Lecciones aprendidas y recomendaciones de seguridad para la prestación de servicios electrónicos

Abstract: *el incremento en la prestación de servicios electrónicos por parte de las diferentes organizaciones, a través de sedes electrónicas, servicios a los ciudadanos y portales del empleado permiten a los usuarios de los mismos realizar múltiples tramites a través de internet. La información intercambiada en estos servicios hace que la seguridad de los mismos sea un factor determinante a la hora de implementarlos y desplegarlos.*

Contenido:

1. OBJETO 1
2. SEGURIDAD EN LA INFRAESTRUCTURA 2
2.1 MÍNIMA EXPOSICIÓN 2
2.2 EXPOSICIÓN SEGURA 3
2.3 SEGURIDAD PERIMETRAL 4
2.3.1 Medidas de seguridad en IPS/IDS 5
2.3.2 Medidas de seguridad en firewalls 6
2.3.3 Medidas de seguridad en WAF 7
2.4 PRODUCTOS BAJO SOPORTE DE FABRICANTE 8
2.5 CONFIGURACIÓN SEGURA DE LOS SISTEMAS 9
2.6 VIGILANCIA CONTINUA DEL SISTEMA 10
2.7 ALTA DISPONIBILIDAD 11
2.8 RESPALDO DE LAS CONFIGURACIONES 11
3. SEGURIDAD EN LA APLICACIÓN 13
3.1 CICLO DE DESARROLLO SEGURO (S-SDLC) 13
3.2 SISTEMAS DE AUTENTICACIÓN. GESTIÓN DE IDENTIDADES 14
3.3 DEFINICIÓN DE ROLES Y PRIVILEGIOS 16
3.4 AUDITORÍA DE ACCIONES 18
4. REFERENCIAS 19

1. OBJETO

En la actualidad, cada vez es más frecuente que los organismos del Sector Público ofrezcan la posibilidad de realizar los diferentes trámites necesarios para los servicios que proporcionan a los ciudadanos a través de internet.

Lo que hace unos años implicaba un desplazamiento a las ubicaciones de la administración correspondiente para realizar cualquier gestión sobre un servicio ofrecido, hoy en día se puede realizar de forma telemática gracias a los servicios electrónicos como pueden ser sedes electrónicas, servicios a los ciudadanos, portales de empleados, etc.

La flexibilidad y facilidad que proporciona un servicio electrónico ha hecho que los ciudadanos demanden cada vez más este tipo de servicios. **El crecimiento del uso de estos servicios también ha hecho que los atacantes pongan su punto de mira en estos sistemas y el número de ciberataques haya aumentado considerablemente sobre estos.**

La información transmitida en los servicios ofrecidos, en la mayoría de los casos de carácter personal, hace que la seguridad de los sistemas que la componen sea un aspecto crítico a tener en cuenta a la hora de implementar este tipo de servicio.

2. SEGURIDAD EN LA INFRAESTRUCTURA

Las infraestructuras sobre las que se despliegan los servicios electrónicos son uno de los puntos más importantes. Los componentes que forman parte de estos servicios, como pueden ser servidores web, servidores de aplicaciones y bases de datos entre otros, deben estar correctamente securizados para protegerse frente a posibles amenazas.

Una potencial vulnerabilidad puede provocar que se comprometa toda la información manejada por estos sistemas. **Para poder mantener un correcto nivel de seguridad en la infraestructura es necesario proteger los elementos que la componen, tanto los que se sitúan internamente como externamente**.

A la hora de implementar cualquier tipo de servicio al exterior, se debe cumplir con el principio de mínima exposición y garantizar la seguridad de dichos servicios. Cualquier elemento de la infraestructura debe estar correctamente actualizado y cumplir con los ciclos de actualización recomendados por los fabricantes, ya que una tecnología, sistema o producto sin actualizar supone un aumento del riesgo en la superficie de exposición.

De manera complementaria, es aconsejable implementar sistemas de protección perimetral como pueden ser IDS (*Intrusion Detection System*), IPS (*Intrusion Prevention System*) y cortafuegos de red. Estos sistemas detectarán y protegerán la infraestructura frente a potenciales ataques y permitirán filtrar las comunicaciones para, de esta manera, reducir la exposición de la organización.

2.1 MÍNIMA EXPOSICIÓN

Una de las máximas que se deben seguir en la infraestructura a la hora de implementar un servicio electrónico es la correcta segregación de servicios. Debe existir un aislamiento entre los servicios que se exponen al usuario y los servicios necesarios para el correcto funcionamiento de un servicio electrónico. Siguiendo el principio de mínima exposición, únicamente se deben exponer los servicios estrictamente necesarios.

Por lo general, un servicio electrónico presenta una aplicación web para que los ciudadanos puedan llevar a cabo los diferentes trámites, por tanto, sólo se debería exponer el servicio HTTPS para acceder a los servicios. Por compatibilidad con clientes web antiguos es común hacer uso del protocolo HTTP, pero este sólo deber ser usado

para redirigir a la versión HTTPS a través de la cual todas las comunicaciones se realizan de manera segura, siempre y cuando la configuración de este servicio y sus componentes se realice de manera correcta.

No se debe publicar de manera externa ningún servicio que, aun siendo necesario para el correcto funcionamiento de la aplicación, no deba ser usado directamente por los usuarios del servicio. Los servicios de aplicación (*backend* del servicio electrónico), de bases de datos o de administración remota son normalmente necesarios para el correcto funcionamiento de estos sistemas, pero nunca deben ser expuestos a internet, ya que suponen un aumento de la superficie de exposición de la organización, siendo vectores de entrada a potenciales ataques. Estos servicios sólo se deben emplear de manera interna y en el caso de tener que acceder a estos de manera remota, se deben implementar mecanismos de acceso remoto seguros[1].

