



TRABAJO DE GRADO

“ANÁLISIS DETALLADO DE VULNERABILIDADES EN LA APLICACIÓN WEB  
DE ADMINISTRACIÓN DE TOKENS BANCARIOS DE UNA ENTIDAD  
FINANCIERA EN COLOMBIA”

EDGAR ISAURO CORREDOR MORALES

UNIVERSIDAD CATÓLICA DE COLOMBIA

FACULTAD DE INGENIERÍA

PROGRAMA DE ESPECIALIZACIÓN EN SEGURIDAD DE LA  
INFORMACIÓN

BOGOTÁ D.C.

2020

TRABAJO DE GRADO

“ANÁLISIS DETALLADO DE VULNERABILIDADES EN LA APLICACIÓN WEB  
DE ADMINISTRACIÓN DE TOKENS BANCARIOS DE UNA ENTIDAD  
FINANCIERA EN COLOMBIA”

EDGAR ISAURO CORREDOR MORALES

Trabajo de grado presentado para optar al título de Especialista en  
Seguridad de la Información

INGENIERA SANDRA MILENA BERNATE BAUTISTA

COORDINADORA ESPECIALIZACIÓN DE SEGURIDAD DE LA  
INFORMACIÓN

UNIVERSIDAD CATÓLICA DE COLOMBIA

FACULTAD DE INGENIERÍA

PROGRAMA DE ESPECIALIZACIÓN EN SEGURIDAD DE LA  
INFORMACIÓN

BOGOTÁ D.C.

2020



## Atribución-NoComercial 2.5 Colombia (CC BY-NC 2.5)

La presente obra está bajo una licencia:  
**Atribución-NoComercial 2.5 Colombia (CC BY-NC 2.5)**  
Para leer el texto completo de la licencia, visita:  
<http://creativecommons.org/licenses/by-nc/2.5/co/>

### Usted es libre de:



Compartir - copiar, distribuir, ejecutar y comunicar públicamente la obra  
hacer obras derivadas

### Bajo las condiciones siguientes:



**Atribución** — Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciante (pero no de una manera que sugiera que tiene su apoyo o que apoyan el uso que hace de su obra).



**No Comercial** — No puede utilizar esta obra para fines comerciales.

## TABLA DE CONTENIDO

1.	INTRODUCCIÓN .....	7
2.	GENERALIDADES.....	11
2.1.	<b>Línea de Investigación.</b>	11
2.2.	<b>Planteamiento del Problema.</b>	11
2.3.	<b>Antecedentes del Problema.</b>	13
2.4.	<b>Pregunta de investigación</b>	16
2.5.	<b>Variables del Problema</b>	16
2.6.	<b>Justificación</b>	17
3.	OBJETIVOS .....	21
3.1.	<b>Objetivo general</b>	21
3.2.	<b>Objetivos específicos</b>	21
4.	MARCOS DE REFERENCIA.....	22
4.1.	<b>Marco Conceptual.</b>	22
4.2.	<b>Marco Teórico.</b>	24
4.3.	<b>Marco Jurídico.</b>	27
4.4.	<b>Marco Geográfico.</b>	29
4.5.	<b>Marco Demográfico.</b>	30
5.	METODOLOGÍA.....	32
5.1.	<b>Fases del Proyecto en General.</b>	32
5.2.	<b>Instrumentos o herramientas utilizadas.</b>	33
5.3.	<b>Población y Muestra.</b>	33
5.4.	<b>Modelo de evaluación y enmarcación.</b>	34
5.5.	<b>Alcance y limitaciones</b>	35
6.	EJECUCION DE LA METODOLOGIA.....	36
6.1.	<b>Aplicación Administración de Tokens.</b>	36
6.1.1.	<b>Proceso de cargue de semillas.</b>	36
6.1.2.	<b>Autenticación del Cliente en Banca Virtual.</b>	37
6.1.3.	<b>Proceso de ingreso de semillas al inventario.</b>	38
6.1.4.	<b>Creación de la Solicitud de Tokens por parte del Cliente.</b>	40
6.1.5.	<b>Asignación de Tokens a la Solicitud radicada por el proceso automático.</b>	40
6.2.	<b>Herramienta Base de Desarrollo del Aplicativo.</b>	41
6.2.1.	<b>Filosofía de GeneXus.</b>	42
6.2.2.	<b>Desarrollo de una aplicación con la herramienta GeneXus.</b>	43
6.2.3.	<b>Objetos GeneXus.</b>	44
6.2.4.	<b>Desarrollo Incremental.</b>	44
6.3.	<b>Modelo GeneXus de Aplicación Administración de Tokens.</b>	45
6.4.	<b>GeneXus y la Versión para Evaluar la Seguridad.</b>	46
6.4.1.	<b>Migración de la Aplicación a La Gx Evolution 3.</b>	46
6.5.	<b>GeneXus Consulting. Mejores Prácticas de Desarrollo.</b>	47
6.5.1.	<b>GeneXus Consulting.</b>	47

6.5.2.	Mejores prácticas.	48
6.6.	Security Scanner. DLL.	48
6.6.1.	Proceso de Instalación.	48
6.6.2.	Pantalla herramienta Scanner en Gx-Evo 3.	51
6.7.	GeneXus (GX) y el OWASP Top Ten 2017.	60
6.7.1.	A1 -Inyección.	61
6.7.2.	A2 -Autenticación Rota.	63
6.7.3.	A3 -Exposición de datos sensibles.	67
6.7.4.	A4 -Entidades Externas XML (XXE).	68
6.7.5.	A5 -Pérdida de control de acceso.	69
6.7.6.	A6 -Configuración de seguridad incorrecta.	70
6.7.7.	A7 -Cross-Site Scripting (XSS) .	70
6.7.8.	A8 -Deserialización Insegura.	71
6.7.9.	A9 -Uso de Componentes con Vulnerabilidades Conocida.	72
6.7.10.	A10 - Registro y Monitoreo Insuficientes.	73
7.	RESULTADOS Y PRODUCTOS OBTENIDOS.....	74
7.1.	Nomenclatura de objetos.	74
7.2.	Scanear la Aplicación de Administración de Tokens.	74
7.3.	Tabla Referencia del escáner de seguridad OWASP top 10 2017.	76
7.4.	Resultado General del Scaneo con Seguridad para GeneXus.	76
7.5.	Síntesis del análisis y extracción de lo vulnerable e importante.	81
7.6.	Vulnerabilidades de características especiales y su tratamiento.	83
7.7.	Análisis de resultados e impactos.	84
7.8.	Debilidades en el uso de Tokens criptográficos.	85
8.	CONCLUSIONES.....	87
9.	BIBLIOGRAFIA .....	89

## LISTA DE FIGURAS

FIGURA. 2-1. <b>ALGUNAS CIFRAS DEL SISTEMA COLOMBIANO EN MATERIA DE INCLUSIÓN BANCARIZADA.</b> .....	12
FIGURA. 2-2. <b>INTERNET IMPULSA LA INCLUSIÓN FINANCIERA EN EL MUNDO.</b> .....	13
FIGURA. 2-3. <b>IMAGEN RADIOGRAFÍA DE DELITOS INFORMÁTICOS EN COLOMBIA EN 2015</b> .....	15
FIGURA. 2-4. <b>NÚMERO DE OPERACIONES (MONETARIAS Y NO MONETARIAS)</b> .....	16
FIGURA. 2-5. <b>DELITOS CIBERNÉTICOS EN COLOMBIA.</b> .....	19
FIGURA. 2-6. <b>PENAS Y DELITOS CIBERNÉTICOS EN COLOMBIA.</b> .....	20
FIGURA. 4-1. <b>CUADRO MARCO TEÓRICO REFERENCIA</b> .....	25
FIGURA. 4-2. <b>GLOBALIZACIÓN – COLOMBIA</b> .....	30
FIGURA. 5-1. <b>METODOLOGÍA DEL PROYECTO</b> .....	32
FIGURA. 5-2. <b>METODOLOGÍA PARA DESARROLLO DE TRABAJO DE INVESTIGACIÓN</b> .....	33
FIGURA. 5-3. <b>SEGURIDAD INFORMÁTICA. OWASP TOP 10 2013 vs OWASP TOP 10 2017</b> .....	34
FIGURA. 6-1. <b>PROCESO DE CARGUE DE SEMILLAS</b> .....	36
FIGURA. 6-2. <b>AUTENTICACIÓN EN BANCA.</b> .....	37
FIGURA. 6-3. <b>AUTENTICACIÓN APLICACIÓN</b> .....	38
FIGURA. 6-4. <b>SISTEMA CARGUE SEMILLAS</b> .....	39
FIGURA. 6-5. <b>EJEMPLO DE CARGUE DE SEMILLAS</b> .....	39
FIGURA. 6-6. <b>PANTALLA DE ASIGNACIÓN INDIVIDUAL DE TOKENS A LA SOLICITUD.</b> .....	40
FIGURA. 6-7. <b>FILOSOFÍA DE TRABAJO CON GENEXUS.</b> .....	42
FIGURA. 6-8. <b>NAVEGADOR DE WINDOWS</b> .....	45
FIGURA. 6-9. <b>MODELO MIGRADO</b> .....	46
FIGURA. 6-10. <b>IMAGEN DESARROLLADOR Gx EVOLUTION 3</b> .....	49
FIGURA. 6-11. <b>NAVEGADOR DE WINDOWS</b> .....	49
FIGURA. 6-12. <b>EJECUCIÓN ADICIÓN ESCANER AL MODELO.</b> .....	50
FIGURA. 6-13. <b>CONFIGURACIÓN PARA EJECUCIÓN DEL ESCANER.</b> .....	51
FIGURA. 6-14. <b>OBJETOS A SER REVISADOS</b> .....	52
FIGURA. 6-15. <b>REGLAS CONFIGURADAS EN ESCANER</b> .....	53
FIGURA. 6-16. <b>DATOS DESTINO INFORME DE SALIDA DEL ESCÁNER.</b> .....	59
FIGURA. 6-17. <b>OWASP. – FLUJO ATAQUES.</b> .....	60
FIGURA. 6-18. <b>OWASP. APARTES.</b> .....	61
FIGURA. 7-1. <b>Gx 90 – Gx Evo 3</b> .....	74
FIGURA. 7-2. <b>ARCHIVO DESTINO</b> .....	75
FIGURA. 7-3. <b>EXCEL DESTINO.</b> .....	75
FIGURA. 7-4. <b>TRANSCRIPCIÓN DE CÓDIGOS OWASP.</b> .....	76
FIGURA. 7-5. <b>RESUMEN VULNERABILIDADES</b> .....	79
FIGURA. 7-6. <b>COMPOSICIÓN VULNERABILIDADES</b> .....	80
FIGURA. 7-7. <b>CÓDIGOS Y COLORES VULNERABILIDADES.</b> .....	81
FIGURA. 7-8. <b>RESUMEN MÁS EXPLÍCITO VULNERABILIDADES.</b> .....	81
FIGURA. 7-9. <b>RESUMEN DEL RESUMEN DE VULNERABILIDADES</b> .....	82
FIGURA. 7-10. <b>RESULTADOS E IMPACTOS.</b> .....	85

## 1. INTRODUCCIÓN

La Superintendencia Financiera de Colombia (SFC) como organismo regulador de las entidades financieras colombianas, mediante la circular externa 007 de octubre de 2018 fortalece la protección de la información de los consumidores financieros ante todos los riesgos, que son innumerables de ciberseguridad en la realización de las pasarelas de pagos.

La SFC ha impartido nuevas instrucciones para todas las entidades financieras de Colombia, estándares de seguridad que obligatoriamente deben adoptarse en las pasarelas de pago y dentro de las instituciones para fortalecer la protección de la información de los consumidores financieros.

De acuerdo con la ley 1328 de 2009 el Consumidor Financiero es todo cliente, usuario o cliente potencial de los productos y servicios ofrecidos por las entidades sometidas a la inspección y vigilancia de la Superintendencia Financiera y además todo aquello que lo determine la ley.

Un sistema de control interno también debe adoptarse al interior de cada institución. El sistema de control interno es el conjunto de políticas, normas, principios, procedimientos y mecanismos de evaluación y verificación que son establecidos por las juntas directivas y las altas direcciones de las organizaciones para buscar un grado de seguridad razonable para la consecución de objetivos, entre los cuales se encuentran:

- Gestionar el riesgo adecuadamente.
- Aumentar la confiabilidad y la operación de la información en custodia.
- Cumplir las normas legales y reglamentarias.
- Prevenir y mitigar la ocurrencia de los fraudes.
- Mejorar la eficiencia de las operaciones financieras.

La adopción de estas normas, ayudan a generar la confianza que los clientes financieros necesitan depositar en las entidades financieras de un país; e incluso la confianza y buen nombre de las financieras internacionales.

