CATALYST DIGITAL

RedDeOficios

Ciberseguridad

Solicitante: I.T.S - Instituto Tecnológico Superior Arias Balparda.
Nombre Fantasía, de la nueva empresa: Catalyst Digital
Grupo: 3°MN
Turno: Nocturno
Unidad Curricular: Ciberseguridad
Integrantes del grupo: Aguerre Leandro, Garay Joel, Olivera Enzo, Román Fabián, Rosas Kevin.

Fecha de entrega: 15/09/2025

Instituto Tecnológico Superior Arias Balparda.
Blvr. José Batlle y Ordóñez 3570 esq. Gral. Flores – Montevideo.

Índice

1. Amenazas específicas para nuestro sistema

Una lista de las amenazas que pondrían en peligro el sistema a desarrollar:

a. Inyección SQL (SQLI).
b. Inyección de código JavaScript(XSS).
c. Usuarios internos maliciosos.
d. Phishing.
e. Spam de usuarios legítimos del sistema.
f. Ataque de denegación de servicio (DoS/DDoS).

¿Por qué estas son amenazas para nuestro sistema?

Inyección SQL:
La Inyección SQL (SQLI) es una de las amenazas más comunes y peligrosas. Un atacante inserta código malicioso en los campos de entrada de un sitio web, como un formulario de inicio de sesión o un cuadro de búsqueda. Este código manipula la consulta a la base de datos, permitiendo al atacante:
Robar información: Puede ver, copiar o robar información sensible, como contraseñas de usuarios, datos de tarjetas de crédito o información personal.
Borrar o modificar datos: El atacante puede eliminar tablas enteras o alterar datos en la base de datos, causando la interrupción total del servicio.
Obtener acceso administrativo: Puede saltarse la autenticación e incluso obtener permisos de administrador, lo que le otorga control total sobre el sitio.

Inyección de JavaScript (XSS):
La Inyección de código JavaScript (XSS) ocurre cuando un atacante inyecta código malicioso en un sitio web. Este código se ejecuta en el navegador del usuario que visita el sitio. Esto permite al atacante:
Robar información del usuario: El código malicioso puede robar cookies de sesión, lo que le permite al atacante suplantar la identidad del usuario sin conocer su contraseña.
Redireccionar a sitios maliciosos: El atacante puede redirigir a los usuarios a páginas de phishing para robar credenciales.
Manipular el contenido de la página: Puede alterar el contenido del sitio web, engañando a los usuarios para que proporcionen información confidencial.

Usuarios internos maliciosos:
Los usuarios internos maliciosos son una amenaza porque ya tienen acceso legítimo al sistema. Pueden ser empleados descontentos, ex empleados que aún conservan acceso, o incluso atacantes que han logrado robar las credenciales de un empleado. Un usuario interno con malas intenciones puede:
Robar datos: Tiene acceso a información confidencial, propiedad intelectual o datos de clientes.
Modificar o borrar datos: Puede manipular o eliminar datos críticos, causando graves interrupciones en las operaciones.
Crear puertas traseras: Puede crear cuentas de usuario ocultas o instalar software malicioso para mantener el acceso incluso si su cuenta original es desactivada.

Phishing:
El Phishing no ataca directamente el sitio web, sino a sus usuarios. El atacante crea una página web falsa (o envía un correo electrónico fraudulento) que se asemeja al sitio real para engañar a los usuarios y que ingresen sus credenciales. Esta amenaza es peligrosa porque:
Compromete credenciales: El atacante roba las contraseñas de los usuarios, que luego puede usar para acceder a sus cuentas en el sitio web original.
Riesgo para la marca: El sitio web legítimo pierde la confianza de sus usuarios, ya que se asocia con el fraude y el robo de datos.

Spam:
Los usuarios malintencionados o bots automatizados pueden saturar el sistema con publicaciones, reseñas o mensajes falsos, generando ruido en la información, sobrecarga de datos innecesarios y una mala experiencia para los usuarios legítimos.

Ataque de denegación de servicio (DoS): Un Ataque de Denegación de Servicio (DoS) o Distribuido de Denegación de Servicio (DDoS) busca sobrecargar un servidor o una red con un gran volumen de tráfico, impidiendo que los usuarios legítimos accedan al sitio. Esto es una amenaza porque:

Interrupción del servicio: El sitio web queda inaccesible para los usuarios, lo que puede resultar en grandes pérdidas financieras para un negocio en línea.