De la misma manera, sólo se deben exponer los servicios correspondientes a la infraestructura en producción. **Un error frecuente es publicar servicios que se encuentran en entornos de preproducción**. Estos entornos suelen ser potencialmente más vulnerables ya que no se corresponden con versiones finales. Generalmente estos servicios se exponen de manera temporal para realizar pruebas, pero comúnmente, tras la finalización de las mismas, los servicios permanecen indebidamente publicados.

Para evitar estos errores, se recomienda la realización de auditorías de seguridad externas de manera periódica, con el objetivo de identificar puertos y servicios no controlados por los administradores, además de potenciales vulnerabilidades que pudieran existir en los productos, servicios y aplicaciones expuestos en la infraestructura.

2.2 EXPOSICIÓN SEGURA

Desde el punto de vista de la seguridad, es igual de importante seguir el principio de mínima exposición, exponiendo la menor cantidad de servicios con el objetivo de reducir la superficie de exposición, como que **los servicios publicados se implementen de manera segura y hagan uso de protocolos robustos**.

Como se ha indicado anteriormente, por lo general, los servicios electrónicos ofrecen todas sus funcionalidades a través del protocolo HTTPS. A través de este protocolo, comúnmente mediante el puerto 443, se sirve la aplicación con la cual los ciudadanos realizarán sus trámites. Para que este servicio sea seguro se recomienda aplicar las siguientes configuraciones en el servidor:

- No dar soporte al protocolo SSL en ninguna de sus versiones, ya que se consideran débiles.

[1] https://www.ccn-cert.cni.es/informes/abstracts/6230-recomendaciones-para-acceso-remoto-seguro-a-informacio-n-sensible-preservando-seguridad-y-productividad/file.html

- Deshabilitar las versiones de TLS 1.0 y 1.1, implementando sus versiones más robustas TLS 1.2 o 1.3.
- Deshabilitar suites de cifrados débiles como las basadas en el algoritmo RC4 o las que hacen uso de cifrados con bloques de 64 bits.
- Implementar la característica TLS_FALLBACK_SCSV.
- No permitir la compresión a nivel de TLS.
- Establecer un orden de preferencia de algoritmos de cifrado en el lado del servidor para asegurar el uso del mejor cifrado.
- Reconfigurar el servidor para no permitir la renegociación de los algoritmos de cifrado utilizados cuando sea iniciada por el cliente.
- Dejar de soportar suites de cifrado no PSK que emplean una clave RSA de transporte.
- Emplear longitudes de clave de 2048 o mayores para los algoritmos Ephemeral Diffie-Hellman (EDH).
- Configurar un certificado válido en el servidor que cumpla las siguientes condiciones:
    - El certificado debe estar emitido para el nombre de dominio correspondiente.
    - Se debe utilizar un certificado que esté en vigor según el periodo establecido para el mismo y cuyo período de validez sea igual o inferior a 397 días.
    - El certificado debe estar generado por una autoridad certificadora confiable que implemente la característica de seguridad OCSP Must-Staple y cuya firma se haya realizado con un algoritmo de hash robusto.
    - Se debe revisar la cadena de certificados para asegurar que el orden es "Certificado emisor-Certificados intermedios-Certificado final" y que no contiene certificados duplicados.

2.3 SEGURIDAD PERIMETRAL

Dentro de la seguridad perimetral que se puede implantar en la infraestructura de red existen distintos niveles atendiendo a las diversas funcionalidades y los cometidos de las diferentes tecnologías.

Es recomendable utilizar una arquitectura por niveles y no un dispositivo que tenga la capacidad de agrupar todas estas funcionalidades: IPS/IDS, Firewall y WAF. La utilización de un dispositivo que agrupe todas estas características, puede suponer un riesgo de seguridad y disponibilidad al tener concentradas todas las funcionalidades en un mismo punto.

Estos dispositivos deben configurarse correctamente para aportar una capa de seguridad a la infraestructura. Para ello, se deben seguir las siguientes recomendaciones comunes a todos los dispositivos cuyo propósito sea la seguridad perimetral de la infraestructura:

- El personal que opera y administra estos dispositivos debe estar debidamente formado en las tecnologías de los mismos.
- Se debe realizar una correcta definición de roles de usuario, proporcionando roles de administración únicamente a los usuarios estrictamente necesarios.
- Debe existir trazabilidad de los usuarios, especialmente de los que tengan roles de administración.
- Se debe disponer de un equipo dedicado a la respuesta frente a incidentes, así como de documentación y procedimientos frente a incidentes de seguridad.
- La administración se debe llevar a cabo a través de protocolos seguros. Se aconseja también cambiar los puertos estándar empleados para la administración.
- Se debe realizar la integración de múltiples factores de autenticación (MFA).
- Las reglas de configuración de los dispositivos se deben personalizar adaptándose a las necesidades de la organización.
- Se debe contar con un respaldo externo de la configuración de los dispositivos, para protegerse frente a cualquier imprevisto que implique realizar la configuración de los dispositivos de nuevo.
- El proceso de actualización del firmware, parches de seguridad y firmas del fabricante debe realizarse de manera continua.
- Es aconsejable contratar el servicio de soporte del fabricante.
- Realización de auditorías periódicas sobre estos dispositivos, con el objetivo de localizar las potenciales vulnerabilidades que pueden aparecer en ellos.

2.3.1 Medidas de seguridad en IPS/IDS

Los dispositivos IDS (*Intrusion Detection System*) e IPS (*Intrusion Prevention System*) en términos generales, son los elementos encargados de la detección y mitigación de ciberataques y el bloqueo de nuevas amenazas.