Por citar alguna de las inclusiones comentadas por el ingeniero Daryan Reinoso de Symantec; Organización estadounidense dedicada al desarrollo de productos con un alto grado de seguridad de la información: “Uno de los delitos más comunes es el robo de la información de los clientes bancarios, y se conoce como ‘troyanos financieros’, esta es una de las amenazas que viene en aumento”, indicó Reinoso y añadió que el cibercrimen mueve anualmente cerca de 10.000 billones de dólares en el mundo.

Nada más observemos esta cifra de dinero, tentadora para los malintencionados delincuentes informáticos que no se sabe dónde están, pero que afecta enormemente la confianza financiera de los clientes que buscan proteger sus intereses económicos a todo dar.

Uno de los mecanismos que obligatoriamente proporciona a los clientes seguridad es el uso de los Tokens Bancarios, a través de los cuales, se suma seguridad a los movimientos financiero que deben hacer los clientes cuando se enfrentan a las plataformas que ofrecen los bancos para realizar sus operaciones de movimiento de dinero. Transferencias bancarias internas o externas hacia otras cuentas de interesados, pago de servicios públicos, pago de obligaciones financieras, consultas de saldos, inscripción de cuentas para transferencias, movimientos internos entre cuentas del mismo cliente, etc. Como se ve, hay innumerables oportunidades para la intersección malintencionada de los ciber-delincuentes.

Los Token bancarios tienen un algoritmo igual al que se tiene en el servidor que se encuentra en la entidad financiera y lo que los hace efectivos es el sincronismo que debe haber entre las dos plataformas de confrontación.

Inicialmente, solo era necesario digitar una clave asignada para aprobar las transacciones bancarias en un ambiente virtual, sin embargo, después la seguridad se vio afectada con el creciente aumento de operaciones fraudulentas, afectando los intereses de los clientes bancarizados, así como la confianza del banco al permitir los desembolsos no realizados por los titulares de las cuentas, debido a esto en Colombia en el año 2010 la Superintendencia Financiera Colombiana decretó la implementación de una segunda clave para los usuarios a la hora de realizar las operaciones, en busca de brindar seguridad.

Esta segunda clave fue también estática y con el tiempo y el incremento de fraudes, llegó a convertirse en dinámica con la ayuda de las tecnologías de software y hardware, es así como hoy en día este código dinámico es producido por demanda cada cierta cantidad de tiempo. Generando así que muchos clientes bancarios sintieran sensación de confianza al realizar transacciones a través de la banca virtual dejando de transportarse a las sucursales bancarias, agilizando sus pagos y transacciones, evitando las largas filas e incrementando el comercio en el ciberespacio, de ahí el creciente número de comercios en la web y servicios de todo tipo transando desde la comodidad del hogar.

Haciendo referencia a todas las posibilidades que tienen los delincuentes para

poder invadir los espacios web y lograr sus intenciones; se hace necesario examinar los diferentes desarrollos web y profundizar de acuerdo a las experiencias por donde las aplicaciones ofrecen espacios que parecieran infalibles de vulnerar. Dando la posibilidad de estudiar la aplicación de administración de tokens para descubrir estas posibilidades e informar de estas posibilidades para las entidades bancarias estandaricen y adopten métodos de programación seguros que proporcionen la seguridad que se necesita para generar confianza.

De acuerdo con lo contextualizado anteriormente, el proyecto busca responder a la pregunta sobre. ¿cómo establecer en términos de ciberseguridad los posibles riesgos que pueden afrontarse para evaluar la confiabilidad, integridad y disponibilidad de la aplicación que apoya el proceso?, a partir de un análisis de los procesos de la aplicación que maneja los tokens de seguridad bancaria.

En el presente documento se encuentra la estructura de planeación del proyecto en la que se incluyen los objetivos, cronograma, así como la metodología a desarrollar entre otros aspectos que intervienen en la construcción del mismo, esperando que sea de gran interés y ayuda a quienes necesitan ahondar en el tema de desarrollo seguro de las aplicaciones bancarias.

En términos de inspección de la aplicación a la cual se hace el estudio para apreciar muchas de las inconsistencias que pueden generar la posibilidad de que se pueda acceder a ella, a sus bondades y a su base de datos para estropear su funcionamiento de ataque mal intencionados en beneficio de quien los haga. Para este estudio se muestran los procesos a través de los cuales la aplicación genera sus atenciones de administración de Tokens y alimenta la seguridad de los accesos por internet de los servicios bancarios. Pero puede ser vulnerada con puertas abiertas que se dejan por falta de experiencia en el desarrollo seguro de las aplicaciones.

Una rápido vistazo a los procesos de apoyo de la aplicación muestran entre ellos el cargue de semillas o Tokens a la aplicación para la administración de los mismos, en este proceso se presume la posibilidad de acceder de alguna forma a la base de datos (BD) y al servidor donde reside la aplicación, puesto que para subir información a los archivos de la base de datos es necesario copiar en un sitio intermedio del contexto web el archivo físico para luego hacer la operación de lectura e inserción de registros en las tablas de la BD. Hay también posibilidades que donde se hacen llamados con código HTML o uso de lenguaje SQL se acceda con facilidad a cambiar instrucciones que den otro rumbo a la aplicación.

Para este desarrollo se estudia de una forma metódica y cronológica por experiencias vividas que la forma de hacer el análisis debe describir varios pasos entre los cuales se encuentran como finalidad la aplicación de un scanner que

pueda presentar las posibilidades de acceso no autorizado a la aplicación. Previamente debe valorarse las versiones del GeneXus en donde se puede hacer uso de esa herramienta, como no es posible en la versión de desarrollo de la aplicación, es necesario subirla a una versión en donde pueda vincularse la posibilidad de ejecutar los procesos de vigilancia de los objetos desarrollados. Finalmente, los resultados deben presentar el estado de la aplicación mostrando en donde puede ser vulnerada y violada la seguridad con la recomendación de lo que debe realizarse para asegurar su operación sin las posibilidades de ser atacada.

## 2. GENERALIDADES

### 2.1. Línea de Investigación.

El proyecto está enmarcado en una línea de software inteligente y de convergencia tecnológica.

Es inteligente porque muestra la realidad de un dispositivo que va a ser entregado a un cliente que deposita toda su confianza en él, ayudando a que sus transacciones bancarias y financieras sean seguras; aplican convergencia tecnológica porque se centran en una realidad de confianza, aplican dinamismo a sus movimientos financieros y cada vez son más modernos en diferentes términos visuales, inter-plata-fórmicos, suman más seguridad y buscan ser la mano derecha de los cuentahabientes financieros.

La investigación es de tipo evaluativa, que desea estudiar, evaluar, cotejar y dar juicio sobre los diferentes procesos involucrados en el otorgamiento del token bancario a través de un sistema de administración de tokens.

Tiene enfoque cuantitativo porque se pretende recolectar información técnica, basados en la ejecución de pruebas, análisis y evaluación sin buscar una medida numérica para establecer los puntos vulnerables de la aplicación, pero si proponer una solución inmediata a las necesidades de ajustes que aplique seguridad a la aplicación de Administración de Tokens.

### 2.2. Planteamiento del Problema.

“Elevado crecimiento de desconfianza en el uso de las plataformas financieras web en materia de ciberseguridad bancaria”

Gracias al crecimiento de los usuarios del sistema financiero, y continua búsqueda de la bancarización de la sociedad (**¡Error! No se encuentra el origen de la referencia.**), se creó la necesidad de brindar a los usuarios instrumentos que aseguren el movimiento de sus cuentas bancarias.

Desde que la Superintendencia Financiera de Colombia obligó a las entidades financieras a fijar instrumentos de seguridad para sus clientes; estas entidades empezaron con la búsqueda de mejores herramientas para lograr brindar mayor seguridad a sus clientes. Logrando así que muchas de las entidades se decidieran por la adopción del Token, por la confianza que generaba, y por la independencia que se lograba con su uso. Por lo que iniciaron el proceso de adopción y socialización a sus clientes sobre este dispositivo.

La Superintendencia Financiera no dio un lineamiento explícito sobre lo que debía hacer cada entidad financiera para lograr seguridad en las transacciones virtuales

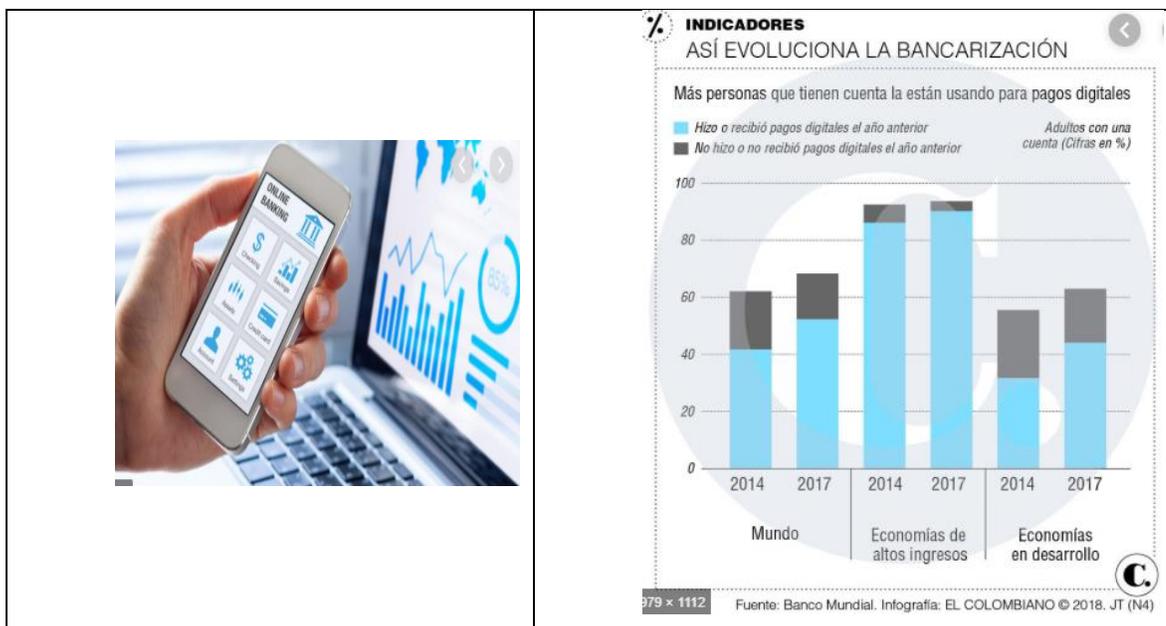
de sus clientes; cada entidad podía escoger la forma segura con la que resguardaría el dinero de sus usuarios. Esto generó para el caso del Token, que el proceso al interior se fuera asimilado de forma diferente, pero con similitudes en su accionar, asociación, y procesos de trazabilidad interna. Y como cada entidad lo hacía según los recursos disponibles para los procesos en los que se intervienen para la consecución, solicitud y entrega de los Token, entonces existen vacíos de seguridad, y riesgos no cubiertos o que no están siendo identificados por las entidades en su totalidad y además por el desconocimiento de los estándares programáticos en algunos casos.

Figura. 2-1. Algunas cifras del Sistema Colombiano en materia de inclusión bancarizada.



Fuente: <https://www.misfinanzasparainvertir.com/retos-economicos-para-el-pais-inclusion-financiera/>

Figura. 2-2. Internet impulsa la inclusión financiera en el mundo.



Fuente: El colombiano (<https://www.elcolombiano.com/negocios/internet-fomenta-la-inclusion-financiera-KM8718408>)

### 2.3. Antecedentes del Problema.

La transabilidad bancaria ha venido evolucionando gracias al desarrollo tecnológico y electrónico fundamentado en avances e investigación aplicada a los procesos de la vida diaria. En los años 80, las transacciones financieras eran engorrosas y sometidas a la presencia de los cuentahabientes en las entidades bancarias; todo se hacía a través de talonarios que identificaban a los clientes y a sus movimientos financieros, claro está que el número de personas era muy pequeño. Las nóminas se pagaban a través de sobres por empleado y en efectivo, procedimientos muy engorrosos. Luego se desarrollaron las bandas magnéticas e inició el uso la era del plástico con la aparición de los dispensadores de dinero; los cajeros electrónicos, algo rudimentarios, pero que también han tenido su desarrollo práctico y multiplicador en servicios a los clientes. En las tarjetas era muy importante proporcionar una clave de seguridad para poder tener acceso a los servicios al ser identificado con éxito. Algunos dispositivos necesitaban de una segunda clave para efectuar una identificación aún más efectiva; aparecen los malintencionados y logran copiar las bandas para apoderarse de las sumas de dinero de los clientes; y también aparecen los que cambian dispositivos en los cajeros electrónicos para

tomar fotos y videos de quienes los usan, para posteriormente copiando la banda magnética de las tarjetas acceder a sus cuentas con la clave que han podido extraer de los dispositivos.