Consumo de recursos: El ataque agota los recursos del servidor (ancho de banda, CPU), lo que puede afectar a otros servicios alojados en el mismo servidor.

Cortina de humo: A menudo, un ataque DDoS se utiliza como una distracción para que los atacantes puedan realizar otras actividades maliciosas, como robar datos, mientras el personal de seguridad está ocupado lidiando con el ataque.

2. Posibles vulnerabilidades en infraestructura (servidor, base de datos, red interna).

Una lista de las vulnerabilidades que surgen en un sistema de este estilo:

a. Servidor: Puertos abiertos.
b. Servidor: Servicios activos que no se usan (FTP, correo).
c. Servidor: Permisos de archivos / carpetas mal configurados.
d. Servidor: Falta de parches de seguridad.
e. Base de datos: Contraseñas almacenadas de forma insegura.
f. Base de datos: Permisos excesivos para usuarios o procesos,
g. Red interna y comunicaciones: Certificados SSL/TLS inválidos o mal configurados.
h. Red interna y comunicaciones:Uso de HTTP en vez de HTTPS.
i. Copias de seguridad:No se realizan.
j. Copias de seguridad:No se prueban.
k. Copias de seguridad:Almacenadas en el mismo servidor que los datos originales.
l. Contraseñas de usuarios internos débiles.

¿Por qué estas serían vulnerabilidades para nuestro sistema?

Puertos abiertos y servicios activos no utilizados: Un servidor con puertos abiertos que no se usan o servicios como FTP o correo sin función activa, es como dejar una puerta sin llave en una casa. Los atacantes escanean estos puertos en busca de puntos de entrada para explotar vulnerabilidades conocidas en esos servicios, incluso si no se están utilizando activamente.

Permisos de archivos/carpetas mal configurados: Si un atacante logra explotar una vulnerabilidad, por ejemplo, en una página web, puede intentar acceder a otros archivos del servidor. Si los permisos de esos archivos o carpetas no están bien configurados, el atacante podría leer, modificar o incluso ejecutar archivos sensibles del sistema, lo que podría llevar a un control total del servidor.

Falta de parches de seguridad: Los desarrolladores de software constantemente liberan parches para corregir fallas de seguridad. Si un servidor no tiene estos parches actualizados, está expuesto a vulnerabilidades conocidas que los atacantes pueden explotar fácilmente. Esta es una de las principales razones de los ciberataques.

Contraseñas almacenadas de forma insegura: Almacenar contraseñas sin **hashing** o **salting** las deja expuestas a un robo masivo. Si un atacante logra acceder a la base de datos, las contraseñas estarían en texto plano o fácilmente descifrables, permitiendo el acceso no solo a ese sitio, sino también a otras cuentas que usen la misma contraseña.

Permisos excesivos para usuarios o procesos: Otorgar permisos de administrador a usuarios o procesos que solo necesitan realizar tareas básicas aumenta el riesgo. Si esa cuenta o proceso es comprometido, el atacante obtiene permisos elevados y podría robar o borrar la totalidad de la base de datos.

Certificados SSL/TLS inválidos o mal configurados: El cifrado SSL/TLS es lo que hace que una conexión sea segura. Un certificado inválido o mal configurado podría no cifrar el tráfico correctamente, exponiendo datos sensibles a ataques de "man-in-the-middle" donde un atacante intercepta la información en tránsito. Esto permite el robo de datos como contraseñas, información de tarjetas de crédito, etc.

Uso de HTTP en lugar de HTTPS: HTTP transmite datos en texto plano, sin cifrado. Cualquier persona en la misma red (como en una red Wi-Fi pública) podría usar un sniffing (rastreo de paquetes) para ver toda la información que se envía, incluyendo contraseñas y otros datos privados.

No se realizan o no se prueban: No tener copias de seguridad es un riesgo masivo. En caso de un ataque de ransomware, un borrado accidental o un fallo del sistema, no habría forma de recuperar los datos. Si las copias de seguridad no se prueban, no hay garantía de que funcionen cuando se necesiten, lo que anula su propósito.

Almacenadas en el mismo servidor: Si las copias de seguridad se guardan en el mismo lugar que los datos originales, un ataque al servidor principal podría comprometer tanto los datos en vivo como las copias de seguridad. La copia de seguridad debe ser independiente y estar en un lugar diferente para ser un verdadero respaldo.