Las principales funcionalidades de un IDS son:

- Detectar un incumplimiento de la política de seguridad.
- Sistema de alertas frente a la detección de patrones maliciosos.
- Monitorización del usuario para detectar acciones maliciosas.
- Monitorización de configuraciones del sistema.

- Monitorización del tráfico de red entrante y saliente de los dispositivos.
- Comparación de archivos mediante firmas de malware.
- Detección de *DoS* (*Denial of Service*) y *DDoS* (*Distributed Denial of Service*).

Las funcionalidades principales de un IPS son:

- Eficacia para bloquear los vectores de ataques conocidos mediante firmas.
- Inspección proactiva del tráfico entrante en la red.
- Bloqueo de paquetes maliciosos.
- Bloqueo de direcciones IP maliciosas manualmente y por geolocalización.
- Sistema de alertas frente a posibles amenazas.
- Utilización de reglas personalizadas para el bloqueo de ciberataques.
- Bloqueo de ciberataques de tipo DoS (*Denial of Service*) y DDoS (*Distributed Denial of Service*).

El escenario recomendable es la utilización de las dos (2) tecnologías dado que un IDS es un sistema detección de amenazas y necesita ser complementado con un IPS para el bloqueo de las mismas. Ambos son elementos expuestos a internet, la primera línea de defensa en la infraestructura de red y, por ello, es importante disponer de personal cualificado a la hora de operar y administrar estas tecnologías. Esto es necesario ya que, si se definen unas políticas excesivamente agresivas se pueden producir falsos positivos y bloquear tráfico legítimo indebidamente, o en caso contrario, una excesiva permisividad en las políticas puede limitar las capacidades de estos dispositivos para detener un ciberataque, pudiendo quedar de esta manera comprometida la información de la organización.

2.3.2 Medidas de seguridad en firewalls

Un firewall o cortafuegos es el elemento intermediario encargado de proteger un dispositivo o subred de una intrusión proveniente desde otro dispositivo o subred. Es un sistema que tiene la capacidad de filtrar los paquetes de datos permitiendo o denegando las comunicaciones.

En un diseño óptimo de una infraestructura de red, un firewall sería el segundo elemento físico o lógico en la línea de defensa de la infraestructura de la organización. Esta tecnología se puede emplear tanto para reducir el riesgo de exposición frente a una amenaza externa como para un posible ataque interno.

Asimismo, se pueden emplear configuraciones de seguridad inherentes a la tecnología como:

- Filtrado de flujos de comunicación basados en reglas.

- Dispositivo con función de puerta de enlace.
- Segmentación lógica de la red.
- Utilización de DMZ para servicios expuestos a internet.
- Fortificación de la seguridad de los dispositivos o subsistemas detrás del firewall.
- Denegación de paquetes TCP fuera de estado (un paquete fuera de estado es un paquete que no sigue el orden establecido en el establecimiento de una conexión TCP).

2.3.3 Medidas de seguridad en WAF

Un WAF (*Web Application Firewall*) es un firewall con un cometido más específico y menos amplio que un firewall tradicional, orientado a proteger las aplicaciones web de la organización, de nueva generación (NGFW - *Next Generation Firewalls*) o de gestión unificada de amenazas (UTM - *Unified Threat Management*).

El WAF se encarga de la protección de la capa de aplicación y está diseñado para analizar el tráfico web, como elemento de acción preventiva. Por lo tanto, dentro de la capa de aplicación, analiza las comunicaciones HTTP y HTTPS.

Normalmente, cuando se habla de WAF se distinguen tres (3) tipos o aproximaciones diferentes:

- WAF físico o Appliance: dispositivo dedicado, instalado dentro de la red, que filtra las peticiones en la capa de aplicación en base a un conjunto de reglas predefinidas y a las creadas y/u optimizadas específicamente para las aplicaciones a proteger.
- WAF Software o en servidor: software específico instalado en los propios servidores web como plugins, complementos o bibliotecas adicionales a los servicios que usan los recursos de estos (por ejemplo, ModSecurity).
- WAF en la nube: permiten proteger los servicios web colocándolos por detrás de los servicios WAF de un proveedor en la nube (Akamai, Cloudflare, AWS, Azure, etc.). Se encarga de revisar todo el tráfico y aplicar las reglas de filtrado.

Uno de los aspectos fundamentales en los WAF es el conjunto de reglas que se deben aplicar para filtrar las peticiones. Existen conjuntos básicos de reglas, como OWASP ModSecurity Core Rule Set, que proporcionan protección frente a múltiples ataques web, incluidos los del OWASP TOP 10.

- SQL Injection (SQLi).
- Cross Site Scripting (XSS).
- Local File Inclusion (LFI).
- Remote File Inclusion (RFI).

- PHP Code Injection.
- Java Code Injection.
- HTTPoxy.
- Shellshock.
- Unix/Windows Shell Injection.
- Session Fixation.
- Scripting/Scanner/Bot Detection.
- Metadata/Error Leakages.

La aplicación de reglas directamente puede provocar el bloqueo de tráfico legítimo dependiendo de cómo estén desarrolladas las aplicaciones y servicios web que se estén protegiendo. **La forma de evitar estos bloqueos no consiste en reducir al mínimo el conjunto de reglas a aplicar, ya que solo se conseguiría que el WAF bloquee los ataques más evidentes, sin sacar el máximo provecho al mismo**. Para ello, los WAF suelen contar con un modo de aprendizaje, que les permite aprender durante un tiempo cuál es el tráfico legítimo para posteriormente pasarlo a modo de bloqueo donde, ya sí, se bloquearía el tráfico que se considere ilegítimo.