Gracias a los avances de la electrónica y de los desarrollos de programación van apareciendo las aplicaciones que tímidamente consultan las cuentas de los clientes y hacen algunas operaciones simples; para tener acceso se ajusta un usuario y una clave que el cliente debe asumir como suya para poder acceder a sus productos; vuelven a jugar un papel muy importante los criminales cibernéticos y nacen los ciberdelincuentes que son capaces de crear código malicioso y engañar a los inocentes clientes para robarles su información de acceso personal y ejecutar sus fechorías.

Hasta aquí junto con otras tecnologías el cliente era el que tenía la responsabilidad de la confidencialidad de sus accesos, todo esto en las aplicaciones web primitivas de Banca Virtual. Ya desde antes se aplicaba la criptografía para proteger la información y consistía en que la información viajara protegida para que los insertadores de datos no la pudieran descubrir con facilidad. El Token nació por los años 2009 con la creación del Bitcoin ó criptomoneda protegida. Esta medida de seguridad se ha desarrollado y ha venido a incorporarse a las aplicaciones financieras de tal manera que se comparte la seguridad ayudando a que algunos datos de seguridad sean proporcionados y administrados por los clientes y otros por las entidades financieras para agregar mayor seguridad a los movimientos de dinero. Es por esto que damos gran importancia a su administración y afinamiento en términos de seguridad.

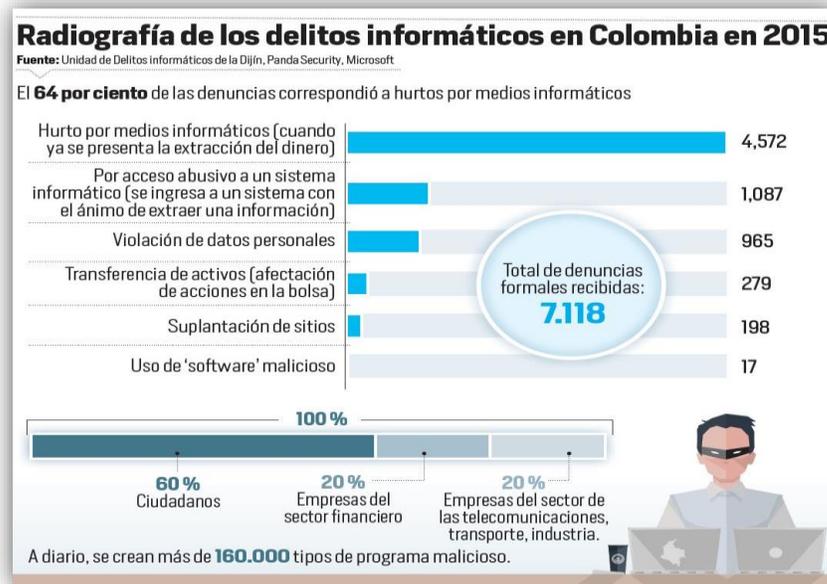
Miles de millones de operaciones bancarias son realizadas en todo el mundo tanto personalmente como a través del ciberespacio; las primeras tienen la seguridad de que quien las está realizando es quien dice ser, y quien fue certificado por la entidad para acreditarse como titular de la cuenta o quien ha sido enviado por el mismo titular para realizar las operaciones, además que no se está conectando al ciberespacio donde es posible que intrusos puedan intersectar sus operaciones bancarias.

Los segundos; los que realizan las operaciones a través del ciberespacio se exponen a la posibilidad de que sus operaciones puedan ser atacadas por los criminales que capturan muchas posibilidades de acceder a sus cuentas y lograr sus objetivos con la intención de apoderarse de lo ajeno afectando tanto los intereses individuales como la credibilidad y confianza que los terceros esperan depositar en las entidades.

Más del 55% de los casos de cibercrimen son a cuentas bancarias. En 2018 el comando de cibercrimen de la policía nacional recibió 12.014 denuncias por robo

mediante medios informáticos. (i1)

Figura. 2-3. Imagen Radiografía de delitos informáticos en Colombia en 2015



**Fuente:** Tomado de: (Periodico el Colombiano, 2018). Consultado el 2 de junio del 2018. Disponible en URL: <http://www.elcolombiano.com/colombia/ciberataques-en-colombia-sexto-pais-mas-vulnerable-en-la-region-AB8535174>

Los servicios bancarios han cambiado debido a la incorporación de las nuevas tecnologías, con ayuda de las famosas cajas fuertes tecnológicas; cifrado de la información, la identificación biométrica, la incorporación de los tokens, entre otros. Pero con ellos también han aumentado los riesgos con los delitos cibernéticos.

La banca en línea utiliza celular y con ellos, la incorporación de ayudas seguras con los tokens electrónicos que se sitúan en los dispositivos móviles para hacer más seguras las operaciones. El 86% de las personas que utilizan celular, y que están entre los 25 y 35 años de edad utilizan banca en línea. Nadie quiere hacer cola en los bancos y desperdiciar tiempo, muchos comparten usuarios y claves, y muchos acceden a los sitios financieros desde puntos públicos, y todo esto son malos hábitos de los clientes que son aprovechados por los cibercriminales.

Se realizaron más de 5.400 millones de operaciones en el año 2.017 con un incremento del 11% respecto del año anterior. Para el año 2.018, el canal de Internet fue el de más auge en las operaciones bancarias. (ii2)

Las crecientes operaciones financieras son las que redundan en importancia del porqué hay que ponerle atención al tema del afinamiento del uso del Token y de su administración segura.

He aquí las cifras hasta el 2017.

Figura. 2-4. Número de operaciones (Monetarias y no monetarias)

Número de Operaciones (monetarias y no monetarias)				
Canal	2014	2015	2016	2017
Internet	1.376.646.150	1.905.341.076	2.295.131.790	2.576.621.515
Datáfonos	413.158.092	460.510.198	516.618.932	568.531.271
Oficinas	700.644.424	664.830.147	655.514.932	615.188.401
Cajeros Automáticos	705.493.171	732.473.320	760.247.270	801.598.435
Telefonía Móvil	119.014.902	132.811.894	197.331.398	330.352.155
Corresponsales Bancarios	118.495.575	147.531.436	184.076.395	235.455.467
Audio Respuesta	94.456.040	93.280.629	98.449.892	112.688.908
ACH	96.256.151	101.734.031	111.933.940	112.047.587
Pagos Automáticos	92.616.193	94.672.878	106.835.895	109.622.735
<b>Total</b>	<b>3.716.780.698</b>	<b>4.333.185.609</b>	<b>4.926.140.444</b>	<b>5.462.106.474</b>

Fuente: (Revista Dinero, 2018). <https://www.dinero.com/economia/articulo/operaciones-financieras-en-colombia-en-2017/256283>

## 2.4. Pregunta de investigación

“En un Sistema de Administración de Tokens Bancarios, ¿cómo establecer en términos de ciberseguridad con base en el top 10 de OWASP los posibles riesgos que pueden afrontarse para evaluar la confiabilidad, integridad y disponibilidad de la aplicación que apoya el proceso? “

## 2.5. Variables del Problema

Innumerables puntos se han tomado para analizar la problemática que se genera en el uso de las nuevas tecnologías para apoyar los movimientos bancarizados de los clientes financieros.

### **Variables internas:**

- Conocer la navegación de la aplicación para elaborar un mapa de posibles riesgos de exposición. (Cualitativa).
- Adaptabilidad a los estándares de programación propuestos). (Cualitativa).
- Establecer los riesgos, y proponer estándares de programación para aumentar la seguridad de la aplicación. (Cualitativa).
- Análisis de riesgos que podrían estar sucediendo en la aplicación de Tokens bancarios. (Profundización en los estándares de programación aplicando normas efectivas y de protección). (Cualitativa).
- Conocer la situación actual de la Aplicación. (Escanear los objetos para conocer su estado de vulnerabilidad). (Cualitativa).

### **Variables externas:**

- Confiabilidad de la información. (Garantizar el aseguramiento y bloqueo de ataques informáticos). (Cualitativa).

## **2.6. Justificación**

Como se dice en la introducción, y al planteamiento del problema, es factible y responsable la realización de este proyecto ya que permite la identificación clara y precisa de las vulnerabilidades que pueden tener los desarrollos que apoyan con especial interés la administración de los Tokens en una entidad bancaria.

Es de gran importancia asegurar que la información producida por la entidad financiera mantenga la confiabilidad, integridad y disponibilidad que los clientes consumidores financieros buscan al encomendar sus activos a la banca nacional

Ahora bien, Proteger los intereses del negocio bancario minimizando los riesgos en ciberseguridad por fugas de información que pueda ser aprovechada por terceros mal intencionados, esta es una prioridad para quien custodia los intereses de los consumidores bancarios.

En términos legales, el tema, da cumplimiento a la Circular externa 007 de 2018 de la Superfinanciera de Colombia.

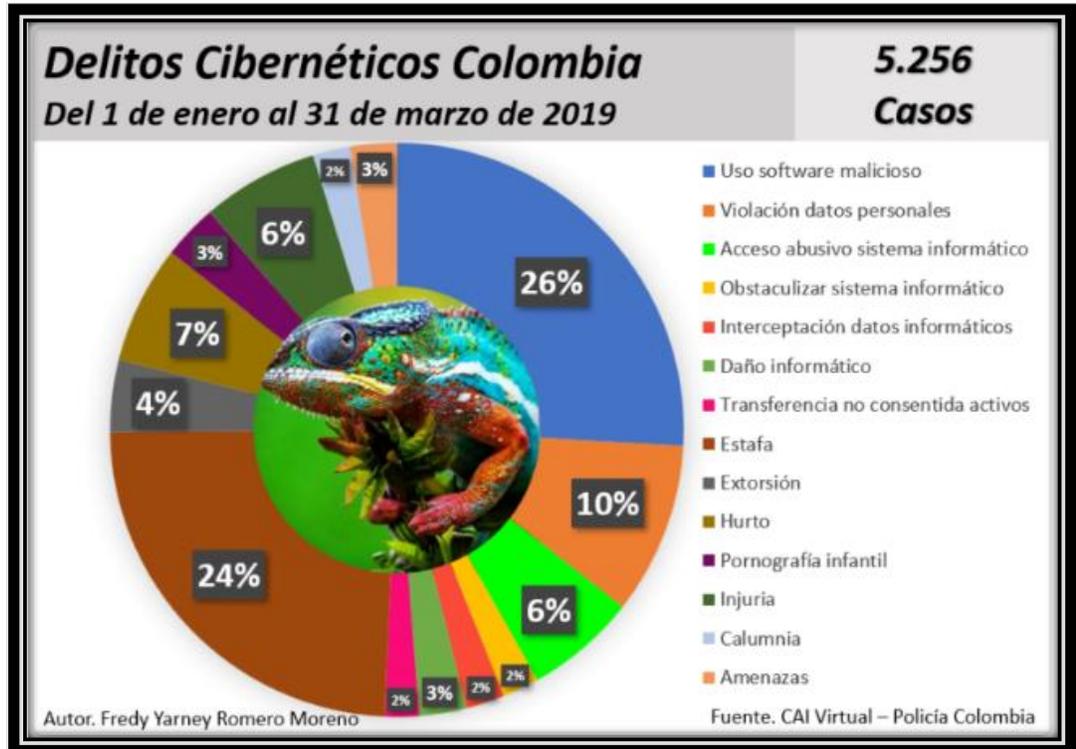
Colombia ocupa el sexto lugar en entre los países de Latinoamérica con mayor número de ataques detectados. (iii3)

Colombia está en el Ranking de países con más ciberataques en Latinoamérica. Los primeros son Argentina, Chile y Brasil. (iv4)

El Ciberdelito es un término general que hace referencia a todo delito a toda actividad delictiva llevada a cabo mediante dispositivos informáticos a través de las redes de internet. El ciberdelito hace referencia sobre varios métodos y herramientas como el phishing, spyware, los virus, ransomware o la ingeniería social, normalmente con el objetivo de robar información personal o de realizar actividades fraudulentas aprovechando la ingenuidad de los usuarios o las vulnerabilidades que tienen las aplicaciones que ofrecen los bancos.

En los tres primeros meses del año 2.019 han sido reportados más de 5250 casos de delitos informáticos en las plataformas de la policía nacional colombiana. Lo muestra la gráfica siguiente. (v5)

Figura. 2-5. Delitos Cibernéticos en Colombia.



Fuente: <https://fyaromo.com.co/2019/04/07/ciberdelitos-en-colombia-corte-a-31-de-marzo-de-2019/>  
Consulta del 30 de septiembre de 2019

Se estima que los incidentes no reportados están por encima de los 50.000. Todas estas cifras son demasiado preocupantes y eso que son solo de los incidentes reportados.