Contraseñas de usuarios internos débiles: Las contraseñas de usuarios internos (como empleados, administradores de TI o personal de seguridad) son particularmente vulnerables a ataques de fuerza bruta o ataques de diccionario.
Ataque de diccionario: El atacante usa una lista precompilada de palabras, frases y combinaciones de caracteres comunes para adivinar la contraseña. Este método es muy efectivo contra contraseñas débiles como "123456", "contraseña", "qwerty", nombres de mascotas, o fechas de nacimiento.
Ataque de fuerza bruta: El atacante utiliza un software para probar sistemáticamente todas las combinaciones posibles de caracteres hasta encontrar la correcta. Si la contraseña es corta y simple, el tiempo para descifrarla es mínimo.
Impacto de un ataque: Si un atacante logra comprometer las credenciales de un usuario interno, puede obtener privilegios elevados y acceder a información confidencial, modificar archivos, instalar malware, o incluso crear nuevas cuentas de administrador para mantener su acceso.

3. Políticas y medidas de seguridad

Respecto a las amenazas y vulnerabilidades mencionadas anteriormente se tomarán las siguientes políticas y medidas de seguridad para evitar accesos no autorizados, spam, prevenir ataques y la pérdida de datos.

- **Política general:**
  Todas las acciones críticas (inicio de sesión, edición de datos y su eliminación, cambio de roles, asignación y quita de permisos) deberán quedar registradas en un sistema de auditoría para su posterior revisión.
  Revisiones mensuales de cuentas activas y permisos asignados.

- **Políticas (Inyección de código SQL):**
  Ningún dato ingresado por el usuario será procesado sin validación y sanitización previa.
  Medidas:
  Uso de consultas preparadas (prepared statements).
  Validación de entradas en frontend y backend (whitelist de caracteres permitidos).
  Restricción de privilegios de las cuentas de la base de datos (principio de mínimo privilegio).

- **Políticas (Inyección de código XSS):**
  No se permitirá mostrar contenido dinámico del usuario sin ser previamente validado/escapado.
  Medidas:
  Escapar siempre el HTML generado dinámicamente.
  Usar funciones de sanitización (ej: htmlspecialchars en PHP).
  Restringir campos que aceptan HTML sólo a caracteres seguros.

- **Políticas contra usuarios internos maliciosos:**
  El acceso del personal deberá estar limitado estrictamente a las funciones necesarias.
  Medidas:
  Definición clara de roles y permisos (cliente, proveedor, administrador, moderador).

- **Políticas contra phishing:**
  Educar al personal y aplicar medidas de control para reducir riesgos.
  Medidas:
  Prohibido usar correos laborales con fines personales.
  Verificación obligatoria de remitentes antes de abrir adjuntos.

- **Políticas contra Spam:**
  El sistema limitará automáticamente las acciones repetitivas para proteger la plataforma.
  Medidas:
  Límite de publicaciones diarias.
  Solo una reseña por servicio contratado.
  Reporte de spam por parte de otros usuarios.
  Implementación de reCAPTCHA en formularios.
  Estará prohibido compartir enlaces hacia otras paginas web dentro de la aplicación.

- **Políticas frente ataques Dos/DDoS:**
  El sistema deberá estar protegido contra tráfico anómalo o abusivo.
  Medidas:
  Firewall: iptables/ufw para bloquear direcciones IP sospechosas.
  Fail2Ban para bloqueo automático de intentos repetidos.
  Protección en la nube Cloudflare (plan gratuito):
    Filtra tráfico malicioso antes de que llegue al servidor.
    Ofrece mitigación básica de DDoS.
    Incluye protección contra bots.
  Google reCAPTCHA en formularios: evita que los bots usen la app para generar carga extra.
  Limitaciones en la aplicación:
    Rate limiting: limitar cuántas peticiones puede hacer un usuario en un período de tiempo (ej: máximo 10 solicitudes por segundo desde una misma IP).
    Bloqueo progresivo: si un usuario intenta hacer demasiadas solicitudes, se lo bloquea temporalmente.
- **Políticas de configuración del servidor:**
  El servidor deberá configurarse siguiendo buenas prácticas de seguridad.

Medidas:
Servicios del servidor que no se usen por el sistema serán deshabilitados.
Cerrar puertos no usados.
Configuración correcta de permisos en archivos/carpetas.
Se actualizarán periódicamente los parches de seguridad del servidor y aplicaciones que se ejecuten dentro de este.