Se debe tener en cuenta que estos dispositivos no son infalibles y que, frente ataques como pueden ser los ataques a la lógica de negocio de la aplicación, son poco efectivos. Es por ello que se debe también tener en cuenta las recomendaciones descritas en los apartados de SEGURIDAD EN LA APLICACIÓN para conseguir un sistema seguro y robusto.

2.4 PRODUCTOS BAJO SOPORTE DE FABRICANTE

Una de las principales problemáticas existentes en la seguridad de la infraestructura es el correcto mantenimiento de los productos y tecnologías.

Con el paso del tiempo van apareciendo nuevas vulnerabilidades que pueden comprometer la infraestructura de la organización. A medida que estas vulnerabilidades van apareciendo, los diferentes fabricantes desarrollan parches de seguridad o mitigaciones para que, una vez aplicados, se impida la explotación de las mismas.

En la mayoría de las organizaciones, las tecnologías se mantienen con el paso de los años, pero, en determinado momento, los fabricantes dejan de prestar soporte para ciertos productos. Esto provoca que pasen a un nivel crítico de seguridad, ya que no se desarrollarán nuevos parches de seguridad que corrijan las futuras vulnerabilidades que se puedan descubrir.

Generalmente, sustituir estas tecnologías es una tarea compleja para las organizaciones ya que, cuanta más antigüedad tienen, mayor es la integración que tienen con el resto

de productos de la infraestructura y el proceso de migración a una versión soportada conlleva una mayor complejidad.

Disponer de un producto fuera de soporte en la infraestructura genera un aumento del riesgo en la superficie de exposición de la organización y, por tanto, debe ser una prioridad el correcto mantenimiento para evitar que los productos y tecnologías que formen parte de la infraestructura queden sin soporte del fabricante.

Por tanto, la aplicación continua de actualizaciones y parches de seguridad es uno de los aspectos vitales a tener en cuenta en los elementos de la infraestructura.

Siempre se debe tener documentado el proceso de actualización, así como, atender a las indicaciones del fabricante para cada dispositivo y versión de software o firmware. Con ello, se asegura la disposición de las últimas actualizaciones de seguridad y por tanto se reducirá el riesgo de sufrir un ciberataque, mejorando la seguridad empleada y la protección de la infraestructura de la organización[2].

2.5 CONFIGURACIÓN SEGURA DE LOS SISTEMAS

Para aplicar una configuración segura en los sistemas, se debe tener en cuenta la documentación y las recomendaciones generadas por parte del fabricante (en función de la tecnología y funcionalidades que se estén empleando), las necesidades de la organización, cuáles son los servicios más críticos o el resto de elementos que componen la infraestructura.

De manera global se deben seguir las siguientes recomendaciones de seguridad a la hora de configurar los sistemas:

- Diseñar una arquitectura de red correcta. Los modelos basados en capas o “tiers” proporcionan seguridad y asilamiento que ayudarán a reducir y mitigar los efectos de un potencial incidente de seguridad.
- Segmentar las redes lógica y/o físicamente en función de los servicios. De esta manera, sólo se permitirá el acceso a los recursos o servicios específicos de cada red y en caso de un potencial ataque, se evitará que este se propague por toda la red.
- Definir los distintos grupos, roles y usuarios necesarios para los diferentes entornos de la organización en base a sus necesidades. De igual manera, se deben revisar periódicamente y añadir, modificar o eliminar según la evolución de la organización.
- Se debe dimensionar el número de usuarios administradores en base a las capacidades de la organización. No se deben proporcionar usuarios con roles administrativos para dar respuesta a una necesidad que impacta con una

[2] https://www.ccn-cert.cni.es/informes/abstracts/5726-riesgos-y-amenazas-productos-fuera-de-soporte/file.html

restricción de la organización. Se debe analizar cada caso, y si es necesario se debe implementar un nuevo rol que permita realizar las acciones requeridas exclusivamente.

- Utilización de protocolos estándares en sus últimas versiones. Se debe priorizar el uso de protocolos que contengan capa de seguridad/cifrado y deshabilitar el uso de los protocolos que no sean seguros, no implementen cifrado o contengan vulnerabilidades, salvo en los casos que sean de uso obligatorio por las necesidades de la organización.
- Mantener un inventario actualizado de los elementos que componen la infraestructura, con el rol que desempeña cada uno y el listado de cambios realizados.
- Monitorizar de forma segura los elementos que componen la infraestructura. En el subapartado VIGILANCIA CONTINUA DEL SISTEMA se pueden encontrar más detalles al respecto.

2.6 VIGILANCIA CONTINUA DEL SISTEMA

Las herramientas SIEM son una solución unificada, cuya finalidad reside en la recopilación de eventos y alertas de seguridad de todos los elementos que componen una red, para así, analizar y correlacionar los datos recibidos, permitiendo detectar en tiempo real comportamientos anómalos, ciberataques en curso o incluso anticiparse a ataques que se vayan a producir.

Estas soluciones permiten una actuación rápida y eficaz ante cualquier amenaza o ciberataque, y por ello es recomendable utilizar estas soluciones en cualquier organización que necesite estar protegida. Más aún, en servicios como los ofrecidos mediante los servicios electrónicos, que generan mucha actividad en los sistemas que las soportan debido a la gran cantidad de usuarios que hacen uso de los mismos.

Las principales funcionalidades y beneficios de implantar un SIEM son:

- Visión centralizada y global del sistema.
- Análisis de comportamientos anómalos de usuarios y equipos.
- Módulo de inteligencia de amenazas. Correlación personalizada de los datos para la detección de ciberataques.
- Respuestas automáticas ante una alerta de amenaza.
- Supervisión de la gestión de incidentes.
- Búsqueda de actividad maliciosa mediante el análisis de *logs*.
- Cumplimiento de normativa.
- Integración con servicios en la nube mediante API.