Estas son las penas y multas que contempla la legislación colombiana en la Ley 599 del 2.000; las cuales son muy importantes y nada despreciables para quienes las tienen que asumir; se resume en el gráfico siguiente:

Figura. 2-6. Penas y Delitos Cibernéticos en Colombia.

Código Penal Colombiano			
Artículo	Delito	Pena Meses	Multa
218	Pornografía con personas menores de 18 años	120-240	150-1500
220	Injuria	16-54	13-1500
244	Extorsión	192-288	800-1800
246	Estafa	32-144	66-1500
269A	Acceso abusivo a un sistema informático	48-96	100-1000
269B	Obstaculización ilegítima de sistema informático o red de telecomunicación	48-96	100-1000
269C	Interceptación de datos informáticos	36-72	-
269D	Daño informático	48-96	100-1000
269E	Uso de software malicioso	48-96	100-1000
269F	Violación de datos personales	48-96	100-1000
269G	Suplantación de sitios web para capturar datos personales	48-96	100-1000
269H	Circunstancias de agravación punitiva (de la ½ a las ¾): Servidores públicos, financiera, fines terroristas, entro otras.	Aumento de la ½ a las ¾	-
269I	Hurto por medios informáticos y semejantes	60-120	-
269J	Transferencia no consentida de activos	48-120	200-1500
347	Amenazas	48-96	13-150

Fuente: <https://yaromo.com.co/2019/04/07/ciberdelitos-en-colombia-corte-a-31-de-marzo-de-2019/>  
 Consulta del 30 de septiembre de 2019

### **3. OBJETIVOS**

#### **3.1. Objetivo general**

Evaluar la aplicación web de administración de tokens bancarios con base en el top 10 de OWASP, para identificar fallas que pongan en riesgo la seguridad de la información de los tarjetahabientes, usuarios de servicios web y móviles.

#### **3.2. Objetivos específicos**

- Identificar los procesos de la aplicación web de administración de tokens bancarios que puedan comprometer la seguridad de la información.
- Evaluar los procesos de la aplicación, en los cuales la forma de programación facilite acciones maliciosas de terceros permitiendo que logren sus objetivos malintencionados.
- Recomendar las posibles acciones que minimicen los atentados contra la seguridad de la información en los desarrollos informáticos del aplicativo.

## 4. MARCOS DE REFERENCIA.

### 4.1. Marco Conceptual.

El sistema financiero de un país y también de manera global, permite la circularización de la moneda tanto nacional como extranjera, este movimiento hace dinámico el negocio del dinero produciendo ganancias para unos y beneficios para otros.

Todos estos movimientos financieros se ven armonizados por la cantidad de transacciones que se efectúan a diario entre cuentas, bancos, entre cuentas y empresas que prestan servicios, etc.

La circularización del dinero se ejecuta armonizando los intereses de muchos cuentahabientes para agilizar la economía de todos los entes que participan en el juego financiero; esto se hace a través de aplicaciones informáticas que facilitan a los clientes cumplir con sus obligaciones financieras. Igualmente, estas aplicaciones están provistas de innumerables mecanismos de seguridad que protegen la incursión de los clientes en un mundo mágico y hasta desconocido para ellos; es por eso que debe buscarse detectar todos los puntos factibles de ser atacados para sacar provecho del dinero ajeno.

El sistema financiero tiene su base en **Productos financieros**, que son definidos como las unidades de trazabilidad existentes en las entidades financieras y ofrecidas a los clientes para el manejo de sus dineros. Dentro de los principales productos financieros se encuentran las cuentas de ahorros, cuentas corrientes, depósitos a término CDTs, unos captan recursos y otros entregan. Algunos de ellos están administrados por **las Fiduciarias**, que son entidades financieras con profesionales expertos en el movimiento de dinero con figuras económicas y que están alerta para aprovechar los movimientos que generan dinero.

**Banco Comercial**, "Institución financiera de intermediación que recibe fondos en forma de depósito de las personas que poseen excedentes de liquidez, utilizándolos posteriormente para operaciones de préstamo a personas con necesidades de financiación, o para inversiones propias. Presta también servicios de todo tipo relacionados con cualquier actividad realizada en el marco de actuación de un sistema financiero.

Para lo cual son utilizados los **Canales de distribución bancaria** estos son, el circuito o vía a través del cual las Entidades Financieras ofrecen a sus clientes actuales o potenciales sus productos o servicios. Un canal es el medio que utilizan las entidades financieras para prestar sus servicios a los clientes y/o usuarios. Lo invitamos a conocer los canales que hacen su vida más fácil y las recomendaciones de seguridad para que su dinero esté siempre a salvo.

La **Sucursal bancaria** u oficina es la dependencia que establece una entidad financiera como principal canal de distribución de sus productos.

La **Banca electrónica** también llamada banca virtual ó online, es un servicio prestado por las entidades financieras que tiene como misión permitir a sus clientes realizar operaciones y transacciones con sus productos de forma autónoma, independiente, segura y rápida a través de Internet. Entre las transacciones más típicas que se pueden realizar a través de este servicio de banca electrónica están las transferencias, el envío y recepción de ficheros o cuadernos de gestión y la consulta de los movimientos de las cuentas. Todas estas unidades efectúan **Transacciones**, que es el término empleado en el uso de dinero para sufragar el costo de un servicio o bien comprado. Uno de los aspectos que más caracteriza a la transacción es que hay una idea común entre las partes que realizan la operación. Para que se realice es preciso disponer de un capital y que alguien proporcione un servicio o bien que se ajuste a la cantidad reclamada.

Se derivan también y con más elevado número las **Transacciones electrónicas**, El término “transferencia electrónica” (o “giro electrónico”) se refiere a cierto método de transferir los fondos; es decir, la transferencia de fondos que generalmente realiza un usuario por medio de una institución bancaria, por lo que se denomina más propiamente transferencia electrónica. Es por todo esto que debe asumirse con responsabilidad la **Seguridad en las transacciones**. El comercio electrónico necesita garantizar una seguridad técnica y jurídica que impida un anormal funcionamiento del negocio o una desconfianza en el medio utilizado para comerciar. En este sentido se han aportado una serie de soluciones, propuestas por los organismos de normalización, para evitar los posibles peligros u operaciones ilegales a los que puede estar sometida Internet. Básicamente se trataría de garantizar cuatro principios.

1. Principio de autenticidad: que la persona o empresa que dice estar al otro lado de la red es quién dice ser.
2. Principio de integridad: que lo transmitido a través de la red no haya sido modificado.
3. Principio de intimidad: que los datos transmitidos no hayan sido vistos durante el trasiego telemático.
4. Principio de no repudio: que lo transmitido no pueda ser repudiado o rechazado.

El Consumidor financiero, es todo cliente, usuario o cliente potencial de los productos o servicios ofrecidos por las entidades sometidas a la inspección y vigilancia de la Superintendencia Financiera, así como todo aquel que determine la ley o el Gobierno Nacional.

La superintendencia financiera de Colombia el 30 junio del año 2000 realiza una síntesis de las características de intermediación financiera. (El proceso de captación

de fondos se entiende como la acción de atraer el dinero de las personas para realizar la administración de este y generar una utilidad, también conocido en el ámbito económico como fundraising, supone la recolección de recursos económicos por parte de una persona u organización para, posteriormente, destinar dichos fondos reunidos a un objetivo ajeno al lucro personal o empresarial.

Los Métodos de autenticación están en función de lo que se utilizan para la verificación de la identidad del cliente y se dividen en tres categorías:

1. Sistemas basados en algo conocido. Ejemplo, un password.
2. Sistemas basados en algo poseído. Ejemplo, dispositivo usb tipo epass token.
3. Sistemas basados en una característica física: Ejemplo, verificación de voz, de huellas, de patrones oculares.

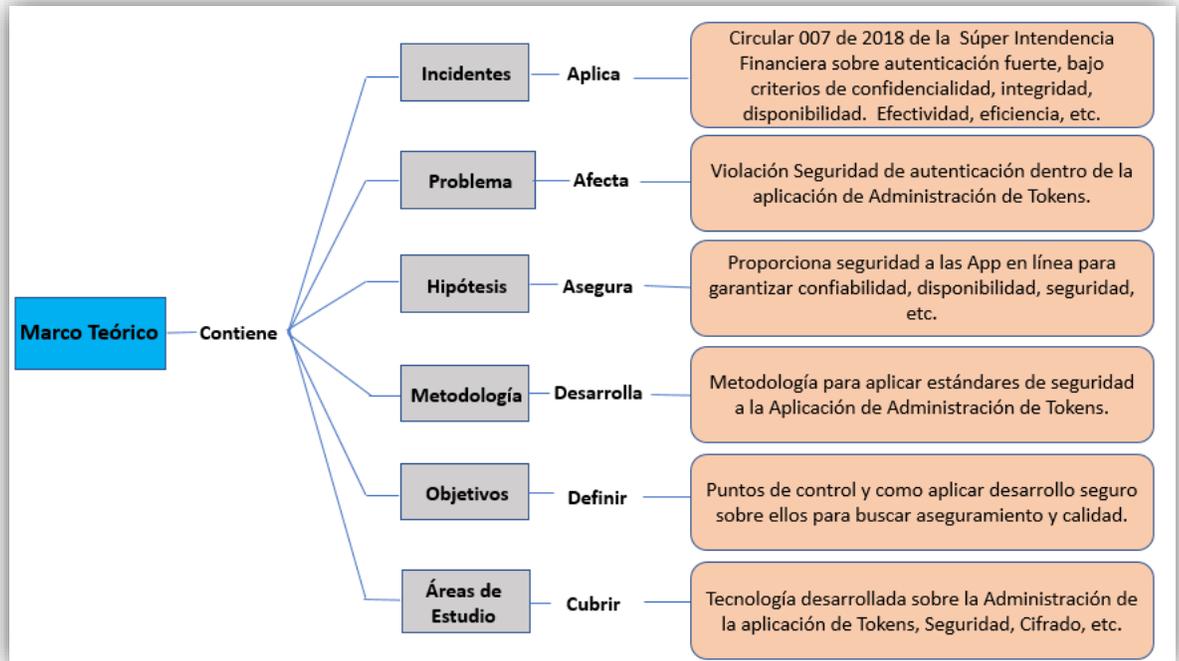
El Token de Seguridad es un doble factor de autenticación que genera códigos de seguridad de # dígitos que cambian constantemente. Estos, nos permiten identificarte como Cliente al momento de realizar tus operaciones en el banco, ofreciéndote un nivel adicional de seguridad. (¶6).

#### **4.2. Marco Teórico.**

Este marco es de gran importancia para las bases y justificación del proyecto; toca temas importantes como la misma definición del problema que lleva a justificar la realización de la protección de la información basados en asumir, enfrentar y mitigar todos los problemas de seguridad en el código de programación efectuado para la construcción de la aplicación, el aprovechamiento de estructuras lógicas, paso de parámetros, uso de códigos de lenguaje nativo para poder hacer ejecución de rutinas que son necesarias en la dinámica de las capturas y recepción de información.

La **¡Error! No se encuentra el origen de la referencia.** muestra la dinámica que contiene cada incidencia dentro del Marco Teórico:

Figura. 4-1. Cuadro Marco Teórico Referencia



Fuente: El Autor. Octubre de 2019

### Transacciones Electrónicas:

A finales de los 90 nació el atractivo de las transacciones por internet, mucho menos que los adelanto a los que se ha llegado; pero ya se hablaba de que el futuro iba a ser totalmente virtualizado, muchísimos menos desplazamientos a las entidades financieras, menos filas en los bancos, multiplicación de las transacciones bancarias, afinamiento en la velocidad alrededor de un clic en un computador o en un celular. Y muchísimas más maravillas a las que ya nos hemos acostumbrado poco a poco. Compras por internet, pagos electrónicos, consultas de diferente índole, entre otras. A lo que da lugar la afinación de la seguridad en las mismas; a través de encriptamiento de la información usando diferentes algoritmos para obtener los hashes de autenticación, nuevas novedades como la biometría, la lectura del iris óptico, y hasta la inserción de chips a nivel de cuerpo humano. Y habrá más novedades que es posible las lleguemos a conocer por la velocidad con la que los electrónicos crean e inventan dispositivos con destinación especial.

Nos ayudaremos de algunos autores para entender los términos especiales de seguridad.

La **Encriptación** o cifrado de información, nos permite ocultar el contenido del mensaje para que sólo el destinatario final pueda leerlo.

La encriptación de datos o cifrado de archivos es un procedimiento mediante el cual los archivos, o cualquier tipo de documento, se vuelven completamente ilegibles gracias a un algoritmo que desordena sus componentes. Así, cualquier persona que no disponga de las claves correctas no podrá acceder a la información que contiene.

De esta forma, según el password, encontramos dos tipos de encriptación de archivos: simétrico o asimétricos.

