Gobierno de Chile

LPC/NOC/JSE/JPC/CAD/BPE

DIVISIÓN JURÍDICA

MINISTERIO DE EDUCACION
07 ENE 2022
DOCUMENTO TOTALMENTE TRAMITADO

APRUEBA GUÍA DE CONTROLES DE SEGURIDAD PARA EL DESARROLLO DE SISTEMAS E INFRAESTRUCTURA RELACIONADA.

Solicitud N° **6396**

SANTIAGO,

RESOLUCIÓN EXENTA N°

000282 07.01.2022

VISTO:

Lo dispuesto en la Ley N° 18.956, que Reestructura el Ministerio de Educación; en la Ley N° 19.880, que Establece Bases de los Procedimientos Administrativos que Rigen los Actos de los Órganos de la Administración del Estado; en el D.F.L N° 29 de 2004, del Ministerio de Hacienda, que fija el texto Refundido, Coordinado y Sistematizado de la Ley N° 18.834, sobre Estatuto Administrativo; en el D.F.L N° 1-19.653, de 2000 del Ministerio Secretaría General de la Presidencia que Fija Texto Refundido, Coordinado y Sistematizado de la Ley N° 18.575 Orgánica Constitucional de Bases Generales de la Administración; en el Decreto N° 83, de 2004, del Ministerio Secretaría General de la Presidencia, que Aprueba Norma Técnica para los Órganos de la Administración del Estado sobre Seguridad y Confidencialidad de los Documentos Electrónicos; en el Decreto Supremo N° 93, de 2006, del Ministerio Secretaría General de la Presidencia, que Aprueba Norma Técnica para la Adopción de Medidas Destinadas a Minimizar los Efectos Perjudiciales de los Mensajes Electrónicos Masivos no Solicitados Recibidos en las Casillas Electrónicas de los Órganos de la Administración y sus Funcionarios; en la Resolución Exenta N° 296 de 19 de enero de 2021 que Aprueba la Nueva Política de Seguridad de la Información del Ministerio de Educación y Define las Labores del Órgano Encargado de Velar por su Cumplimiento; en el Memorándum N° 39 del Encargado de Seguridad de Información; y en la Resolución N° 7, de 2019, de Contraloría General de la República.

CONSIDERANDO:

Que, en conformidad a lo dispuesto en la Ley N°18.956 de 1990, que Reestructura el Ministerio de Educación, esta Cartera es la encargada, entre otras funciones, de fomentar el desarrollo de la educación en todos sus niveles.

Para apoyar su cumplimiento, el Ministerio de Educación, ha desarrollado una plataforma tecnológica a través de la cual se registra, procesa, transmite y produce el almacenaje de datos, mediante diferentes dispositivos, que permiten interactuar tanto con los funcionarios en todo el país, como con la comunidad escolar y la ciudadanía en general.

En este orden de ideas, mediante la Resolución Exenta N° 4.503 de 11 de julio de 2014, de esta Subsecretaría de Educación, estableció la función de Encargado de Seguridad de la Información y, aprobó luego, mediante Resolución Exenta N° 296 del 19 de enero de 2021, "la Nueva Política de Seguridad de la Información" definiendo las labores del órgano encargado de velar por su cumplimiento.

Que, con el objetivo de definir los requerimientos y controles de seguridad para asegurar la confidencialidad, en el ciclo de la vida de los sistemas de información del Ministerio de Educación e infraestructura relacionada, focalizando en la adquisición, desarrollo y mantención de los sistemas gestionados por la Coordinación Nacional de Tecnología, se hace necesario establecer una herramienta o guía básica para el desarrollo de dichos sistemas, tanto para proveedores externos como funcionarios de la Coordinación Nacional de Tecnología.

Que, en dicho sentido, este procedimiento es elaborado por el Encargado de Seguridad de la Información, según lo señalado en el Memorándum N° 39, de 2021.

Que, conforme lo anterior, atendida a la normativa vigente y sus disposiciones en esta materia, se hace necesario sancionarlo mediante el presente acto administrativo.

RESUELVO:

1.Apruébase la "Guía de Controles de Seguridad para Desarrollo de Sistemas e Infraestructura Relacionada" del Ministerio de Educación, cuyo texto y sus anexos, se adjuntan al presente acto y se entiende forman parte integrante del mismo.