- Apoyo de tecnología BigData para aplicar modelos de aprendizaje.
- Evaluación de vulnerabilidades.

2.7 ALTA DISPONIBILIDAD

La disponibilidad permanente del servicio se debe considerar como uno de los principios fundamentales de seguridad, ya que se debe tratar de garantizar que los usuarios tengan siempre acceso al servicio que se está prestando.

Para ello es necesario asegurar que se cuenta con la suficiente capacidad para satisfacer la demanda, contando con arquitecturas escalables, procedimientos y contramedidas frente a ataques de denegación de servicio.

Es necesario distinguir entre tolerancia a fallos, donde se habla de un entorno donde no existe la interrupción del servicio, y alta disponibilidad, donde lo que se busca es que la interrupción del servicio sea la mínima posible.

En un entorno de alta disponibilidad, es necesaria la replicación de componentes físicos, la existencia de recursos compartidos en el sistema y disponer de software que soporte esta configuración. Esta es la solución más habitual en los servicios al ciudadano, donde las aplicaciones deben poder restaurarse rápidamente, pero también pueden tolerar un breve período de interrupción (a diferencia de lo que pudiera suceder en un entorno industrial, por ejemplo).

Se indican las siguientes recomendaciones genéricas a tener en cuenta en un sistema de alta disponibilidad:

- Utilizar enlaces dedicados únicamente para proporcionar la alta disponibilidad entre los dispositivos.
- No mezclar otro tipo de flujos de datos con los protocolos de alta disponibilidad.
- Usar protocolos estándar y no protocolos propietarios de fabricantes.
- Si el protocolo lo permite, usar cifrado en el intercambio de mensajes.
- Utilizar la misma tecnología entre los elementos que conformen la alta disponibilidad.
- Utilizar configuraciones idénticas en todos los elementos que conformen la alta disponibilidad.

2.8 RESPALDO DE LAS CONFIGURACIONES

Los respaldos o copias de seguridad son un elemento fundamental que se debe tener en cuenta en un organismo para que, en caso de ciberataque o fallo del elemento, disponer de la configuración y poder subsanar de manera ágil, tanto el dispositivo afectado, como el sistema del que forma parte.

Así mismo, los respaldos nunca deben ser almacenados en el mismo dispositivo del que se realiza el backup o copia de seguridad. Estos se deben almacenar en un elemento independiente de la plataforma, aplicando bastionados de seguridad acordes a la criticidad del dispositivo.

De igual forma, **se debe aplicar una capa extra de seguridad, de manera que exista una segmentación lógica en todos los dispositivos y que se use solo para la realización de los respaldos o copias de seguridad**. De esta manera, se evitan posibles fugas de información, alteración de datos, robo de datos sensibles de la organización, vulnerabilidades y exposición de elementos a vectores de ataque.

Otro aspecto a tener en cuenta es la frecuencia con la que se deben realizar las copias de seguridad. Para definir este valor, se pueden tener en cuenta los siguientes factores:

- El volumen de datos generados o modificados.
- El coste del almacenamiento de los datos.
- Las obligaciones legales de la organización. Por ejemplo, las referentes al Reglamento Europeo de Protección de Datos (RGPD).

De igual manera y adaptándose a las necesidades de la organización se pueden llevar a cabo diferentes tipos de copia de seguridad. A continuación, se citan las de más relevancia:

- RAID1: copia de seguridad en espejo, se genera una copia exacta de los datos en tiempo real según se trabaja con ellos. Esta copia se puede realizar en local o en una ubicación externa.
- Copia de seguridad completa: consiste en realizar una copia de seguridad de todos los datos del sistema en otro soporte. Se destaca de este tipo de copia la facilidad para realizar una restauración de los datos, por el contrario, y dependiendo de la periodicidad con la que se realicen, es posible que los datos no se encuentren actualizados.
- Copia de seguridad incremental: sólo se copian los datos que hayan variado desde la última modificación, de esta manera el proceso de generar las copias de seguridad es más rápido.

Teniendo en cuenta todos estos aspectos, las organizaciones deben seleccionar los que más se adecuen a las posibilidades, necesidades y requerimientos de las mismas. **Es una buena práctica implementar la regla de copias de seguridad 3-2-1** consistente en:

- Disponer de tres (3) copias de seguridad.
- Dos (2) de las copias deben encontrarse en soportes distintos. Los soportes pueden ser discos en red, cintas, dispositivos extraíbles u otros.

- Una (1) de las copias debe estar fuera de la organización. Por ejemplo, en un proveedor de almacenamiento de nube.

De esta manera la organización se asegurará que siempre tiene disponible una copia de seguridad para restablecer el sistema en caso de incidente.

3. SEGURIDAD EN LA APLICACIÓN

Los ataques a las aplicaciones y servicios web se encuentran entre uno de los problemas fundamentales asociados a los incidentes en los últimos años.

Por ello es necesario que, de forma complementaria a todas las medidas de protección vistas hasta el momento, la aplicación que va a dar servicio implemente todas las medidas de seguridad necesarias, con el objetivo de minimizar el número de vulnerabilidades y que, si estas se dan, tengan el menor impacto posible.

3.1 CICLO DE DESARROLLO SEGURO (S-SDLC)

Es necesario desarrollar las aplicaciones siguiendo un ciclo de vida de desarrollo seguro (S-SDLC, *Secure Software Development Life Cycle*), donde la seguridad esté presente en todas sus fases. Actualmente, en los modelos *DevOps*, se agrupa el desarrollo de software y las operaciones de IT, pero es necesario moverse a modelos *DevSecOps*, donde se incluyan en este ciclo automatismos desde el punto de vista de la seguridad.