El sistema de cifrado simétrico es aquel que utiliza una misma clave para cifrar y descifrar, mientras que, en la encriptación de datos asimétrica se usan diferentes claves: una clave pública para cifrar y una de carácter privado para descifrar, de forma que sea imposible deducir la contraseña privada a través de la pública. <sup>(vii7)</sup>

Los **Hash** o funciones de resumen, son algoritmos que consiguen crear a partir de una entrada (ya sea un texto, una contraseña o un archivo, por ejemplo) una salida alfanumérica de longitud normalmente fija que representa un resumen de toda la información que se le ha dado (es decir, a partir de los datos de la entrada crea una cadena que *solo* puede volverse a crear con esos mismos datos).

Estas funciones no tienen el mismo propósito que la criptografía simétrica y asimétrica, tiene varios cometidos, entre ellos está asegurar que no se ha modificado un archivo en una transmisión, hacer ilegible una contraseña o firmar digitalmente un documento. <sup>(viii8)</sup>

Los dispositivos de incrustación o **Ships** son los dispositivos electrónicos que guardan información de quien se acredita para la autenticación. Fueron implantados para mejorar la seguridad que ofrecían dejó de ofrecer las bandas magnéticas.

La nueva forma de identificación física es la **Biometría**; Sincronismo seguro con características individuales de cada persona que lo identifican como único y que permiten dar autenticidad a permitir los controles de accesos. Se basa en mediciones y cálculos corporales de las métricas y características humanas.

En un sistema de Biometría típico, la persona se registra con el sistema cuando una o más de sus características físicas y de conducta es obtenida, procesada por un algoritmo numérico, e introducida en una base de datos. Idealmente, cuando entra,

casi todas sus características concuerdan; entonces cuando alguna otra persona intenta identificarse, no empareja completamente, por lo que el sistema no le permite el acceso. Las tecnologías actuales tienen tasas de acierto que varían ampliamente (desde valores bajos como el 60%, hasta altos como el 99,9%). (ix9)

#### **4.3. Marco Jurídico.**

Nacional,

Circular 007 de 2018, La Circular Externa 007 de 2018 se expidió teniendo en cuenta el auge de la digitalización de los servicios financieros, la mayor interconectividad de los agentes y la masificación en el uso de canales electrónicos, entre otros, y complementa las normas existentes con relación a la administración de los riesgos operativos y la seguridad de la información.

Así, la entidad vigilada deberá informar a los consumidores financieros sobre los incidentes cibernéticos que se hayan presentado y en los que se vieran afectadas la confidencialidad o integridad de su información, al igual que las medidas adoptadas para solucionar la situación.

Dentro de los requerimientos que deberán cumplir las entidades vigiladas en materia de ciberseguridad también está la conformación de una unidad que gestione los riesgos de seguridad de la información y la ciberseguridad.

En este aspecto, es importante la actualización permanente y especializada sobre las nuevas modalidades de ciberataques que pudieran llegar a afectar a la entidad, por lo que deben realizar capacitaciones periódicas para los funcionarios en ciberseguridad.

Adicionalmente, las entidades vigiladas deberán establecer una estrategia de comunicación e información para el envío de reportes a las autoridades que hacen parte del modelo nacional de gestión de incidentes cibernéticos.

Por otro lado, la Circular Externa 007 de 2018 establece que las entidades deberán incluir en el plan de continuidad del negocio la respuesta, recuperación, reanudación de la operación en contingencia y restauración ante la materialización de un ataque cibernético.

Estas nuevas instrucciones empezarán a regir dentro de seis meses y las entidades vigiladas deberán darle cumplimiento en tres etapas: la primera se deberá implementar dentro de los próximos seis meses; la segunda, dentro del año siguiente y la tercera durante los próximos 18 meses.

## **Operaciones con pasarelas de pago.**

La Superfinanciera expidió la Circular Externa 008 de 2018 mediante la cual se establecen mecanismos de protección de la información de los consumidores financieros al realizar operaciones monetarias usando los servicios de las pasarelas de pago.

En la norma se establecen los estándares de seguridad para que estas plataformas puedan prestar sus servicios a través de las entidades vigiladas por la Superfinanciera (bancos y redes de pago).

Cabe señalar que las administradoras de pago o pasarelas de pago no son entidades vigiladas por la Superintendencia Financiera y prestan servicios de aplicación de comercio electrónico para almacenar, procesar y/o transmitir el pago correspondiente a operaciones de venta en línea. <sup>(10)</sup>

**La Ley 1266** sancionada en el 2008 por el Gobierno Nacional, regula el manejo de la información contenida en bases de datos personales, en especial la financiera, crediticia, comercial, de servicios y la proveniente de terceros países.

El artículo 21 de la ley establece un régimen transitorio en que se prevé el beneficio de la exclusión o borrón de la información negativa de los deudores, es decir de la información relativa a la mora en el pago de las cuotas u obligaciones en los siguientes casos:

Primer caso: Si usted tiene obligaciones sobre las cuales existen reportes negativos en las centrales de información, pero a la fecha de entrada en vigencia de la ley usted se encuentra al día en el cumplimiento de todas sus obligaciones, usted puede obtener el beneficio señalado, según se encuentre en una de las siguientes situaciones:

Situación 1: Si a la fecha de entrada en vigencia de la ley su información negativa ha cumplido un año de permanencia en las centrales de información desde la fecha de pago, ésta deberá ser excluida de su historial de manera inmediata.

Situación 2: Si a la fecha de entrada en vigencia de la ley su información negativa aún no ha cumplido un año de permanencia en las centrales de información desde la fecha de pago, la misma deberá ser excluida de su historial cuando complete un

año de permanencia.

Segundo caso: Si usted tiene obligaciones sobre las cuales existen reportes negativos en las centrales de información y a la fecha de vigencia de la ley usted no se encuentra al día en el cumplimiento de todas sus obligaciones, podrá obtener el siguiente beneficio siempre y cuando se ponga al día en todas sus obligaciones durante los seis meses siguientes a la vigencia de la ley:

Beneficio: La información negativa será excluida de su historial apenas trascurra un año de permanencia en las centrales, contado desde la fecha de pago. <sup>(xi11)</sup>

**Ley 1273 de 2009**, "Por medio de la cual se modifica el Código Penal, se crea un nuevo bien jurídico tutelado - denominado "de la protección de la información y de los datos"- y se preservan integralmente los sistemas que utilicen las tecnologías de la información y las comunicaciones, entre otras disposiciones".

El 5 de enero de 2009, el Congreso de la República de Colombia promulgó la Ley 1273 "Por medio del cual se modifica el Código Penal, se crea un nuevo bien jurídico tutelado – denominado "De la Protección de la información y de los datos"- y se preservan integralmente los sistemas que utilicen las tecnologías de la información y las comunicaciones, entre otras disposiciones".

Dicha ley tipificó como delitos una serie de conductas relacionadas con el manejo de datos personales, por lo que es de gran importancia que las empresas se blinden jurídicamente para evitar incurrir en alguno de estos tipos penales. <sup>(xii12)</sup>

#### **4.4. Marco Geográfico.**

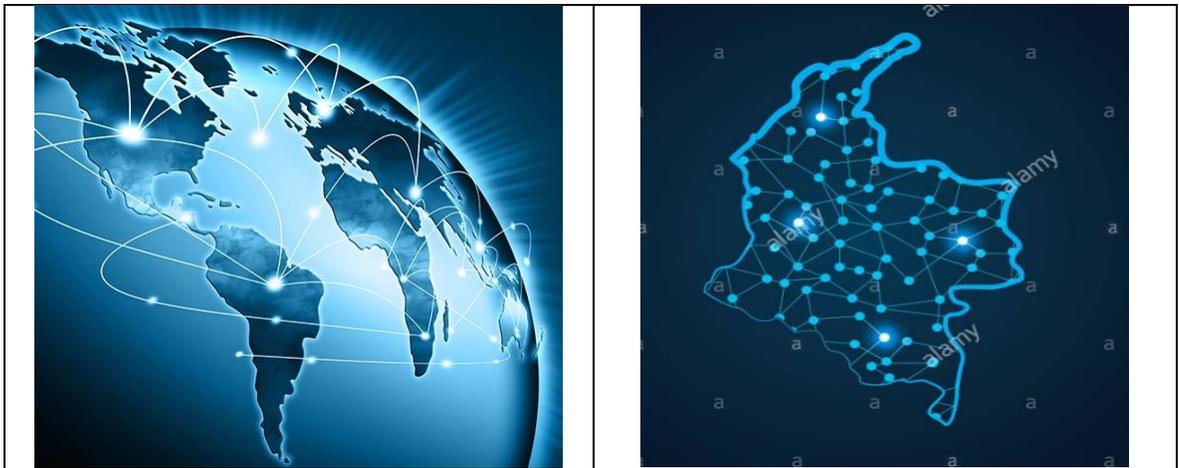
Este proyecto tiene un impacto nacional e internacional gracias a las facilidades que proporciona conectarse a una red de internet. Ya sea a través de la Banca Virtual o la telefonía móvil. De igual manera debe usarse el Token como medio de autenticación en las aplicaciones en internet.

El desarrollo del proyecto se lleva a cabo en las instalaciones de una entidad financiera que usa estos elementos de autenticación. Es allí donde se almacenan los programas fuentes de la aplicación que también son de trato confidencial, ya que toda entidad financiera es celosa de sus activos además de que es obligada a protegerlos de acuerdo con las medidas de la Superfinanciera de Colombia.

La entidad financiera opera en todo el país con diferentes sucursales físicas que le permiten llegar a un gran número de clientes y que facilita realizar las operaciones monetarias con gran facilidad. Pero los canales a través de los cuales se llega con gran facilidad al cliente son los electrónicos que aseguran la información a través de los tokens de autenticación y acceso.

Todos estos conceptos hacen parte de la globalización puesto que ayudan a realizar los movimientos desde cualquier parte del mundo. Este proceso es económico, político, tecnológico, social y cultural que abarca a todos los países y culturas del mundo entero. Facilitan la unión de mercados, culturas haciendo dinámico su intercambio.

Figura. 4-2. Globalización – Colombia



Fuente: Globalización. Revista contable. <https://sp.depositphotos.com/220580808/stock-illustration-global-logistics-network-concept-communications.html>

En Colombia es entendido como comunicación el intercambio de todo tipo de información en cuanto a imágenes, señales, correos, textos, transferencias de dinero, consultas de saldos y movimientos ya en todo lo relacionado con clientes de los bancos. Estas comunicaciones cubren la mayoría del territorio colombiano y dependen del desarrollo de cada región.

#### 4.5. Marco Demográfico.

Proyecto que beneficia a una población que es mayor de edad, es decir supera los 18 años. Hombres o mujeres de toda clase social, y también a los menores que tienen autorización expresa para el manejo de productos financieros. En Colombia según proyecciones del DANE en 2018 cerca de 40 millones de personas. Pero los bancos y entidades financieras tienen su selección de clientes por los filtros a los

que sujetan a los aspirantes a poseer un producto financiero.

En 2015 la Asobancaria dijo que más o menos el 75% de la población tenían un producto bancario, para el 2018 aspiraban que este porcentaje subiera al 85%.

La inclusión financiera es el proceso de integración de los servicios financieros a las actividades económicas cotidianas de la población, lo cual puede contribuir al crecimiento económico en la medida en que permita reducir los costos de financiación y transacción, y ofrezca un manejo seguro y eficiente de los recursos. La medición de la inclusión financiera abarca diferentes dimensiones del acceso y uso de productos financieros. En esta edición se abordan los aspectos regionales y locales de la inclusión financiera y el acceso a servicios financieros en Colombia. Para este fin, se emplea información sobre cartera de crédito, captaciones, provisiones y número de entidades a nivel de municipio/establecimiento de crédito publicada trimestralmente por la Superintendencia Financiera de Colombia con corte a diciembre de 2017. Esta información se complementa con las proyecciones de población, las estimaciones de valor agregado a nivel municipal publicadas por el DANE e información sobre el número de homicidios por municipio publicada por Medicina Legal con corte a diciembre de 2017, con el fin de servir como variables de escala y variables asociadas con la inclusión financiera a nivel local. (xiii13)

## 5. METODOLOGÍA

### 5.1. Fases del Proyecto en General.

Con el fin de dar cumplimiento al objetivo general se estableció un desarrollo en 4 fases como se evidencia en la **¡Error! No se encuentra el origen de la referencia..** El cual plantea en la fase 1 la generación del ambiente del proyecto, en la fase 2 se plantea la dinámica del trabajo, en la fase 3 el análisis de resultados y finalmente en la fase 4 el desarrollo del informe final.