ANÓTESE, COMUNÍQUESE Y ARCHÍVESE

JORGE POBLETE AEDO
SUBSECRETARIO DE EDUCACIÓN

Distribución:
- Of. Partes
- Gabinete Sr. Ministro
- Gabinete Sr. Subsecretario
- División Jurídica
- División de Planificación y Presupuesto.
- Exp. N° 39.060

REPÚBLICA DE CHILE
MINISTERIO DE EDUCACIÓN

GUIA DE CONTROLES DE SEGURIDAD PARA DESARROLLO DE SISTEMAS E INFRAESTRUCTURA RELACIONADA

MINISTERIO DE EDUCACIÓN

2021

INDICE

FIRMA DE LOS RESPONSABLES

ELABORADO POR	REVISADO POR	APROBADO
Rodrigo Parada	Juan Antonio Serrano	León Paul Castro

CONTROL DE VERSIONES

REVISIONES DEL DOCUMENTO DE POLITICA				
Nº Revisi	Fecha Aprobación	Motivo de la revisión	Páginas Modificadas	Autor
1		Elaboración inicial	Todas	Rodrigo Parada
2	23-01-2020	Agregar controles faltantes	4.1. Validación de entradas 4.2. Autenticación 4.3. Manejo de sesiones 4.4. Gestión de Cookies 4.5. Manejo de errores 4.6. Logs Auditoria 4.7. Cifrado de Datos	Ingrid Hermosilla
3	23-01-2020	Nuevas sesiones de Control	4.9. Configuración de los sistemas 4.10 . Seguridad de Base de Datos 5. SEGURIDAD EN CLOUD 6. CONTROLES DE SEGURIDAD EN CMS 7. CONTROLES DE SEGURIDAD EN APIs	Ingrid Hermosilla
4	06-10-2021	Se agregan referencias normativas y se ajusta numeración.	Todas	Paola Fuentes

1. OBJETIVO

Definir los requerimientos y controles de seguridad para asegurar la confidencialidad, integridad y disponibilidad, en el ciclo de vida de los sistemas de información del Ministerio de Educación e infraestructura relacionada, focalizado en la adquisición, desarrollo y mantenimiento de los sistemas gestionados por la Coordinación Nacional de Tecnología.

Establecer esta guía como una herramienta básica para el desarrollo de sistemas, tanto para proveedores externos como funcionarios de la Coordinación Nacional de Tecnología.

2. ALCANCE

Este documento está dirigido a todos los funcionarios tanto internos como externos y a las diferentes áreas que intervienen en las distintas etapas del ciclo de desarrollo en los sistemas informáticos en el Ministerio de Educación.

3. REFERENCIAS

Política de seguridad en la adquisición, desarrollo y mantención de sistemas, Dex 1125 del 18/12/2018.
Procedimiento de desarrollo seguro y segregación de ambientes, Of(O) 1809 del 02/09/21.
Procedimiento Estándares QA, Rex 1796 del 05/04/21.
Guía de Stándares, Rex 1797 del 05/04/21.
Norma técnica NCh-ISO 27001-2013.

4. CONSIDERACIONES DE SEGURIDAD PARA EL DESARROLLO DE SISTEMAS

4.1. Consideraciones para empresas de desarrollo externo

a) Los requerimientos de seguridad son los controles que deben cumplir las empresas proveedoras de desarrollo externo contratadas por el Ministerio de Educación, las cuales deben estar especificadas en el listado de requerimientos definidos.

b) Cualquier empleado de una empresa externa que preste servicios al Ministerio de Educación, ya sea en forma remota o local, que detecte, identifique, explote consciente o inconscientemente una vulnerabilidad o bien detecte un incidente de seguridad, debe informar inmediatamente al jefe de proyecto, jefe de área que corresponda y al Encargado del área de Seguridad de la Información de esta Cartera de Estado oportunamente, a través de los medios de comunicación disponible.

c) Los equipos informáticos de empresas externas que se conecten a la Red del Ministerio de Educación deben tener implementado una solución de antivirus actualizada.

d) Toda herramienta de software usado dentro de la Red Mineduc por parte del personal de la empresa externa deberá estar previamente autorizado por el área de Seguridad de la Información de este Ministerio.

e) La empresa externa debe adjuntar los procedimientos para gestionar incidentes que se generen en sus sistemas o servicios tecnológicos, que pudieran afectar algún recurso tecnológico de la Ministerio de Educación.

f) Toda empresa externa que preste servicio de desarrollo externo debe disponer de acuerdos de nivel de servicio (SLA por sus siglas en inglés), mutuamente acordados y especificados en el contrato de servicios.

g) Respecto al código de programación desarrollado, debe estar acorde a los requerimientos funcionales del proyecto, además no debe contener ningún código malicioso o que permita funciones no especificadas que permitan vulnerar el sistema.

h) La(s) empresa(s) que presten servicios al Ministerio de Educación deben firmar un acuerdo de privacidad o confidencialidad (NDA), que restringa la información tratada perteneciente a este Ministerio, asegurando la confidencialidad de esta.

i) La empresa que preste servicios al Ministerio de Educación debe firmar un acuerdo contractual que incluya el conocimiento relativo a propiedad intelectual, el cual debe asegurar en todo momento que el Ministerio de Educación es el exclusivo dueño del recurso desarrollado.

j) La empresa que preste servicios al Ministerio de Educación y que deba conectarse a la Red Mineduc desde el exterior, deben utilizar una conexión segura VPN, previa autorización del acceso por parte del área de Seguridad de la Información

k) Se prohíbe el acceso por parte de empresas externas a cualquier recurso informático dentro de la Red Mineduc, que no se encuentre autorizado por parte del área de seguridad de la información.