Dentro del ciclo de desarrollo seguro, deben tenerse en cuenta los siguientes elementos:

- Formar a los equipos de desarrollo en seguridad, donde es necesario que estos entiendan los ataques web, conozcan las contramedidas a aplicar y tengan siempre presente ese enfoque del atacante para poder trasladarlo al código de las aplicaciones.
- Definir los requisitos de seguridad iniciales, donde se incluyan todos los requisitos legales, los estándares de programación, las guías de desarrollo seguro a utilizar (propias o de terceros como SANS u OWASP) que sirvan de línea base durante el desarrollo.
- Introducir en los procesos de especificación, análisis y diseño los modelos de amenazas que permitan identificar y planificar las medidas para mitigar las amenazas a las que se va a encontrar sometida la aplicación, módulo o sistema.
- Someter al código (e infraestructura si se está en un entorno DevOps con contenedores, por ejemplo) a análisis estáticos de seguridad de forma automática y continuada. Cuando se habla de contenedores es necesario analizarlos antes de agregarlos al registro y si se trata de la aplicación propiamente dicha, es necesario someterla a análisis SAST (Static Application Security Testing) y análisis de dependencias para detectar riesgos de seguridad.

- Someter a la aplicación a análisis dinámicos de seguridad (DAST - Dynamic Application Security Testing) de forma periódica y automática, que permita detectar vulnerabilidades en las aplicaciones dentro del proceso de integración continua que se siga.
- Someter a la aplicación a *Test de Penetración* por parte de especialistas de seguridad (propios o de terceros) que, utilizando técnicas automáticas y manuales, detecten debilidades en las aplicaciones. Estos test deberían realizarse antes de la publicación de una nueva versión y con cierta periodicidad.

3.2 SISTEMAS DE AUTENTICACIÓN. GESTIÓN DE IDENTIDADES

A la hora de publicar un nuevo servicio electrónico, un elemento fundamental a definir y diseñar es el modelo de administración de identidad y acceso, el cual definirá los sistemas de autenticación y autorización que se implementen en la aplicación.

De forma tradicional, las aplicaciones gestionan de forma local sus identidades (mediante bases de datos de usuarios), de modo que cada aplicación gestiona su propio sistema de autorización (basado en roles y/o permisos) y su propio sistema de autenticación (tradicionalmente basado en usuario y contraseña).

Este modelo presenta muchos inconvenientes, ya que en cada aplicación debe implementarse toda la lógica, se mantienen múltiples repositorios de usuarios con las implicaciones de gestión que esto conlleva y los usuarios deben autenticarse en diferentes plataformas.

Actualmente, dentro de los procesos de trasformación digital, **lo que se busca es que las aplicaciones confíen en un proveedor de identidades (IDP - *Identity Provider*), que será el encargado de autenticar al usuario mediante alguno de los sistemas de autenticación que soporte**, y como resultado del proceso de autenticación, este contará con un *token* que intercambiará con las aplicaciones a las que desee conectarse.

Las ventajas de este método son evidentes para todos los actores, tanto para desarrolladores que no implementan los sistemas de autenticación evitando puntos de fallo, como para los usuarios, ya que solo existe un punto común donde gestionar las identidades y un único punto de inicio de sesión.

Los usuarios gestionados por el IDP pueden estar almacenados en diferentes repositorios (bases de datos, Active Directory, LDAP, Cloud, etc.) que son provisionados por este a las aplicaciones. Esta sería la aproximación que se seguiría cuando los usuarios están en la organización y los servicios que se van a publicar son para los usuarios de esta. Este modelo se podría extender entre organizaciones haciendo uso de servicios de federación de identidades.

Cuando desde la administración se publican servicios electrónicos, la aproximación es ligeramente diferente. En este caso, se cuenta con un identificador único (Certificado

digital del ciudadano o DNIe), el cual se autentica a través del IDP (@clave), que además actúa como sistema de federación de identidades. De este modo, cuando el usuario se autentica, se genera un *token SAML* firmado digitalmente por el certificado del IDP con la identidad del usuario validado (NIF, Nombre y Apellidos), el cual es enviado por el navegador del usuario al servicio que deberá verificar la firma del *token* para confiar en la información que viene en el mismo.

En esta integración con el gestor de Identidades, y específicamente con @Clave, es necesario validar el *token XML* del protocolo *SAML*, ya que este *token* no se encuentra cifrado y, dado que es enviado por el navegador del usuario, puede ser manipulado. **Si no se verifica el *token* sería posible suplantar a cualquier usuario en la aplicación.**

Si el servicio publicado está utilizando @Clave u otro IDP, la verificación del *token* (independientemente del protocolo de federación de identidades) para evitar ataques de suplantación es el punto más crítico. Ya que el sistema de autenticación sería proporcionado por el IDP, la validación debe incluir la firma digital del mensaje, si ha sido firmado por el IDP adecuado, la validez temporal, etc.

En caso de que sea necesario publicar un servicio con una aproximación más tradicional (usuarios y sistemas de autenticación propios), sería recomendable tener en cuenta los siguientes puntos:

- Utilizar sistemas de autenticación fuertes, basados en certificados digitales de cliente o usuario y contraseña con doble factor de autenticación (OTP, SMS, etc.).
- Realizar toda la comunicación a través de protocolos seguros y suites de cifrados robustas.
- Implementar sistemas de bloqueo de cuentas temporales tras varios intentos de inicio de sesión erróneos, junto a sistemas que minimicen la eficacia de los ataques por fuerza bruta (políticas de contraseñas robustas, *captchas* tras varios intentos, MFA, etc.)
- Si se utilizan contraseñas, estas deben almacenarse utilizando algoritmos criptográficos seguros, preferiblemente con funciones específicas para el almacenamiento de contraseñas o funciones de derivación de claves (como *scrypt* o PBKDF2, que usan saltos y múltiples iteraciones), frente al uso simple de funciones HASH de propósito general (SHA1, SHA2, etc.).
- Si se implementan sistemas para recuperar las contraseñas de un usuario, se debe enviar al correo asociado un enlace de un solo uso con tiempo de expiración, no enviando nunca contraseñas por correo electrónico.
- Evitar proporcionar excesiva información en los mensajes de error, no informando de la existencia o no del usuario en los intentos de inicio de sesión, o de la existencia de los correos electrónicos en los intentos de recuperación de contraseñas.