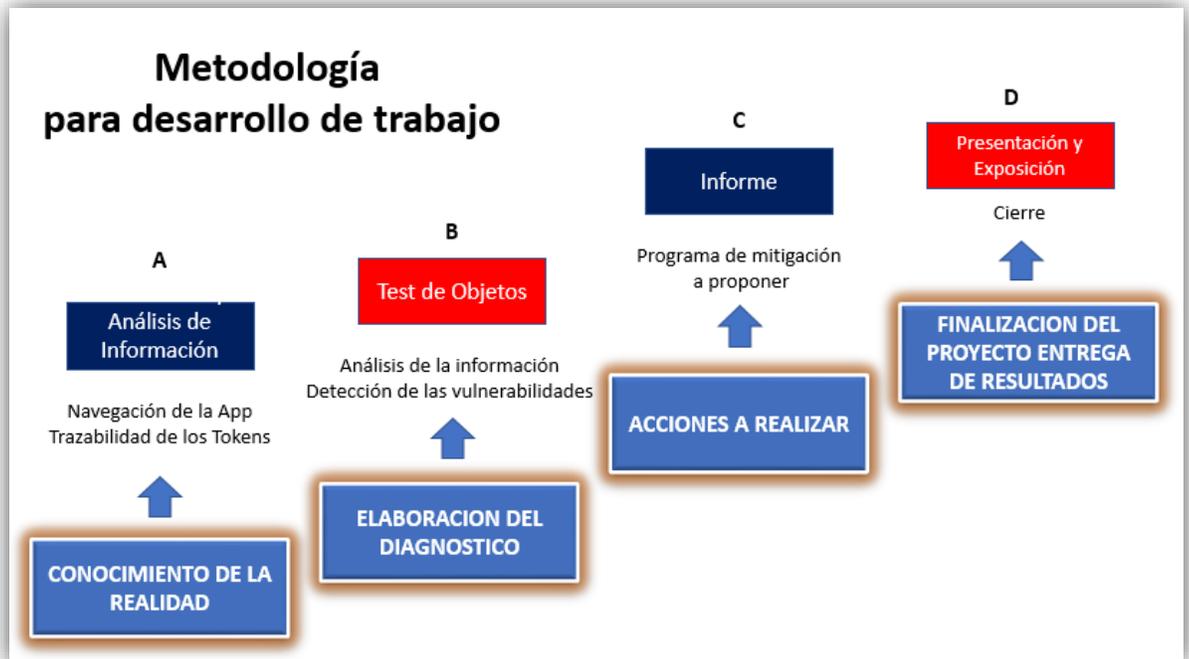
Figura. 5-1. Metodología del proyecto



Fuente: El autor

La generación del ambiente del proyecto se dio a partir de la elaboración del diagnóstico, evaluando los objetos presentes en el código de la aplicación web, en donde el escaneo detectó las falencias existentes y muestra las posibilidades de puertas abiertas para que los maliciosos puedan aprovecharse de ellas.

Figura. 5-2. Metodología para desarrollo de Trabajo de Investigación.



Fuente: Autor, septiembre 2019

El informe final entrega resultados específicos de todos los puntos a tener en cuenta en el desarrollo programático y que deben ajustarse para entrar en el camino de los Desarrollos Seguros mitigando todos los conflictos y huecos de seguridad de la aplicación.

## 5.2. Instrumentos o herramientas utilizadas.

**Modelo GeneXus:** Se toma el Modelo GeneXus de la Aplicación de Administración de Tokens, y se aplica el scaneo de objetos.

**DII Security Scanner:** Herramienta gratuita de adición al Desarrollador GeneXus (estancia de trabajo de GeneXus) que sirve para establecer todas las vulnerabilidades que tiene los códigos GeneXus programados para una generación de código java en web.

## 5.3. Población y Muestra.

Para el tema se toma todo el universo de la aplicación y todas las posibilidades que

hay en las vulnerabilidades a encontrar. Aquí no hay una población de clientes que usen la aplicación y que podrían facilitar la búsqueda de código débil que pudiera exponer fácilmente el ingreso de ataques a través de la aplicación.

#### 5.4. Modelo de evaluación y enmarcación.

OWASP es una organización conocida a nivel mundial que agrupa una gran cantidad de profesionales de la seguridad informática para generar conocimiento alrededor de la seguridad web, cuenta con una cantidad de proyectos impresionante, proyectos de innovación en seguridad en la que participan muchas personas y que son un referente en la industria.

Uno de los proyectos más populares de **OWASP** es el famoso OWASP TOP TEN (OWASP TOP 10), un listado de los problemas más comunes encontrados en las aplicaciones web organizados por criticidades súper respetado y es un estándar de facto en las auditorías web. El Top Ten es traducido a múltiples idiomas y estas traducciones se llevan a cabo por los miembros que conforman la Open Web Application Security Project (OWASP) que se ha encargado de traducir el Top Ten a Español, la versión más actual del top es la 2017 que se encuentra en su Release Candidate 2 (RC 2) o versión candidata a ser la definitiva, todavía pero lo más seguro es que las posiciones dentro del top no cambien, por eso hemos generado esta imagen comparativa para la versión 2013 y 2017 en español. (xiv14)

Figura. 5-3. Seguridad Informática. OWASP TOP 10 2013 vs OWASP TOP 10 2017

OWASP TOP 10 - 2013 (MEJOR)	OWASP TOP 10 - 2017 (NUEVO)
A1 - Inyección	A1 - Inyección
A2 - Pérdida de Autenticación y Gestión de Sesiones	A2 - Pérdida de Autenticación y Gestión de Sesiones
A3 - Secuencia de Comandos en Sitios Cruzados (XSS)	A3 - Exposición de Datos Sensibles
A4 - Referencia Directa Insegura a Objetos	A4 - Entidad externa XML (XXE)
A5 - Configuración de Seguridad Incorrecta	A5 - Control de acceso roto (fusionado)
A6 - Exposición de Datos Sensibles	A6 - Configuración de Seguridad Incorrecta
A7 - Ausencia de Control de Acceso a las Funciones	A7 - Secuencia de Comandos en Sitios Cruzados (XSS)
A8 - Falsificación de Peticiones en Sitios Cruzados (CSRF)	A8 - Deserialización Insegura (Nuevo)
A9 - Uso de Componentes con Vulnerabilidades Conocidas	A9 - Uso de Componentes con Vulnerabilidades Conocidas
A10 - Redirecciones y reenvíos no validados	A10 - Insuficiente registro y monitoreo (Nuevo)

Fuente: <https://www.dragonjar.org/owasp-top-ten-project-en-espanol.xhtml>

## **5.5. Alcance y limitaciones**

### **Alcances**

A partir de la evaluación y análisis de los procesos evidenciar las fallas en los desarrollos informáticos de la Aplicación de Administración de Tokens y recomendar las posibles acciones correctivas que mitiguen los riesgos. Las recomendaciones serán entregadas el banco en un documento formal para su respectivo tratamiento.

### **Limitaciones.**

El proyecto se centra en detectar las fallas en la aplicación web de administración de tokens para el banco y sugerir las posibles correcciones, pero no contempla la implementación de las mismas.

## 6. EJECUCION DE LA METODOLOGIA

Para dar cumplimiento a la metodología propuesta es necesario que se tengan claro algunos conceptos y procedimientos desarrollados como los que se explican a continuación.

### 6.1. Aplicación Administración de Tokens.

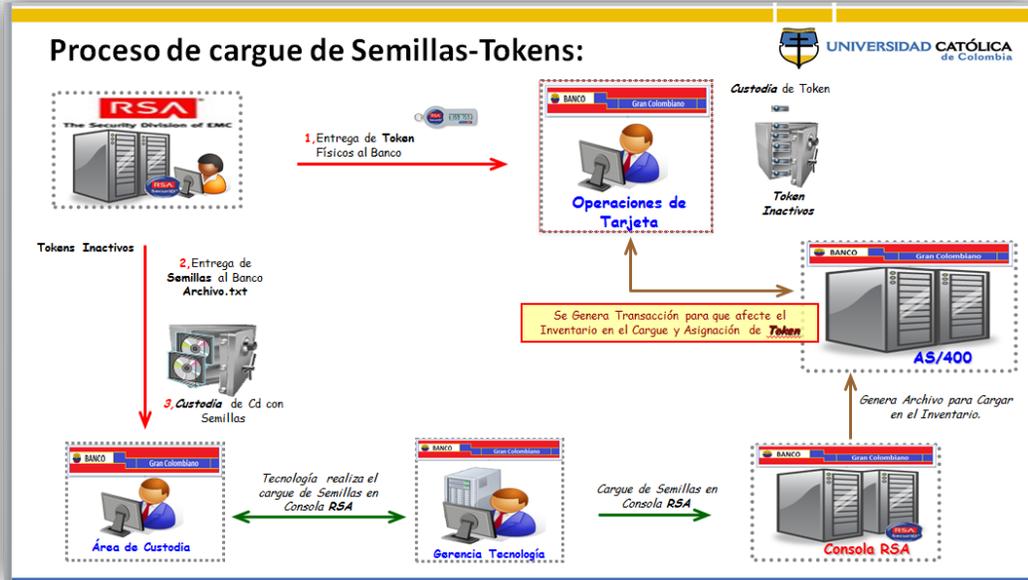
La aplicación de administración de Tokens es un aplicativo paralelo a la Banca Virtual y a la Banca Móvil, gracias a su administración y uso se proporciona seguridad especial a las autenticaciones de los clientes para dar ingreso a la gestión personal de sus cuentas; consulta de saldos, consulta de productos financieros, pago de servicios públicos, pago de obligaciones financieras, manejo de inversiones, vinculación de cuentas de otras entidades financieras para intercambio de servicios, etc. En la banca empresarial, pago de nóminas, manejo de grupos financieros, dobles autorizaciones y autenticaciones, subida de archivos planos para pagos a otras entidades, entre otros.

Esta aplicación consta de varios módulos que son: Acceso seguro, Login por usuarios de inscripción en AS400, parametrización, proceso de cargue de semillas tanto a la consola de administración como al inventario, gestión de administración de Tokens, manejo de inventario por Tokens físicos, Tokens de Software, y de 2 proveedores RSA y VASCO. Aunque de los primeros ya se está agotando el inventario y finalmente se van a dejar solo los del proveedor Vasco. Otros módulos son las autorizaciones realizadas por personal administrativo superior, el transporte a través del proveedor Domesa, entrega de Tokens a los clientes, activación de los mismos, reposiciones, y el manejo de informes y reportes.

#### 6.1.1. Proceso de cargue de semillas.

Como se observa en la **¡Error! No se encuentra el origen de la referencia.**, en este proceso se hace el cargue de las semillas al inventario de Tokens, su almacenamiento se hace en la DB de AS400 y quedan ubicados en la tabla: BDOD14 de la biblioteca WebCorDat.

*Figura. 6-1. Proceso de cargue de Semillas*



Fuente: IT Banco Industrial Colombiano. (Banco ficticio de Prueba)

Opera desde que se inyectan las Semillas en el Inventario y en la consola, son sincronizadas y se hace la asignación a los clientes entregándoles los Tokens y luego inicializárselos a estado Activo.

El gráfico anterior muestra a través de las flechas el camino del Token al inventario de semillas y al inventario de administración, cuya alimentación inicia los procesos de vida de los mismos.

### 6.1.2. Autenticación del Cliente en Banca Virtual.

Es el proceso a través del cual, el cliente (independientemente si son personas naturales o jurídicas) se autentica de manera fuerte en la Banca Virtual con la ayuda personal del uso del Token. En la **¡Error! No se encuentra el origen de la referencia.** se evidencia el diagrama de flujo del proceso de autenticación en la banca.

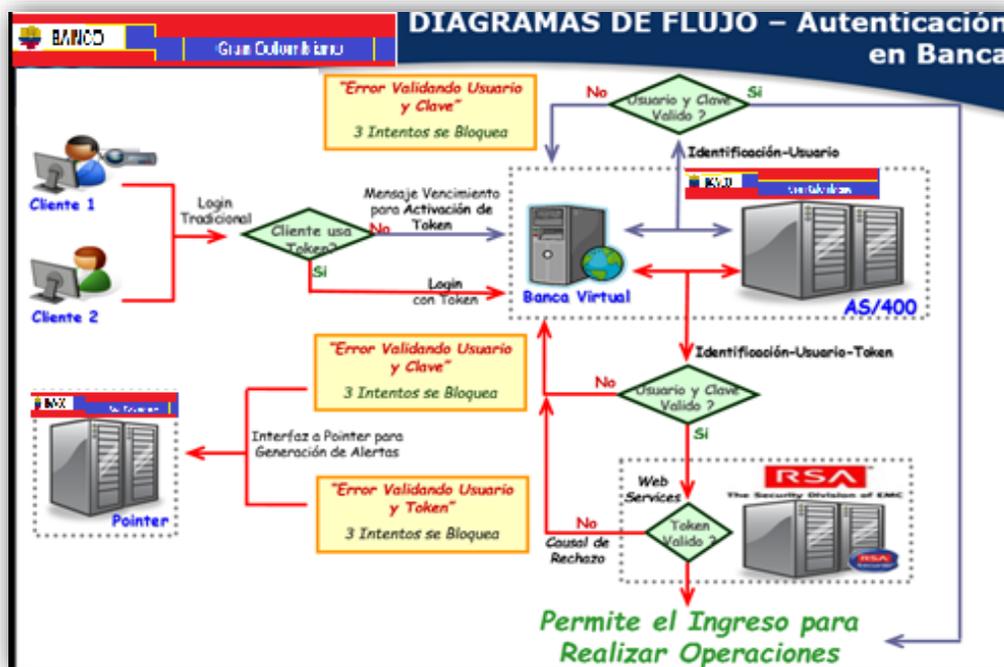
Figura. 6-2. Autenticación en Banca.