- **Políticas sobre comunicaciones inseguras.**
  Toda comunicación entre usuarios y servidor deberá ser cifrada.
  Medidas:
  Uso obligatorio de HTTPS.
  Certificados SSL/TLS válidos y actualizados.
  HSTS (HTTP Strict Transport Security) habilitado en el servidor.

- **Políticas de sesiones**
  Las sesiones con 20 minutos de inactividad serán cerradas para evitar tráfico extra y por seguridad.

- **Políticas de copias de seguridad.**
  Las copias de seguridad deberán realizarse periódicamente y almacenarse en un entorno separado.

Medidas:

Que respaldar	Base de datos / archivos importantes.	Sistema web / configuraciones
Cuando respaldar y qué tipo de respaldo	Diario: Incremental Mensual: Diferencial Trimestral: Completo	RESPALDO COMPLETO: En caso de actualización del sistema web o cambios de configuración, se realizará un respaldo anterior a la actualización .
Donde almacenar	En la nube / Servidor de respaldo/ Físico en caja fuerte.	En la nube / Servidor de respaldo/ Físico en caja fuerte.
Quien accede	Usuario del servidor “backup” o “administrador”	Usuario del servidor “backup” o “administrador”
Verificación	El administrador verificará las copias de seguridad creadas: Semanalmente en caso de las diarias / Finalizado el respaldo (Diferencial o Completo) se verificará su validez.	El administrador verificará el respaldo una vez se haya realizado.

4. Definición de políticas de contraseñas

a. Políticas de contraseñas para usuarios que usen la plataforma web:

i. **Composición:** Las contraseñas deberán contar con una combinación de:
   1. **Letra mayúscula** (A-Z): 1 como mínimo.
   2. **Letras minúsculas** (a-z):1 como mínimo.
   3. **Números:** 1 como mínimo.
   4. **Carácteres especiales:** " !, @, #, $, %, &, *, (,), -, _, +, = " 1 como mínimo.

ii. **Longitud:** 8 dígitos como mínimo para clientes y proveedores y 12 para administradores y moderadores.

iii. **Original:** No se permite que el nombre de usuario, correo o dato personal se utilice para la contraseña.

iv. **Reutilización:** No se permite la reutilización de las últimas 5 contraseñas.

v. **Las contraseñas caducan:** Cada 12 meses (en caso de administrador o moderador cada 3 meses).

vi. **Almacenamiento:** Las contraseñas se guardan hasheadas.

vii. **Bloqueo:** Se bloqueará la cuenta temporalmente (15 minutos) tras 3 intentos fallidos de ingreso, si luego de transcurrido el tiempo vuelve a fallar 3 intentos quedará la cuenta bloqueada permanentemente, desbloqueando solo a través de un mail a su correo.

viii. **2FA:** Se permitirá el uso de un 2FA (obligatorio para administradores y moderadores).

b. Políticas de contraseñas para usuarios que usen el servidor:

i. **Composición:**Las contraseñas deberán contar con una combinación de:
   1. Letra mayúscula A-Z, cantidad 3 como mínimo.
   2. Letras minúsculas a-z cantidad 3 como mínimo.
   3. Números, cantidad 4 como mínimo.
   4. Carácteres especiales permitidos: " !, @, #, $, %, &, *, (,), -, _, +, = " cantidad 2 como mínimo.

ii. **Longitud:** 16 dígitos como mínimo.

iii. **Original:**No se permitirá que el nombre de usuario, correo o dato personal se utilice para la contraseña.

iv. **Reutilización:** No se permite la reutilización de las últimas 5 contraseñas.

v. **Las contraseñas caducan:** Las contraseñas caducan cada 3 meses.

vi. **Almacenamiento:** Las contraseñas almacenadas se guardan hasheadas.

vii. **Bloqueo:** Se bloqueará la cuenta tras 3 intentos fallidos de ingreso, desbloqueandose solo por un administrador.

viii. **Las contraseñas visibles:** Personal que utilice la plataforma web como el servidor no podrán tener sus contraseñas visiblemente expuestas.

ix. **Compartir cuentas:** Estará prohibido que el personal comparta cuentas y contraseñas.

5. Controles de acceso y roles de usuarios

a. Usuarios del entorno web:

i. Clientes (usuario externo a la empresa)

Permisos:

Edición de sus datos personales.