- Registrar y monitorizar todos los intentos de inicio de sesión fallidos, tanto por usuario como por contraseña, al igual que todos los bloqueos de cuentas.
- Solicitar re-autenticación para aquellas acciones que se consideren críticas o manejen información sensible, donde se requiera una firma digital (si se utilizan certificados), validar alguna acción con el sistema 2FA o volver a enviar credenciales en caso de ser el único mecanismo disponible.

Por último, si el servicio que se ofrece al ciudadano no cuenta con un sistema de autenticación propiamente dicho, sino que se va a identificar a este en base a determinados datos que solo este debería conocer, se recomienda tener en cuenta además estos aspectos:

- Solicitar varios datos de modo que no todos se encuentren en el mismo documento del ciudadano. Por ejemplo, solicitar un DNI y Fecha de Nacimiento (presentes en el DNI) y un teléfono (no presente en el DNI) o el número de la seguridad social (presente en la tarjeta sanitaria), etc.
- Reducir el servicio a lo estrictamente necesario, evitando que este sistema proporcione acceso a otra información del ciudadano, servicios o documentación.

3.3 DEFINICIÓN DE ROLES Y PRIVILEGIOS

Una vez autenticado el usuario, otro de los puntos fundamentales a considerar son los procesos de autorización, de modo que todas las acciones que se realizan son verificadas para determinar que se tiene derecho a realizarlas.

En este aspecto, se recomienda tener en consideración los siguientes elementos:

- Utilizar en todo momento el principio de menor privilegio, comenzando por las fases de diseño, donde se establecen los diferentes tipos de usuarios, las acciones que pueden realizar y sobre qué recursos.

  Esta información debe estar disponible al inicio del proyecto o módulo (bien por *rol* o por atributos) para poder implementarla correctamente y realizar las pruebas (manuales y automáticas) oportunas durante el desarrollo.
- Utilizar siempre la aproximación de “Denegado por defecto” de modo que, si al tratar de realizar una acción no se cuenta con el permiso específico, se deniegue en vez de asumir la idea de “Acceso por defecto”.
- Validar el acceso en cada petición, independientemente del origen y propósito de esta. Se debe validar que el usuario tiene permiso para la acción que va a realizar y para el recurso sobre el que va a actuar. Todas estas validaciones deben realizarse en la parte de la aplicación que se ejecuta en el servidor.
- Para el control de acceso se utilizan dos (2) aproximaciones: RBAC (control de acceso basado en roles) y ABAC (Control de acceso basado en atributos).

La decisión de seguir una u otra debe tomarse nuevamente al inicio del proyecto. Actualmente la tendencia son los modelos ABAC, sobre todo en aquellas aplicaciones más complejas donde se requiere mayor flexibilidad, integraciones entre organizaciones o donde el número de roles sería excesivo y más si no existe jerarquía entre ellos. Si bien, en otras ocasiones, para aplicaciones más pequeñas y acotadas, el modelo RBAC puede ser perfectamente válido.

- Evitar la exposición de identificadores internos de recursos (tradicionalmente son identificadores numéricos secuenciales), que puedan incitar a la manipulación por parte de un usuario malicioso, ya que un fallo en el control de acceso daría la posibilidad al atacante de acceder a dichos elementos.
- Incorporar los recursos estáticos al control de acceso siempre que sea posible, de modo que tanto las imágenes, documentos y otros ficheros estáticos estén sometidos a este control.

Todas estas comprobaciones de acceso se realizan en base a la sesión del usuario que se autentica, la cual es identificada a través de un *token* o ID de sesión, de modo que, proteger este identificador es fundamental para evitar suplantar a cualquier usuario. Para proteger este identificador se deben tener en cuenta los siguientes aspectos:

- El identificador de la sesión no debe mantener información sobre el usuario o funcionamiento de la aplicación, debe ser únicamente un valor que se intercambia entre cliente y servidor.
- Este identificador no debe ser predecible, debe ser generado de forma aleatoria y con una entropía suficiente para evitar que un ataque por fuerza bruta localice una sesión válida.
- La sesión debe invalidarse cuando el usuario cierra sesión o cuando pasa un tiempo determinado sin actividad.
- La política de caché de la aplicación debe evitar que el identificador de sesión se almacene en el equipo o *proxies* intermedios, impidiendo también que este identificador se trasmita en peticiones GET.
- La aplicación no debe permitir la fijación de sesión, de modo que, al iniciar la sesión se invaliden los identificadores existentes y se generen unos nuevos.
- Cuando el identificador de sesión se trasmite a través de una *cookie* (lo habitual en una aplicación web), esta debe estar marcada con los atributos adecuados: *HttpOnly*, *Secure*, *SameSite*, *domain*, *path* y expiración.
- Verificar que la conexión entre navegador y cliente se realiza a través de HTTPS, con una configuración robusta de certificado, protocolos y suites de cifrado, que se utiliza la cabecera de Respuesta HSTS (*HTTP Strict Transport Security*) para evitar ataques de tipo *SSL Strippping* (proxy para pasar de HTTPS a HTTP) y, si se

quiere dar soporte a aplicaciones antiguas, que el soporte HTTP redirige únicamente a la versión HTTPS.