Fuente: Biblioteca Informática Banco Industrial Colombiano.

### 6.1.3. Proceso de ingreso de semillas al inventario.

Usando esta pantalla se da ingreso a cargar los Tokens al inventario, con base en un archivo plano que proporciona la consola de Tokens en el momento de cargar las semillas para ser expuestas a autenticación.

El protocolo de cargue de semillas al inventario debe ser realizado por la jefe del área de tarjetas desde su usuario de gestión para la aplicación, el cual al ser usado registra un log de ejecución en el mismo archivo. Este ingreso se hace por una única vez, cada vez que se ingresan semillas nuevas al inventario. En caso de ella no



esté, lo debe hacer su asistente, la subdirectora con su usuario autorizador. Después de éste cargue, el área de gestión informática hace la verificación tanto del inventario como de la contabilidad.

En la **¡Error! No se encuentra el origen de la referencia.** se evidencia la forma de autenticación de los administradores y gestores de la aplicación.

Figura. 6-3. Autenticación Aplicación



Fuente: El Autor

Una vez se ha realizado el acceso al sistema se encuentra la relación de los tokens con los usuarios del sistema como se evidencia en la **¡Error! No se encuentra el origen de la referencia..**

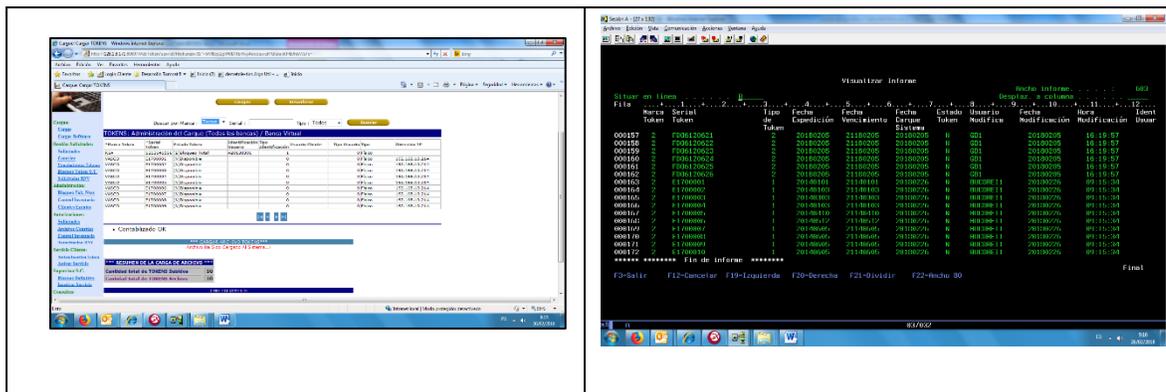
Figura. 6-4. Sistema Cargue Semillas

*Marca Token	*Serial Token	Estado Token	Identificación Usuario	Tipo Identificación	Usuario Cliente	Tipo Usuario	Tipo	Dirección IP
RSA	1213141554	(B)Bloqueo Total	A0138301	1			0 Físico	
VASCO	FD06112494	(A)Activo	41397929	1	41397929	1	Software	190.64.72.210
VASCO	FD06112497	(A)Activo	39960526	1	39960526	1	Software	190.64.72.210
VASCO	FD06112486	(T)Bloqueo x Servicio	17127053	1	17127053	1	Software	172.24.7.25
VASCO	FD06112499	(A)Activo	19063707	1	19063707	1	Software	180.253.35.152
VASCO	FD06112500	(A)Activo	32873213	1	32873213	1	Software	190.64.72.210
VASCO	FD06112501	(A)Activo	199749	3	199749	3	Software	172.24.7.25
VASCO	FD06112502	(I)Inactivo	32085864	1	32085864	1	Software	172.24.7.25
VASCO	FD06112503	(A)Activo	17184719	1	17184719	1	Software	180.253.35.152
VASCO	FD06112504	(A)Activo	30096068	1	30096068	1	Software	172.24.7.25

Fuente: Biblioteca Informática Banco Industrial Colombiano.

Para mayor comprensión del proceso, se realiza un ejemplo de 10 semillas el cual se evidencia en la **¡Error! No se encuentra el origen de la referencia..**

Figura. 6-5. Ejemplo de cargue de semillas



Fuente: Biblioteca Informática Banco Industrial Colombiano.

#### 6.1.4. Creación de la Solicitud de Tokens por parte del Cliente.

Esta solicitud generalmente es de creación automática en el momento en que los clientes crean su Banca Virtual por primera vez, o cuando se hace la reposición de Tokens por daño o por fecha de vencimiento.

#### 6.1.5. Asignación de Tokens a la Solicitud radicada por el proceso automático.

Cuando se ha creado la Solicitud de Tokens por medio del ingreso del cliente a la Banca Virtual, esta llega al área de Tarjetas en donde es regularizada por la encargada de gestión de las asignaciones de Tokens; el proceso comienza viendo en la bandeja de Solicitudes las que estén en estado P, 'pendiente'. Las solicitudes pueden venir de personas o de empresas.

El Banco para definir la veracidad de un cliente y decidir hacer la asignación de Tokens a sus solicitudes ya ha hecho verificaciones de evidente, las cuales son procesos ya definidos por Servicio al Cliente que ratifican que quien hace la solicitud primero ha sido autenticado tanto en el área de cuentas existentes como en las áreas de mesa de ayuda. Por tanto al hacer la asignación de tokens ya es seguro de que quien lo va a poseer es quien dice ser ante el Banco; por otra parte el token se entrega al cliente inactivo y al final cuando está en su poder, el cliente sigue las instrucciones de un instructivo mensaje que se le entrega junto con el token; luego el cliente debe llamar a Servicio al Cliente, donde se le hace nuevamente un evidente para proceder a realizar la activación del dispositivo.

Figura. 6-6. Pantalla de Asignación individual de Tokens a la Solicitud.

**Servicio Cliente:**

[Activa/Inactiva Token](#)

[Activar Servicio](#)

**Supervisor S.C.**

[Bloqueo Definitivo](#)

[Inactivar Servicio](#)

**Consultas:**

[Solicitudes](#)

[Novedades Tokens](#)

[Solicitudes RNY](#)

[Cerrar Sesión](#)

\*\*\* ASIGNACION TOKENS A CLIENTES \*\*\*

**\*Número SOLICITUD**      **109**

CLIENTE para asignar el TOKEN: 32016308 PABLO JAVIER 375560 BENGOCHEA MORENA

Tipo Identificación: 1 - CEDULA DE CIUDADANIA

Fecha Solicitud: 25/10/16

---

Administrador 1: PABLO JAVIER 375560 BENGOCHEA MORENA

Administrador 2:

---

Cantidad Total Tokens: 1      Tokens Asignados: 0

Token x Asignar: 1

---

Digite TOKEN a asignar:     Proveedor    

Tipo:

---

Asignación automática:     Proveedor        

---

**Asignación de Token a la Solicitud**

Marca Token	Serial Token	Fecha Asignación	Tipo	Marca Nuevo Token	Nuevo Token	Cambiar	Desasignar
GNB SUDAMERIS							

Fuente: Datos simulados de un Cliente a ser atendido.

Otros proceso que colaboran con la operatividad de la aplicación son: Trazabilidad del camino que lleva el Tokens desde su asignación, aprobación, armado del paquete de envío custodiado al transportador. Envío por parte del transportador Domesa al cliente final que dará uso al Token para sus autenticaciones, ativaciones de tokens, inactivaciones de tokens, Proceso de reposición en cadena, informes y reportes del aplicativo. En fin otra serie de objetos y procesos que completan la cadena de valor de la administración de Tokens bancarios. Para los cuales se hará un examen riguroso de la seguridad y efectividad de objetos en la aplicación.

## 6.2. Herramienta Base de Desarrollo del Aplicativo.



**GENEXUS:** Es una herramienta de cuarta generación para el desarrollo de aplicaciones generadas en cualquier lenguaje, con cualquier base de datos y en diferentes plataformas de ejecución de gran uso en la Banca principalmente en la

administración de Tokens

Es una herramienta de desarrollo incremental basada en prototipos que paulatinamente se van convirtiendo en una gran aplicación. La herramienta es principalmente para desarrollo empresarial de aplicaciones web, aplicaciones para dispositivos móviles y también en términos básicos de cliente servidor y desarrollos locales.

El entorno de desarrollo contiene un módulo de normalización que crea y mantiene una estructura óptima de base de datos.

A través de ella se pueden llegar a generar lenguajes de programación como: C#, .Net, java, Cobol, RPG, Ruby, Visual Basic, para Android, entre otros.

Los Sistemas de Bases de Datos soportados son: SQL Sever, Oracle, DB2, Informix, PostgreSQL, MySQL.

GeneXus es una herramienta desarrollada por la empresa Uruguaya ARTech Consultores SRL.

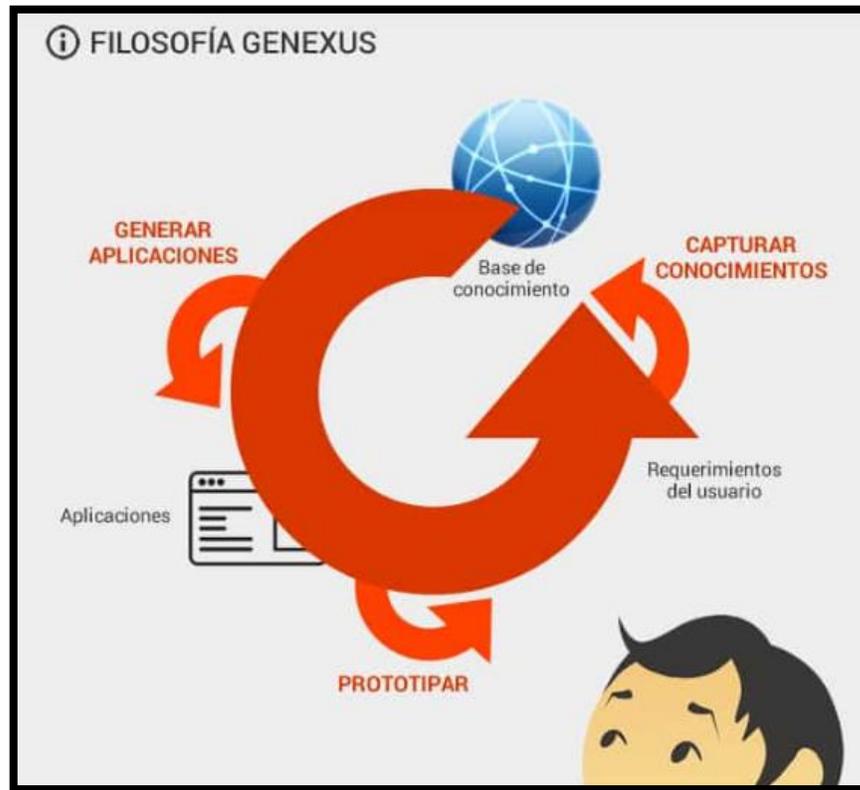
Durante los últimos años ha tenido una gran evolución llegando a la última versión codificada como Gx 16 , y antes están, la Gx 15, Gx Evoluxion, Gx 90, Gx 80, Gx 7.5, Gx 6.1, entre otras, cuya información se encuentra disponible en la página [www.genexus.com](http://www.genexus.com)

### **6.2.1. Filosofía de GeneXus.**

Es una Herramienta primeramente inteligente para el desarrollo de aplicaciones y sistemas de información que permite crear, desarrollar y mantener en forma automática programas, bases de datos y aplicaciones de misión crítica en multilenguajes para diferentes plataformas. Los desarrollos son evolutivos de acuerdo a las condiciones cambiantes del negocio y el mercado.

El ciclo de la gráfica es continuo; cuando se genera nuevo conocimiento, los desarrollos generados dejan aplicar cambios para seguir evolucionando y para seguir cubriendo más áreas de negocio de tal manera que puede llegar a convertirse en un Core que abarque las necesidades globales de las empresas.