Tareas:

Contratar y agendar servicios publicados, con posibilidad de cancelarlos.

Enviar mensajes a proveedores.

Realizar reseñas de servicios contratados.

1. Login con usuario y contraseña
2. Sesiones expiran tras 20 minutos de inactividad

ii. Proveedores (usuario externo a la empresa)

Permisos:

Edición de sus datos personales.

Tareas:

Realizar publicaciones, editarlas y eliminarlas.

Crear calendario de disponibilidad.

Aceptar contratos de servicios.

Responder mensajes de clientes.

1. Login con usuario y contraseña
2. Sesiones expiran tras 20 minutos de inactividad

iii. Administradores (usuario de la empresa)

Permisos:
Total sobre la plataforma web.
Tareas:
Resolverán errores de alto nivel.
Actualización y eliminación de datos.
Actualización de configuraciones.
Actualizaciones de futuras versiones del sistema.
Actualización de parches de seguridad.
Verificarán respaldos y logs de errores.
Actuarán ante intentos de ciberataques a la plataforma.
Revisaran

1. Login con usuario y contraseña
2. Sesiones expiran tras 10 minutos de inactividad
3. 2FA
4. Acceso a puertos USB
5. Acceso a sala de servidores a través de huella digital.
6. Acceso a mail propio interno de la empersa.
7. Acceso al mail interno de soporte de la empersa.

iv. Moderadores (usuario de la empresa)
Permisos:
Edición y eliminación de datos.
Tareas:
Estos se encargan de procesar solicitudes de los usuarios de la plataforma.
Evacuaran dudas, recibirán errores reportados, trataran las denuncias de publicaciones y moderarán el contenido publicado en base a determinadas reglas.
Resolverán errores de nivel medio y bajo

1. Login con usuario y contraseña
2. Sesiones expiran tras 10 minutos de inactividad
3. 2FA
4. Deshabilitación de puertos USB
5. Sin acceso a la sala de servidores.
6. Acceso a mail propio interno de la empersa.
7. Acceso al mail interno de soporte de la empersa.

b. Usuarios del servidor:

i. Administrador

Acceso total a todo el sistema de archivos.
Puede crear, eliminar y modificar cualquier archivo o configuración.
Permite administrar servicios, usuarios, redes y hardware.
Puede instalar y desinstalar software.
Usuario sólo para ser usado antes problemas graves o realizar tareas críticas por razones de seguridad.
Toda actividad con este usuario será registrada.

1. Login con usuario y contraseña
2. Sesiones expiran tras 10 minutos de inactividad

ii. adminweb

Puede ejecutar comandos administrativos usando sudo.
Puede reiniciar servicios como Apache, MySQL, etc.
Permiso de lectura y escritura en archivos de configuración del sistema.
Puede acceder a los logs del sistema.
No tiene acceso directo sin autorización a cuentas sensibles como root.

1. Login con usuario y contraseña
2. Sesiones expiran tras 10 minutos de inactividad

iii. desarrollador

Acceso de lectura y escritura sobre los archivos del proyecto web.
No tiene permisos para reiniciar servicios, modificar configuraciones del sistema ni acceder a otras carpetas del sistema operativo.
Puede subir o modificar archivos del backend y frontend.
Ideal para tareas de programación sin comprometer la seguridad del servidor.

1. Login con usuario y contraseña
2. Sesiones expiran tras 10 minutos de inactividad

iv. srvweb (cuenta de servicio automático)

Este usuario ejecuta el servidor web.
Permisos de lectura y escritura solo en las carpetas necesarias del sitio.
No tiene permisos sobre otras partes del sistema ni acceso a comandos administrativos.
Su acceso está restringido para minimizar daños si el sitio web es vulnerado.
Solo se usará para configuraciones y su actividad en estos casos será registrada.

1. Login con usuario y contraseña
2. Sesiones expiran tras 10 minutos de inactividad

v. mysql (cuenta de servicio automático)

Utilizado exclusivamente por el servicio MySQL/MariaDB.
Tiene acceso solo a su carpeta de datos y archivos de configuración.
No puede ejecutar comandos fuera del entorno de base de datos.
Solo se usará para configuraciones y su actividad en estos casos será registrada.