3.4 AUDITORÍA DE ACCIONES

Desde el punto de vista de la seguridad, **es necesario registrar y monitorizar las acciones que se realizan en la aplicación, con múltiples objetivos, que van desde evitar el no-repudio a identificar incidentes de seguridad o defenderse de ataques**.

De forma adicional al registro que se realiza desde otras fuentes de datos, como son los Firewall, WAF, Bases de datos, IDS o los sistemas operativos, las aplicaciones deben registrar las acciones que sobre ellas se realizan.

Independientemente del sistema de registro que se implemente, se deberían tener en cuenta los siguientes elementos:

- Los ficheros de registros pueden almacenarse en disco o en base de datos (SQL o NoSQL). En cualquier caso, es recomendable que sean particiones de disco o bases de datos independientes, y por supuesto, en el caso de ficheros, no accesibles desde la parte externa de la aplicación.
- Utilizar protocolos seguros para el envío y almacenamiento de la información, así como formatos estándar que faciliten la integración con otras fuentes.
- Integrar estos registros en un sistema centralizado, como un SIEM, que permita una mejor monitorización y seguimiento, así como la generación de alertas que permitan ser preventivos frente a ataques o anomalías.
- Registrar todos los eventos que puedan suponer un riesgo de seguridad (evaluados durante el diseño de la aplicación), tales como fallos en la validación de entrada, intentos fallidos de inicio de sesión, accesos denegados a acciones, errores de aplicación, acciones críticas como la modificación de usuarios o pagos, acciones con implicaciones legales, etc.
- El registro debe incluir, al menos, el momento de suceso, desde dónde sucede (dirección IP, equipo, aplicación web, módulo, etc.), el usuario que provoca la acción y la acción en sí misma con la descripción. El registro debe evitar toda información sensible, tales como credenciales, contraseñas, *tokens*, datos personales, etc.
- Los registros deben estar protegidos frente a manipulación, tanto en tránsito como en reposo, asegurando la integridad de estos, manteniendo sobre ellos monitorización en el acceso y aplicando los mismos principios de seguridad como el de menor privilegio, ya que sobre ellos solo sería necesario un acceso de lectura.

4. REFERENCIAS

- Real Decreto 203/2021, de 30 de marzo, por el que se aprueba el Reglamento de actuación y funcionamiento del sector público por medios electrónicos: https://www.boe.es/buscar/act.php?id=BOE-A-2021-5032
- Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo, de 27 de abril de 2016, relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que se deroga la Directiva 95/46/CE (Reglamento general de protección de datos): https://www.boe.es/buscar/doc.php?id=DOUE-L-2016-80807
- Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos Personales y Garantía de los Derechos Digitales: https://www.boe.es/eli/es/lo/2018/12/05/3
- Real Decreto 1494/2007, de 12 de noviembre, por el que se aprueba el Reglamento sobre las condiciones básicas para el acceso de las personas con discapacidad a las tecnologías, productos y servicios relacionados con la sociedad de la información y medios de comunicación social: https://boe.es/diario_boe/txt.php?id=BOE-A-2007-19968
- Real Decreto 311/2022, de 3 de mayo, por el que se regula el Esquema Nacional de Seguridad: https://www.boe.es/buscar/doc.php?id=BOE-A-2022-7191
- Guías CCN-STIC de las Series 400: https://www.ccn-cert.cni.es/en/guides/complete-series/400-guias-generales.html
- Guías CCN-STIC de las Series 600: https://www.ccn-cert.cni.es/en/guides/complete-series/600-guias-de-otros-entornos.html
- Guías CCN-STIC de las Series 800: https://www.ccn-cert.cni.es/en/guides/complete-series/series-800-ens-html.html
- Riesgos y amenazas en productos fuera de soporte: prevención y protección: https://www.ccn-cert.cni.es/informes/abstracts/5726-riesgos-y-amenazas-productos-fuera-de-soporte/file.html
- Recomendaciones para acceso remoto seguro a información sensible preservando seguridad y productividad: https://www.ccn-cert.cni.es/informes/abstracts/6230-recomendaciones-para-acceso-remoto-seguro-a-informacio-n-sensible-preservando-seguridad-y-productividad/file.html
- Prevención proactiva: proceso de auditoría para evitar situaciones de riesgo: https://www.ccn-cert.cni.es/informes/abstracts/5873-prevencion-proactiva-proceso-de-auditoria-para-evitar-situaciones-de-riesgo/file.html

- Ataques DDoS. Recomendaciones y buenas prácticas: https://www.ccn-cert.cni.es/informes/abstracts/4925-ataques-ddos-recomendaciones-y-buenas-practicas/file.html
- Guía de Seguridad de las TIC CCN-STIC 140: https://www.ccn-cert.cni.es/pdf/guias/series-ccn-stic/guias-de-acceso-publico-ccn-stic/2518-ccn-stic-140-taxonomia-de-referencia-para-productos-de-seguridad-tic/file.html
- CCN-CERT BP/07 Recomendaciones implementación HTTPS: https://www.ccn-cert.cni.es/informes/informes-de-buenas-practicas-bp/2109-ccn-cert-bp-07-recomendaciones-implementacion-https-1/file.html
- OWASP Cheat Sheet Series: https://cheatsheetseries.owasp.org/index.html

CENTRO CRIPTOLOGICO NACIONAL
cn=CENTRO CRIPTOLOGICO NACIONAL,
2.5.4.97=VATES-S2800155J, ou=CENTRO
CRIPTOLOGICO NACIONAL, o=CENTRO
CRIPTOLOGICO NACIONAL, c=ES
2022.06.19 22:46:58 +02'00'