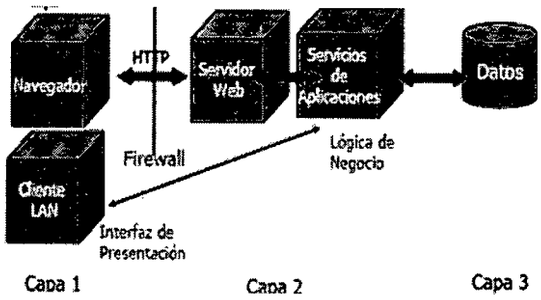
l) Está prohibido para las empresas externas intentar o bien explotar cualquier vulnerabilidad o debilidad de los sistemas de la red Mineduc.

m) El personal externo que preste servicios al Ministerio de Educación, deberá firmar una cláusula de confidencialidad para persona natural (NDA), lo cual será controlado por el encargado del área de seguridad de la información.

4.2. Arquitectura de seguridad

a) Las tecnologías usadas en proyectos desarrollados por la CNT y los desarrollos externos deberán cumplir los lineamentos y buenas prácticas de desarrollo, la que fijará los requerimientos técnicos para asegurar rendimiento, escalabilidad, disponibilidad, auditoría y seguridad del sistema a desarrollar.

b) Se debe preferir modelos de desarrollo físico y lógico en tres capas MVC (Modelo Vista Controlador), las cuales entregan niveles mínimos de seguridad en el producto final.



c) Los Sistemas informáticos deben adaptarse al control de accesos por medio del anillo de seguridad conformado por los diferentes dispositivos de seguridad perimetral implementados en el Ministerio.

5. CONTROLES DE SEGURIDAD EN EL DESARROLLO

En términos generales, el costo de construir software seguro es mucho menor que reparar problemas de seguridad cuando un sistema ya se encuentra en producción.

Para evitar lo mencionado anteriormente, a continuación, se presentan una lista de controles de seguridad, para así mitigar las vulnerabilidades más comunes.

5.1. Validación de entradas

a) Validar todos los datos del cliente antes de procesarlos, incluyendo todos los parámetros, URLs y contenidos de cabeceras HTTP, input fields, hidden fields, drop down list, etc. Caso especial es la revisión de los ficheros que pueda proveer el usuario en forma directa o a través de otro sistema.

b) Validar los datos en todas las capas.

c) Controlar tipos y largos de datos no esperados especialmente caracteres de tipo (< > " ' % ( ) & + \ / \ %null unicode hexadecimal, etc/password, +unión, 1=1, 1=2).

d) Todas las fallas en la validación deber terminar en el rechazo del dato en el sistema.

e) Controlar los recursos descargables evitando cualquier modificación de parámetros para evitar ataques de tipo CSRF.

f) Controlar sobre la capa de interfaz violaciones de seguridad que expongan datos sensibles del sistema.

g) Limitar los caracteres de entrada que un usuario puede introducir en los formularios.

h) Añadir al protocolo HTTP una cabecera llamada X-Frame-Options. La que permite especificar si el navegador debe permitir o no que un navegador muestre una página en una etiqueta <frame>, <iframe> u <object>.

i) Utilizar sesiones validas cuando se utilicen recursos del sistema.

j) Controlar contextualmente todas las salidas de datos no confiables hacia consultas SQL, JSON y XML.

k) Evitar generar código con valores ingresados por el usuario.

l) Asegurar que la aplicación funciona correctamente cuando se presentan grandes volúmenes de peticiones, transacciones y/o tráfico de red.

m) Asegurar que la aplicación no permite a un atacante reiniciar o bloquear cuentas de usuario.

n) Asegurar que la aplicación refuerza su modelo de control de acceso asegurando que ningún parámetro disponible a un atacante proporcionará recursos o datos adicionales.

o) Asegurar que no hay datos sensibles en el HTML que podrían conducir a un atacante a montar un ataque enfocado.

p) Controlar los campos de formulario en especial los de tipo "hidden", los cuales pueden contener parámetros modificables que permitan accionar acciones no deseables en el sistema.