1. Login con usuario y contraseña
2. Sesiones expiran tras 10 minutos de inactividad

vi. backup (cuenta de servicio automático)

Este usuario realizará de forma automática los backups de información necesarios.
Permisos de lectura sobre archivos críticos como los del sitio web y base de datos.
Permisos de escritura en una carpeta de destino para almacenar copias de seguridad.
No puede ejecutar tareas administrativas ni acceder a configuraciones del sistema.
Solo se usará para configuraciones de futuros backups y su actividad en estos casos será registrada.

1. Login con usuario y contraseña
2. Sesiones expiran tras 10 minutos de inactividad

Cuadro resumen:

Rol	Acceso	Sesión	2FA	Permisos clave
Cliente	Web	20 min	No	Contratar, reseñas, mensajes
Proveedor	Web	20 min	No	Publicar servicios, responder mensajes
Moderador	Web	10 min	Sí	Moderar publicaciones, atender reportes
Administrador	Web	10 min	Sí	Acceso total, parches, logs, copias
adminweb	Server	10 min	No	sudo, reinicio de servicios
desarrollador	Server	10 min	No	Modificar frontend/backend
srvweb (servicio)	Server	N/A	N/A	Ejecutar servidor web
mysql (servicio)	Server	N/A	N/A	Manejo de BD
backup (servicio)	Server	N/A	N/A	Copias automáticas

6. Validaciones aplicadas

Frontend (navegador del usuario):

- Validación de formularios (campos requeridos, formato de correo válido, longitudes mínimas/máximas).
- Validación de contraseñas (longitud, uso de mayúsculas, números, símbolos).
- Uso de *Google reCAPTCHA* para evitar bots.
- Sanitización de entradas antes de enviarlas al servidor (ej. evitar caracteres peligrosos como `<script>`).

Backend (servidor PHP/MySQL):

- Validación duplicada: Significa que todo lo que se valida en el frontend se vuelve a validar en el backend.
- Uso de consultas preparadas en SQL para prevenir inyecciones.
- Validación de tipos de datos (ej. precios solo numéricos, fechas en formato correcto).
- Limitación de tamaño de archivos subidos (imágenes de servicios).
- Sanitización de datos antes de almacenarlos en BD o mostrarlos en HTML.

7. Encriptación de contraseñas aplicada en RedDeOficios

Hashing seguro con bcrypt o Argon2
Uso de *salts* automáticos para cada contraseña.

En servidor: nunca almacenar contraseñas en texto plano.

8. Métodos de registro de auditoría

Registro de actividad (logs) en servidor:

- Accesos exitosos y fallidos (se registrará de manera independiente los logs de inicio de sesión de usuarios de la empresa y los visitantes de la página).
- Modificaciones de usuarios, publicaciones y reseñas (se registran en logs independientes).
- Intentos de inyección o accesos no autorizados detectados.

Auditoría de base de datos:

- Tabla “auditoria” que registre: usuario, acción, fecha/hora, tabla/registro afectado.

Logs del sistema:

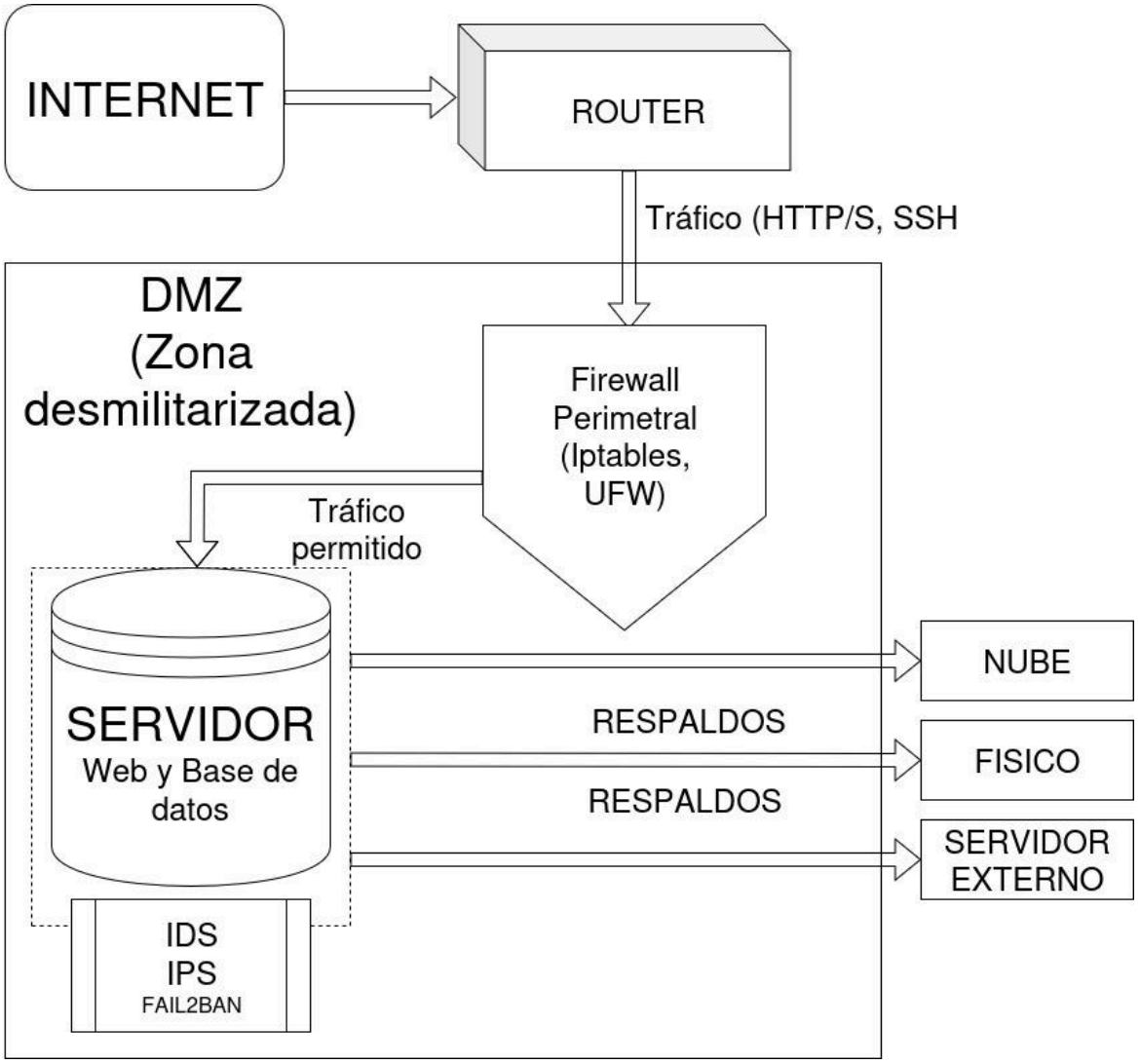
- Apache: Registros de accesos y errores.
- IDS: Registro de las alertas de detección enviadas.
- Fail2Ban: Intentos de autenticación bloqueados.
- IPS: Registro de amenazas bloqueadas.
- Iptables: Intentos de conexión bloqueados.

Retención mínima: 6 meses para análisis forense.

9. Esquema de red con implementaciones de seguridad

- **Servidor web y base de datos**

- **Firewall perimetral:** Iptables, UFW.
- **Router**
- **Protección interna:** IPS, IDS, Fail2Ban
- **Respaldos:** Nube, físico, servidor externo.



10. Implementaciones de seguridad para servidores

Firewall (iptables/ufw):

Iptables: Funciona como un firewall para definir y gestionar reglas que controlan el tráfico de red, permitiendo o bloqueando paquetes de datos según criterios específicos

UFW: Es una herramienta de línea de comandos diseñada para simplificar la configuración de iptables. Es una interfaz amigable que traduce comandos sencillos a las reglas complejas de iptables.

Fail2Ban:

Herramienta de software de código abierto que se utiliza para prevenir ataques de fuerza bruta en servidores. Actúa como un escudo de seguridad que monitorea los archivos de registro (logs) de tu servidor en busca de intentos de inicio de sesión fallidos repetidamente.

IDS/IPS (Intrusion Detection/Prevention System):

La herramienta Snort de uso gratuito que cuenta con IDS e IPS se usará para monitorear y alertar sobre amenazas, está detecta tráfico anómalo o ataques comunes.

- IDS: Es un sistema pasivo que se encarga de monitorizar el tráfico de la red en busca de comportamientos maliciosos o sospechosos. Si detecta una amenaza, su única acción es alertar a los administradores de la red.
- IPS:Es un sistema proactivo que no solo detecta amenazas, sino que también actúa automáticamente para bloquearlas. Se interpone en el flujo de tráfico, revisa los paquetes de datos y, si encuentra una actividad maliciosa, la detiene de inmediato.

11. Registro de simulación de ataques al sistema

Hoja testigo:
