**ANEXO No. 8**

**BUENAS PRÁCTICAS DE DESARROLLO PARA SOFTWARE SEGURO**

a) En cuanto al desarrollo de la aplicación, el proveedor deberá tener en cuenta:

1. Garantizar los principios de gestión, fortaleza, usabilidad, escalabilidad y capacidad en los mecanismos de autenticación considerando la sincronización de usuarios con el OID.
2. Definir un mecanismo de autorización de acuerdo con las mejores prácticas vigentes y según las normas NTC-ISO/IEC 27001.
3. En cuanto al manejo de la confidencialidad de la información se deben implementar mecanismos de criptografía para la información sensible.
4. Para todos los desarrollos se deben considerar los lineamientos correspondientes al tratamiento de información confidencial que sean dados por el estado colombiano.
5. Deberán implementarse mecanismos en el código para prevención de ataques informáticos tales como SQL injection y Cross site scripting (XSS), entre otros.
6. Definir e implementar mecanismos para garantizar la integridad de la información (tales como PKI, MAC).
7. Gestión de errores y verificación de bitácoras: Se debe definir mecanismos para el manejo de responsabilidad en las transacciones (tales como Logs, mecanismo de no repudiación adoptado, etc.)
8. Uso de servicios de seguridad para elementos distribuidos (tales como Kerberos, SAML, etc.).
9. Se debe considerar los diferentes aspectos de desempeño en cuanto a proveer los servicios de seguridad y considerar este como posible blanco de ataques.
10. Se deben ejecutar pruebas de vulnerabilidad semestrales para encontrar posibles fallas tanto del software como en la infraestructura, entregando el informe correspondiente y estableciendo un plan de remediación en caso de ser necesario.

b) Para el despliegue de la solución el Contratista deberá implementar alguno de los siguientes framework o alguno técnicamente equivalente:

1. GSS-API – Generic Secure Service Application Program Interface.
2. PAM – Pluggable Authentication Modules.
3. CDSA – Common Data Security Architecture.

c) Por tratarse de un sistema Web el Sistema de Información FONPET (SIF) debe tener las siguientes consideraciones en la capa de presentación:

1. Considerar las implicaciones de autenticación vía Web, y la implementación de infraestructura PKI (uso de certificado digital).
2. Considerar el envío de información sensible a través del método POST.
3. Garantizar la confidencialidad de la información sensible, evitando cualquier tipo de estrategia que persista esta información en el cliente (por ej. cookie, local storage, etc.). Con excepción de la cookie encriptada de sesión.

4. Determinar las estrategias para el manejo de extensiones de archivos expuestos en el servidor Web.
5. Determinar las estrategias para evitar posibles ataques de Cross Site Scripting y SQL Injection.
6. Garantizar un manejo seguro en la preservación de sesiones en el servidor Web.
7. Generar una estrategia adecuada para el manejo adecuado de mensajes de error que puedan revelar información sensible de configuración o ubicación física de archivos.
8. Definir mecanismos de logs con la información necesaria para realizar auditorías de seguridad.

d) Respecto a la capa de negocios se deben tener en cuenta los siguientes aspectos:

1. Determinar de manera detallada los servicios de autenticación, autorización, confidencialidad, integridad, soporte a no-repudiación y administración en esta capa de la solución.
2. Verificar, evaluar y hacer las recomendaciones pertinentes sobre los permisos otorgados a nivel de código y el esquema de detalle de evidencias para verificar identidad (Code Based Acces Control).
3. Verificar, evaluar y hacer las recomendaciones pertinentes sobre los permisos otorgados a nivel de responsabilidades y objetivos de usuarios en la solución (Roled Based Acces Control).
4. Verificar, evaluar y hacer las recomendaciones pertinentes sobre el envío de mensajes local y remotamente de manera segura (Secure communication).
5. Verificar, evaluar y hacer las recomendaciones pertinentes sobre el asegurar que el código no será modificado utilizando soluciones criptográficas ni archivos de distribución firmados (Secure Code and Data Protection).

e) A nivel de la capa de base de datos, se debe considerar igualmente:

1. Estrategia para asegurar al máximo la utilización de los mecanismos de seguridad existentes en el servidor de Base de datos.
2. Documento donde se identifique claramente los diferentes tipos de clientes que tengan acceso a los recursos de la base de datos.
3. Identificar y crear mecanismos selectivos para los diferentes tipos de operaciones.
4. Proveer los mecanismos de seguridad para generar las copias de seguridad requeridas sobre la información especialmente sensible.
5. Proveer los mecanismos de seguridad para recuperar las copias de seguridad requeridas sobre la información especialmente sensible.
6. En la base de datos no se debe guardar información como claves de acceso, ya sea a otros sistemas o del mismo usuario en texto plano.

f) En cuanto a protocolos de comunicación se deben considerar los siguientes aspectos:

1. Realizar pruebas de desempeño en caso de la implementación de SSL haciendo uso de este únicamente para información sensible. (recomendado que todo este por servidor seguro)

2. Para la comunicación interna del aplicativo se debe garantizar que los servidores pertenezcan a un segmento de red (VLAN) diferente al de los usuarios de red, a fin captaciones fraudulentas de la información.
3. Generar documento de recomendaciones para garantizar el uso de los diferentes protocolos seleccionados y su implementación a través de Firewalls.

g) En cuanto al proceso desarrollo de software y despliegue:

1. Debe existir un proceso claramente definido y documentado de los procesos de paso a pruebas y producción, donde se detallen claramente todos los actores involucrados y los recursos necesarios para un proceso seguro del mismo.
2. Se debe contar con mecanismos que garanticen la acreditación de los instaladores a través de los procesos de paso a prueba y paso a producción, de tal forma que no puedan ser alterados.
3. El proponente debe garantizar que el código fuente sea almacenado en un repositorio centralizado de código fuente implementado en la tecnología AZURE DEVOPS Server versión 2019 (mínimo) o en la herramienta que sea determinada por el Ministerio de Hacienda.

h) Verificación:

1. Proveer mecanismos de confirmación o pruebas que permitan verificar cada uno de los puntos aquí detallados.