5.2. Autenticación

a) Toda aplicación accesible desde Internet, que requiera autenticación de usuarios, deberá contar con un sistema Recaptcha en cada ventana de Login y en el proceso de recuperación de contraseña, a fin de mitigar los posibles ataques de autenticación por fuerza bruta, para el caso de las ventanas de Login se recomienda después del tercer o quinto intento de Login erróneo.

b) Para los mecanismos de autentificación de las credenciales, se deben utilizar algoritmos de integridad seguras, recomendando técnicas como SALT.

c) Además, en el caso de las aplicaciones Internas, estas deben autentificarse bajo el LDAP interno (MS Active Directory).

d) Requerir autenticación y sesión válida para todos los recursos y páginas excepto aquellas específicamente clasificadas como públicas e informativas.

e) Para los desarrollos Internos y externos utilizar los mecanismos de autentificaciones estandarizados por este Ministerio denominado (Arquetipo).

f) La transmisión de las credenciales de autenticación o cualquier información sensible debe ser vía método HTTP POST y nunca con GET.

g) Todas las páginas (URLs) deben cumplir con los requisitos de autentificación.

h) Establecer y utilizar servicios de autenticación usados por este Ministerio (LDAP).

i) Se debe asegurar utilizar contraseñas robustas, como lo establece Política de control de acceso lógico y el Procedimiento de Inicio y Cerrado de Sesión Seguro de PC y Uso de contraseña.

j) El proceso de cambio y reseteo de contraseñas requieren los mismos niveles de control implementados en la autenticación de cuentas.

k) Para los procesos de recuperación de contraseñas nunca usar Link de restauración, con valores adivinables, se recomienda que el sistema entregue una contraseña temporal.

l) Deshabilitar la funcionalidad de "recordar" campos de contraseñas, para evitar que las credenciales queden guardadas en el navegador del usuario.

5.3. Manejo de sesiones

a) Validar los ID y datos de sesión.

b) Los ID de sesión deben ser complejos y robustos, para mitigar el riesgo de ataques de fuerza bruta.

c) Las sesiones de usuario deben ser controladas y no pueden ser reutilizadas por otro usuario o sesión, se recomienda el uso de eventos como cierre de ventanas, el uso del botón para el término de sesión o el bloqueo de usuario después de cierto número de intentos de acceso fallidos.

d) Se debe tener control del tiempo de expiración o Time Out, que engloba eventos no controlados por el usuario, como es el caso de caídas de la aplicaciones o intercepciones maliciosas que provoquen una expiración. Por lo anterior, es de suma importancia definir el tiempo de expiración ´de la sesión según la naturaleza del sistema, pero en ningún caso se debe dejar sin especificar una métrica para este valor.

e) Controlar y restringir las sesiones múltiples de un usuario en el mismo sistema, parametrizando la gestión de sesiones multithreaded/multi-user. En el caso de las conexiones por usuario, deben estar restringidas a sólo una conexión activa por usuario.

f) Mostrar mensajes genéricos de error en la validación de credenciales o accesos fallidos, para mitigar los riesgos de enumeración de cuentas de usuario.

g) Asegúrese que una vez que un usuario válido ha iniciado sesión, no sea posible cambiar el parámetro con el ID de sesión para reflejar otra cuenta de usuario.

h) Verifique si es posible acceder páginas o funciones del sistema sin una sesión valida.

5.4. Gestión de Cookies

a) Las cookies generadas por una aplicación Web, ya sea para almacenar sesiones, preferencias, o cualquier otro dato; deben quedar inalcanzables para aplicaciones en otros dominios, a menos de que efectivamente estas deban compartir información.

b) Las cookies del usuario no deben contener datos que puedan ser manipulados en otra sesión y no deben contener información sensible para el sistema.

c) Almacenar sólo el identificador de la sesión y no almacenar información sensible.

d) Asegurar de no mantener cookies que permitan realizar operaciones no autorizadas por medio de la manipulación de estas.

e) Usar cifrado y protocolos seguros HTTPS. Además se recomienda usar HTTP Strict Transport Security (HSTS).

f) Asegurarse de cifrar una cookie completa, si contiene información sensible.

g) Si el desarrollador utiliza cookies para almacenar datos, entonces debe revisar y utilizar algoritmos públicos fuertes (como AES-SHA2).

h) Todas las transiciones de estados en el código de la aplicación deben poseer uso seguro de cookies.

i) Las cookies deben contener la mínima información privada posible.

j) Definir todas las cookies que usa la aplicación, sus nombres y para qué son necesarias.

5.5. Manejo de errores

a) Se debe manejar todos los errores y excepciones del sistema, para asegurar que el sistema nunca entregue información del mismo sistema, como por ejemplo librerías, códigos de programación, estructuras de base de datos, cualquier recurso de desarrollo, rutas internas, IP's internas, Registros de Logs, etc.

b) Todas las llamadas a métodos / funciones que devuelven un valor tienen su control de errores y además debe comprobar el valor devuelto.

c) Gestionar adecuadamente las excepciones y los errores.

5.6. Logs Auditoria

a) Los registros de Log del sistema nunca deben ser alcanzables desde la internet y deben ser guardados en rutas internas protegidas de posibles atacantes.

b) Para todos los sistemas que entreguen beneficios deben incorporar registro de auditoría, según lo establece la "Política de registro de Log" del Ministerio de Educación.

c) Asegurar que la aplicación falla de un modo seguro.

d) No registrar información sensible en el log en caso de error.

e) Definir y controlar la longitud máxima de una entrada de log.

f) Asegurar que no registramos datos sensibles en el log: cookies, método HTTP "GET", credenciales de autenticación.

g) Determinar si al hacer debug estamos registrando en el log datos sensibles.

h) Determinar si la aplicación auditará las operaciones lanzadas desde el cliente, sobre todo la manipulación de datos: Create, Update, Delete (operaciones CUD).

i) Registrar en el log las operaciones de autenticación (fallidas o exitosas).

j) Registrar en el log los errores de la aplicación.

5.7. Cifrado de Datos

a) Se recomienda el uso de certificados SSL, para todo lo que se encuentre expuesto a internet.

b) Para los sistemas que incorporen mecanismos de autentificación se debe utilizar HTTPS como protocolo de comunicaciones del sistema, según el Decreto Supremo N° 1 articulo 15 ("Norma técnica sobre sistemas y sitios web de los órganos de la administración del Estado"), esta resolución evitará transmitir datos sensibles en claro.

c) Para los sistemas que utilicen HTTPS, nunca habilitar el protocolo HTTP simultáneamente a excepción de ser utilizado como redirección al canal seguro.

d) Se debe asegurar que el cifrado TLS no contenga vulnerabilidades que afecten la confidencialidad y la disponibilidad del sistema. Sólo dejar habilitados protocolos seguros (TLS 1.2 y 1.3 en adelante).

e) Se debe tener el resguardo correspondiente a la llave privada, respaldando estos archivos en sistemas seguros.

f) Se debe tener un control sobre la validez de los certificados digitales utilizados.

g) Se debe usar entidades certificadoras KPI reconocidas y fiables.

h) Para cifrar los datos es recomendable utilizar algoritmos públicos fuertes. AES 256 o superior y Sha256 o superior de 2048 bits o superior.

i) Asegurar que la aplicación implementa métodos criptográficos efectivos y actualizados.

j) Cifrar credenciales usando cifrado no reversible, como algoritmos de resumen (Hash), y una semilla para evitar ataques de diccionario.

5.8. Entorno de Código Seguro

a) Verificar que en la estructura de ficheros no exista algún componente que esté directamente accesible para los usuarios, lo cual debe ser evitado.

b) Comprobar la gestión de memoria (reservar/liberar).

c) Eliminar el código comentado (aunque sea para pruebas,) que pueda contener información sensible.

d) Asegurar que todas las bifurcaciones de código tengan su cláusula default (if-else, switch- default, etc).

e) Asegurar que no hay "kits de entorno de Desarrollo" en los directorios en explotación.

f) Reemplazar sentencias SQL dinámicas por StoredProcedures

5.9. Configuración de los sistemas

a) Se debe asegurar que los servidores, los Frameworks y los componentes del sistema están con la última versión aprobada por este Ministerio, con el propósito de mitigar las vulnerabilidades.

b) Asegurar que el servidor no soporta la métodos y servicios riesgosos en Internet (ejemplo; PUT, DELETE, WEBDAV).

c) Se debe deshabilitar el listado de directorios en todo el sistema en especial para aquellos sistemas que se encuentren expuestos a la Internet.

d) Se debe eliminar todos los códigos de ejemplos y de prueba en ambientes productivos.

e) Se debe eliminar todos los registros de control de versiones en ambientes productivos.

f) Los permisos de archivos y carpetas deben tener los mínimos privilegios que permitan el funcionamiento del sistema.

g) Se prohíbe cualquier código de programación riesgosa que pudiera afectar la confidencialidad, integridad y disponibilidad del sistema.

h) Se debe restringir el acceso desde la Internet a los portales de administración del sistema.

i) Se debe restringir cualquier puerto o servicio con el objetivo de reducir la superficie de ataque, salvo que sea estrictamente necesario para el sistema.

j) No se debe permitir que las aplicaciones ejecuten comandos directamente en el sistema operativo, menos mediante la invocación de una Shell.

k) Se debe deshabilitar o eliminar los usuarios de administración por defecto en los sistemas (ejemplo; superadmin, admin, root, Sysadmin, administrador, etc.)

l) Asegurar que ningún archivo de respaldo se encuentre en ambientes productivos y accesible de la aplicación.

m) Asegurar que el componente del servidor Web no tenga ninguna vulnerabilidad de seguridad.

n) Asegurar que los servicios WebServices utilicen mecanismos de autenticación y las respuestas sean controladas.

o) Evitar y/o eliminar la existencia de directorios accesibles con información sensible en la aplicación.

5.10. Seguridad de Base de Datos

a) La aplicación debe utilizar el mínimo nivel de privilegios cuando accede a la base de datos.

b) La administración de las bases de datos, solo pueden ser accedidas por personal autorizado y con el uso de protocolos seguros.

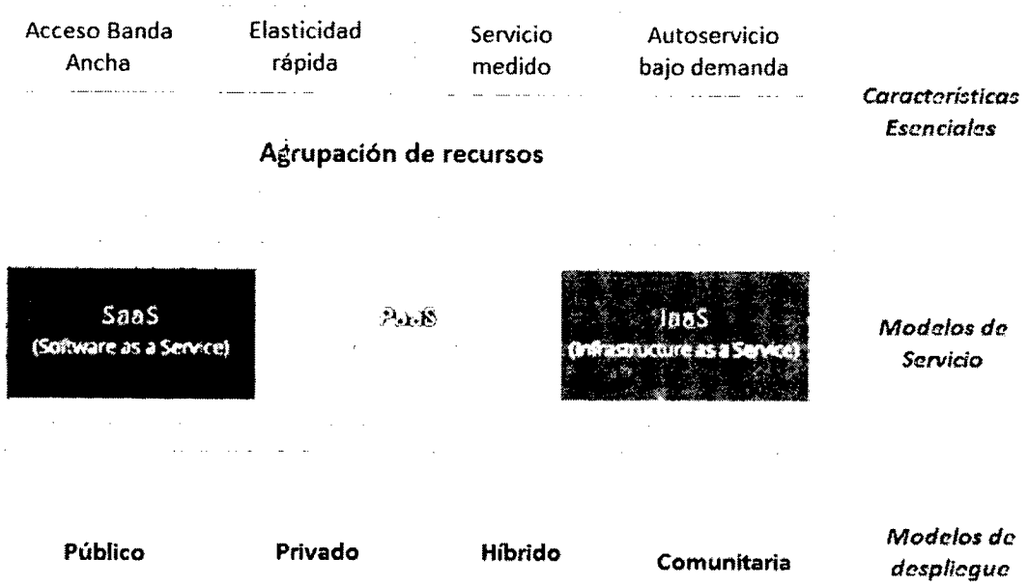
c) Las bases de datos deben ser protegidas por medio de firewall.

d) Para los administradores de Base de Datos se prohíbe la instalación y uso de software descargados de sitios no confiables, solo se permitirán las aplicaciones previamente autorizados por el área de seguridad de la información.

e) Deshabilite todas las funcionalidades innecesarias de la base de datos (por ejemplo: procedimientos almacenados innecesarios, servicios no utilizados, paquetes de utilidades, cuentas por default).

f) Modifique las credenciales por defecto para las cuantas de sistema o de administración de Bases de Datos. (SA, SYS, Sysadmin, etc)

6. SEGURIDAD EN CLOUD



6.1 Modelos de Servicios

SaaS (Software as a Service)

Las aplicaciones basadas en Cloud Computing se ejecutan remotamente en la propia infraestructura de servidores, las cuales pertenecen a entidades de terceros.

PaaS (Plataform as a Service)

Proporciona un entorno basado en infraestructura Cloud Computing cumpliendo los requisitos necesarios para brindar soporte para la creación y distribución de aplicaciones orientada a la plataforma web.

IaaS (Infraestructure as a Service)

Provee a las empresas solicitantes, recursos informáticos, considerando servidores, redes e incluso espacio en centros especializados en el almacenamiento de datos.

6.2 Modelos de despliegue

Cloud Pública Concierne a empresas de terceros que se encargan de ofrecer acceso instantáneo a los recursos informáticos para organizaciones o personas naturales. De esta forma, los usuarios no necesitan adquirir software, hardware o infraestructura de soporte.

Cloud Privada Pertenece propiamente a una entidad única que controla el modo en que se utilizan los recursos y servicios en distintas líneas de negocio. Adicionalmente de permitir un mayor control, evita la tenencia múltiple.

Cloud Híbrida Utiliza base de Cloud Privada en combinación con el uso estratégico de servicios de Cloud Pública para obtener un ambiente heterogéneo. Esta metodología que facilita la portabilidad común de los datos y de las aplicaciones en función.

6.3 Seguridad en Cloud

a) Los sistemas Cloud deben contar con mecanismos para dar de baja usuarios o remover permiso de acceso en el momento en el que lo considere necesario.

b) Los sistemas Cloud deben ofrecer la posibilidad de asignación de roles y perfiles en base a necesidades de negocio.

c) Los proveedores de servicios Cloud, deben asegurar la confidencialidad, integridad y disponibilidad de la información.

d) Los servicios Cloud deben proporcionar encriptación de las comunicaciones de extremo a extremo, para la protección de claves de acceso, protección y privacidad de datos.

e) Los servicios Cloud deben contener autenticación (soportando múltiples métodos y factores de autenticación).

f) Los servicios Cloud deben incluir control de acceso, con perfiles de acceso, segregación de roles y control de altas y bajas de usuarios

g) Los servicios Cloud deben poseer monitorización y filtrado de tráfico web, servicios de análisis y bloqueo de malware, spyware y arranque desde red, bloqueo de sitios identificados como phishers.

h) Los servicios Cloud deben tener enrutadores, firewalls e IPS para la identificación de intentos de intrusión y de violaciones de las políticas establecidas.

i) Idealmente los proveedores de servicios Cloud, deben proporcionar a sus clientes servicios de escaneo y evaluación de vulnerabilidades, test de intrusión.

j) Los proveedores de servicios Cloud, deben proporcionar a sus clientes servicios de Hardering.

k) Toda transmisión de datos entre dispositivos de usuarios y el servicio en nube debe estar cifrada, bien sea empleando protocolo TLS versión 1.2 o

a través de VPN site-to-site o remotas que empleen al menos algoritmo AES-256.

l) Toda información que se encuentre en el sistema Cloud, deberá estar debidamente inventariado, indicando al menos el proveedor del sistema en nube donde se encuentra, ubicación geográfica, el propietario o dueño de los datos, la clasificación de la información, la forma como se tendrá acceso a ella y el personal, grupos de personas o unidades de negocio que podrán tener acceso a ella, en conjunto con el privilegio.

m) El proveedor de servicio Cloud, debe contar con procedimientos de destrucción de datos tanto física como lógica, que garanticen que los mismos no pueden ser recuperados, independientemente de la razón por la que Mineduc requiera la destrucción de estos.

n) Debe existir un acuerdo de no divulgación de información entre Mineduc y el proveedor de servicio Cloud.

7. CONTROLES DE SEGURIDAD GENÉRICOS

A continuación, se presentan controles que son aplicables a distintos componentes en el ámbito del desarrollo, de los cuales un caso particular son los gestores de contenido (CMS), sin embargo, estos lineamientos son aplicables también otros entornos.

En el caso de los CMS, los fallos en los gestores de contenidos web generalmente son problemas originados por errores en la administración, tanto del CMS como del sistema que lo soporta (hosting –Servidores-BD), o de la instalación de módulos y componentes de terceros que pudiesen derivar de errores de programación en el propio gestor.

7.1 Recomendaciones de seguridad a Nivel de Infraestructura

a) Cambie sus contraseñas regularmente y cambie las credenciales por defecto del software base. Utilice una combinación aleatoria de letras, números o símbolos y evite usar nombres o palabras que puedan ser encontradas en un diccionario o que coincidan con el nombre de la aplicación o sistema.

b) Administre y establezca correctamente roles, sesiones y permisos de accesos al Gestor de Contenidos.

c) Limite los intentos de acceso al panel de administración. Se sugiere utilización Captchas/Recaptcha de Google.

d) Asigne correctamente los permisos de archivos o ficheros que componen el Gestor. (Considerar .htaccess para gestión de directorios).

e) Maneje adecuadamente el control de errores presentados en el Gestor CMS.

f) Mantenga el sistema Gestor actualizado. Al menos una vez al mes realice proceso de actualización, siempre y cuando la versión del CMS escogida cubra las necesidades.

g) Oculte el número de versión del Gestor y panel de administración (Los fallos de gestores habitualmente afectan a versiones concretas).

h) Active opciones de Seguridad propias del Gestor de Contenido, con asesoría del área de Seguridad.

i) Al usar un servicio compartido de hosting, asegúrese de que ningún otro usuario en el servidor pueda ver o acceder a los archivos de su sitio. Por ejemplo, a través de cuentas shell, cpanels, etc.

j) Realice backups o copias de seguridad regularmente de archivos de su sitio y su base de datos.

k) Utilice canales de comunicación cifrados con protocolos de conexiones seguras SSL/TLS.

l) Después de implementar su Gestor, elimine u oculte directorios y archivos con información relevante que no sea necesaria.

m) Para evitar indexación de directorios del Gestor, se sugiere implementar fichero Robots.txt.

n) Utilice un sistema de prevención o detección de intrusos (IDS/IPS) para bloquear y alertar sobre solicitudes HTTP maliciosas. También se recomienda la utilización de WAFs (Web Aplications Firewall) y DBF (Database Firewall).

7.2 Recomendaciones a Nivel de Servidor Web

a) Active y controle los logs de acceso en busca de actividad sospechosa.

b) Establezca parámetros de seguridad básicos relacionados con permisos y accesos.

c) Actualice y aplique todos los parches necesarios para el software base que sustenta al gestor.

7.3 Recomendaciones a Nivel de Servidor Base de Datos (MySQL)

a) Asegúrese que las cuentas de administración o de instalación queden configuradas con acceso limitado, posterior a su implementación.

b) Configurar correctamente los usuarios y los permisos en la Base de Datos.

c) Limitar los accesos de cada usuario a sus propias bases de datos. No deben quedar públicas.

d) Protección de almacenamiento de datos por medio de algún mecanismo de cifrado robusto.

e) Los accesos a la Base de Datos deben ser locales, no remotos.

f) Siga el principio de "Least Privilege" (El menor privilegio)

7.4 Recomendaciones para Extensiones (Componentes, Módulos, y plugins)

a) Limitar y controlar la instalación de módulos de terceros, que no hayan sido auditados y que ofrezcan un nivel de seguridad débil que pueda comprometer la plataforma.

b) Controlar los pluggins, desactivar los que no se utilicen.

c) No utilice extensiones que requieran register_globals ON.

d) Descargue extensiones solo de sitios de confianza.

e) Independientemente de las copias de seguridad planificadas con la frecuencia estipulada por el administrador del sitio, se recomienda que siempre se realice una copia de seguridad de su sitio y de la base de datos, antes de instalar nuevas extensiones.

f) Desinstale cualquier extensión no usada, y revise que los directorios y archivos relacionados hayan sido borrados.

8. CONTROLES DE SEGURIDAD EN APIs

a) Validar todos los datos: cabeceras, query parameters, path parameters y cuerpos petición/respuesta. La validación debe incluir: validación formato, tipo de datos, valor, longitud y/o rango según aplique.

b) Para las funciones de autenticación y gestión de sesiones habilitar comunicación segura con two-way SSL y estándares de autenticación y autorización (como OAuth 2.0 y openId Connect).

c) Utilizar ofuscación de datos, Ofuscación log, criptografía en el canal de comunicación y utilizar Two-way SSL, para la protección frente a la exposición de datos sensibles.

d) Aplicar autenticación con OAuth 2.0, validación de acceso a los recursos con planes y a cada plan asociar un límite de cuota de consumo por unidad de tiempo.

e) Disponer de entornos debidamente separados y protegidos con control de acceso y autorización en todos los niveles, así como mantenerlos actualizados frente a posibles vulnerabilidades haciendo uso de OAuth con scopes y gestionando perfiles con sus correspondientes permisos para su gestión.

f) Aplicar políticas y mediadores con el objeto de validar que las peticiones y las respuestas no contengan scripts o posible presencia de patrones maliciosos.

g) No aceptar objetos serializados a partir de fuentes no confiables o permitir solo serialización que permita tipos de datos primitivos. Si eso no fuera posible, implementar verificaciones de integridad o criptografía de los objetos serializados para evitar la creación hostil de objetos o adulteración de datos. Es importante almacenar en los logs las excepciones y fallas de deserialización.

h) Restrinja o monitoree la conectividad de entrada y de salida y configure alertas para detectar problemas con la serialización.

i) Habilitar políticas que permitan monitorear y alertar las peticiones y respuestas en los logs

j) No dejar parámetros expuestos en la API, como datos sensibles de manera abierta y/o identificadores utilizados en las URL, los que pueden quedar almacenados en el historial del navegador, en memoria y en el servidor de logs de la aplicación.

k) No utilizar la API Key como una credencial autorizada ni almacenar dicha información en el código fuente.

l) Validar las peticiones a todos lo niveles: cabeceras, uri, parámetros de consulta, cuerpo de la petición verificando su estructura y el formato tanto en cabecera como en cuerpo. Limpiar aquello que no sea estrictamente necesario evitando exponer cualquier dato sensible.

m) Aplicar autenticación y autorización de manera rigurosa.

n) Aplicar actualizaciones de las librerías y dependencias de procesadores XML.