2. OWASP-ZAP (Zed Attack Proxy)**

Herramienta de seguridad para aplicaciones web que ayuda a encontrar vulnerabilidades como:

- Inyección SQL
- Cross-Site Scripting (XSS)
- Fugas de información
- Autenticación insegura

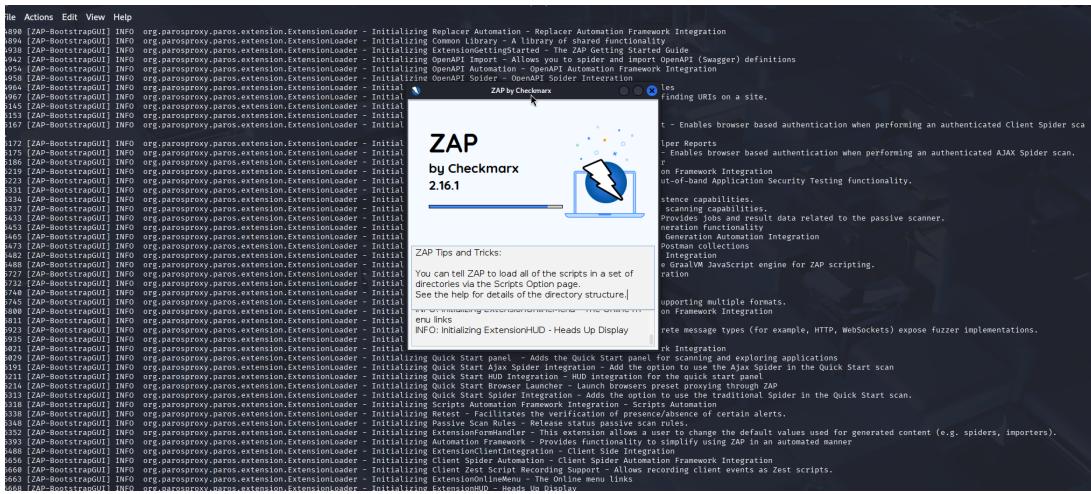


Figura 19: Interfaz de inicio de OWASP-ZAP

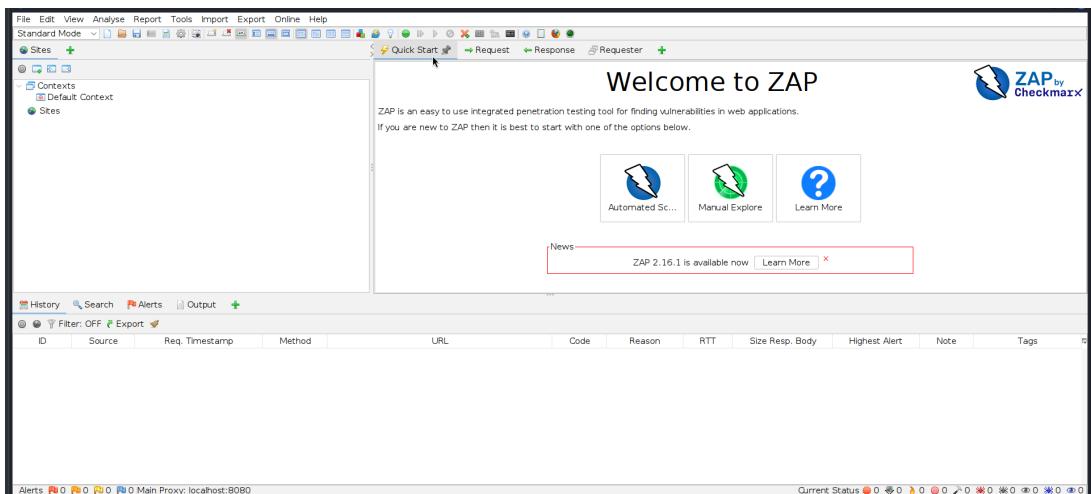


Figura 20: Vista general de la interfaz de OWASP-ZAP