*Figura. 6-7. Filosofía de trabajo con GeneXus.*



Fuente: Genexus, tomado de <https://www.itsitio.com/mx/el-desarrollador-debe-evolucionar-genexus/>

### 6.2.2. Desarrollo de una aplicación con la herramienta GeneXus.

El desarrollo de una aplicación implica tareas de análisis, diseño e implementación. GeneXus libera a los desarrollos de tareas automatizables como la creación de la Base de datos, su mantenimiento y normalización; dando prioridad a las tareas difíciles de diseño y creación que no son automatizables.

Desafortunadamente esta aplicación en particular desarrollada en 2010 no se hizo bajo estos lineamientos de desarrollo seguro y menos con el uso de esta herramienta; en esos momentos el tema no era de moda como lo es hoy en día ni tampoco se habían desarrollado requisitos explícitos de seguridad en la misma.

GeneXus emplea una metodología que se enfoca de forma diferente a las demás metodologías comúnmente utilizadas.

El tema arranca con el conocimiento de la realidad, y cada uno de los involucrados conoce partes del gran objetivo a ser resuelto. Para lo cual se debe extraer y plasmar el conocimiento de los usuarios, las reglas que lo gobiernan y los cálculos

que deben realizarse, así como la definición de una parametrización acorde al negocio.

Con la realidad vivida por el ser humano y la actividad que desarrolla dentro de un todo se comienza a plasmar el conocimiento hasta moldear poco a poco la realidad obteniendo una solución tecnológica acorde con la problemática que plantean los actores involucrados.

### 6.2.3. Objetos GeneXus.

Luego de moldeada la realidad, el paso siguiente es empezar a utilizar los objetos que proporciona GeneXus para dar solución tecnológica al problema. Los principales objetos ofrecidos para tal fin por GeneXus son:

**Transacciones:** Primeros objetos a ser diseñados, estos crean las bases de datos paralelamente a su mantenimiento en creación, modificación y borrado de registros. Estos objetos traen implícita la integridad referencial. Dada por el manejo que se les dan a los datos, garantizando la relación entre las tablas durante las operaciones de actualizado y borrado.

**Procedimientos:** Estos objetos escritos línea a línea permiten recuperar la información de la base de datos; también permiten la actualización de la base de datos, son los únicos objetos que pueden dejar de lado la integridad referencial.

**Web Panel / Work Panel:** son objetos que permiten administrar los datos de las bases de datos, permiten mostrar los datos en forma organizada para efectuar las operaciones de mando (Crear, modificar y borrar) hacia las Bases de datos.

**Reportes:** Objetos que permiten mostrar informes de las bases de datos ya sea impresos o por pantalla.

**Data Providers:** facilita el cargue de datos en forma jerárquica para intercambiar información entre objetos de la misma aplicación o de otras aplicaciones.

**Data Views:** Permite vincular a las aplicaciones tablas que no están en su contorno o inclusive en entornos de plataformas diferentes.

### 6.2.4. Desarrollo Incremental.

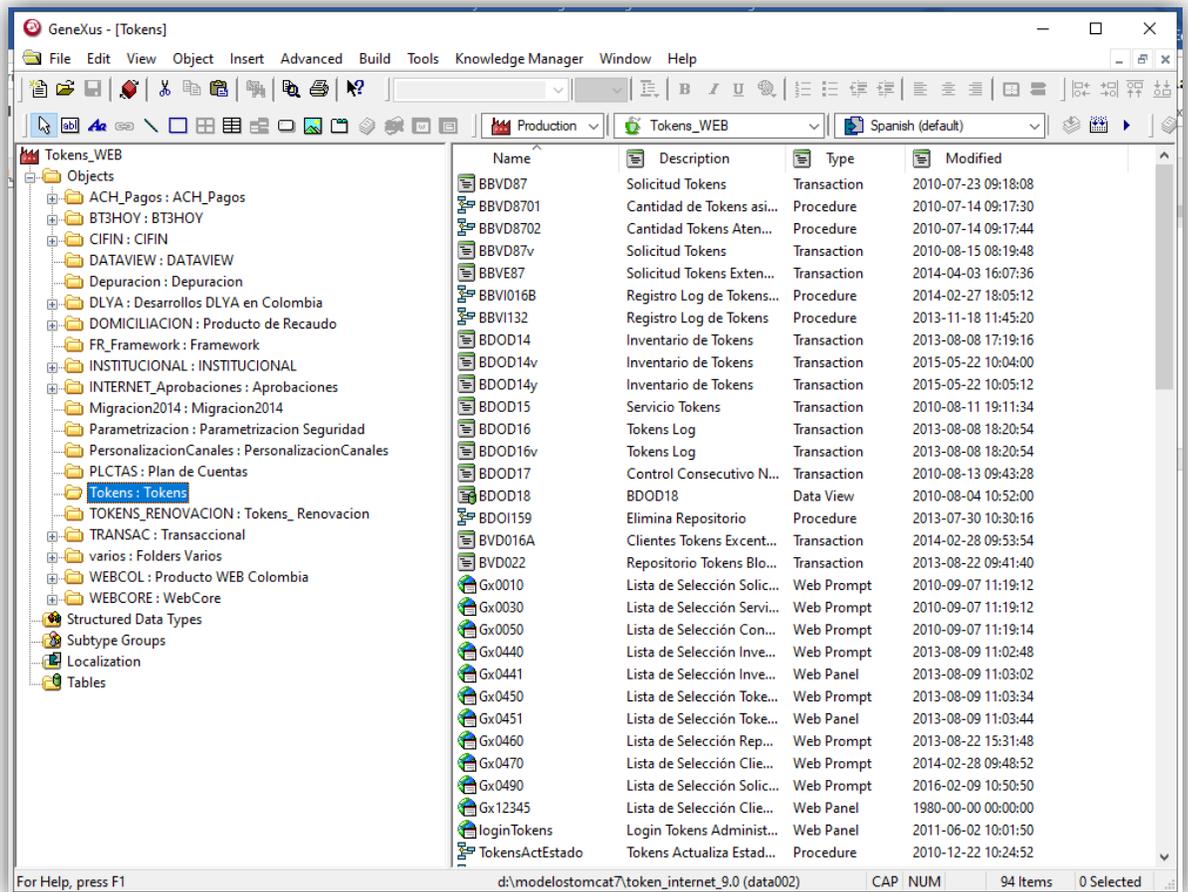
Este es el aporte y desarrollo sucesivo de adiciones a las aplicaciones de tal manera que se creen soluciones más completas y de mayor cubrimiento cada vez.

Esta filosofía ayuda enriquece los desarrollos tecnológicos y los hace más unificados para mostrar que en el futuro las aplicaciones no pierden su dominio general como base de conocimientos administrativa.

### 6.3. Modelo GeneXus de Aplicación Administración de Tokens.

El Modelo GeneXus es la estancia o entorno de Desarrollo de los aplicativos GeneXus a través del cual se producen todos los objetos que realizan la Administración de los Tokens en la entidad financiera. El Modelo está organizado por carpetas de desarrollo que permiten administrar la aplicación para hacer un desarrollo sano y estructurado, como se evidencia en la **¡Error! No se encuentra el origen de la referencia..**

Figura. 6-8. Navegador de Windows



Fuente: Autor

Esta versión de desarrollo es la GeneXus 9.0, 4 versiones anteriores a la versión más actualizada, pero de igual manera ya produce objetos de la aplicación web. Todo el modelo contiene el desarrollo realizado para la administración de Tokens, y muchas otras carpetas que ayudan en el tema; parametrización, partes del Core de la entidad, migraciones, conexiones con otras plataformas, y otros aplicativos, etc.

## 6.4. GeneXus y la Versión para Evaluar la Seguridad.

La aplicación formalmente se encuentra desarrollada en la versión 9.0 de GeneXus. Para realizar la evaluación y escaneo de todos los objetos es necesario migrar el Modelo GeneXus 9.0 a GeneXus Evolution 3 que es donde se puede montar la dll de scanner para poder hacer el análisis de la seguridad de los objetos producidos en el entorno web que al ser corridos pueden facilitar la intromisión de ataques maliciosos por parte de terceros inescrupulosos.

La forma de hacer la migración es preparar el entorno del desarrollador instalando GeneXus Evolution 3 (estancia de trabajo de GeneXus) y haciendo la exportación de la GeneXus 9.0 a la GeneXus Evolution 3.

### 6.4.1. Migración de la Aplicación a La Gx Evolution 3.

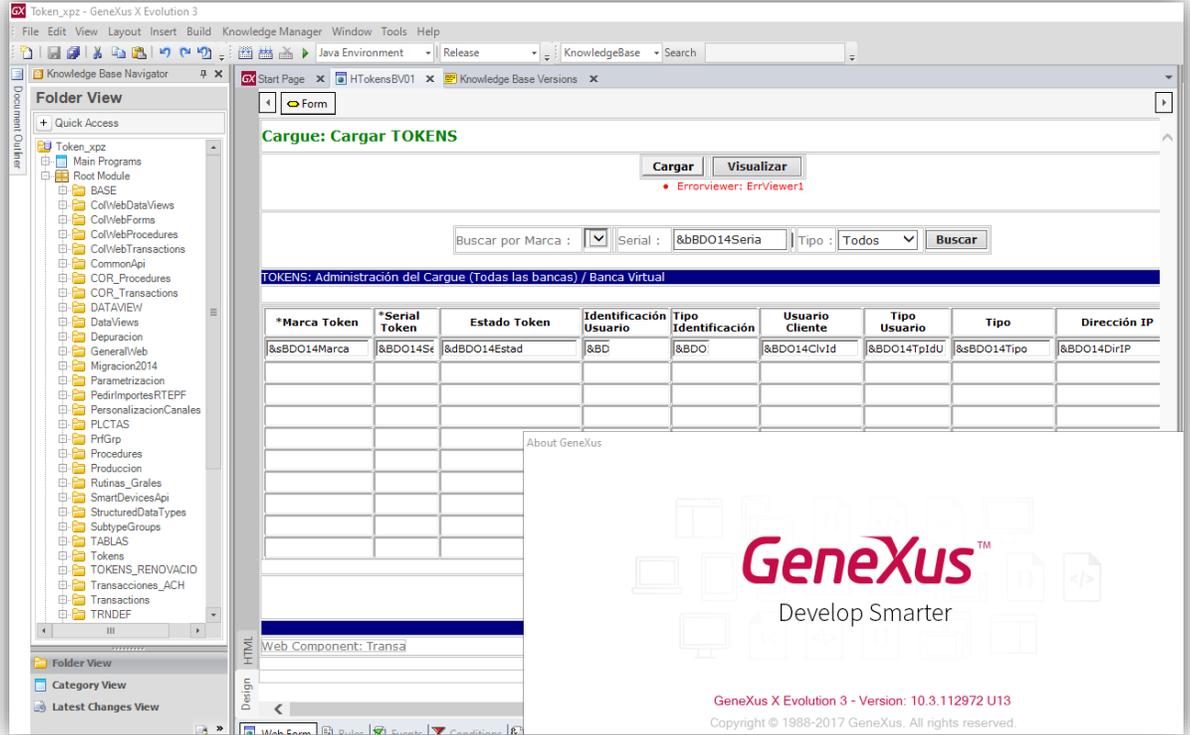
Para la migración de la aplicación, se baja de la página de Artech GeneXus, productos por versiones (Older versions) <https://www.genexus.com/en/products/genexus/versions> y de la versión de GeneXus Evolution 3 (Conseguir el producto, inscribirse para poder obtener la versión) : <https://www.genexus.com/es/productos/genexus/versiones/genexus-x-evolution-3-contacto>

Posteriormente se instala y se licencia para poder hacer la migración.

Y finalmente se abre el desarrollador (estancia de trabajo de GeneXus) de la aplicación Gx Evo 3 y desde la misma app se abre el modelo de Tokens a ser migrado.

En la **¡Error! No se encuentra el origen de la referencia.** se evidencia el Modelo ya migrado:

*Figura. 6-9. Modelo Migrado*



Fuente: Autor

Al ser abierto el Modelo de la aplicación Gx 9.0 en la nueva versión Evolution 3, el desarrollador hace la conversión objeto por objeto hasta terminar todo el modelo. Luego arroja una serie de inconsistencias que deben resolverse para la nueva versión, para el caso no hubo ningún problema y fue migrada normalmente.

Se migra para poder hacer el análisis de seguridad y vulnerabilidades que puede poseer el Modelo en cada uno de sus objetos.

## 6.5. GeneXus Consulting. Mejores Prácticas de Desarrollo.

Se describen a continuación algunas posibles puertas abiertas que se suelen dejar en las aplicaciones por desconocimiento y que pueden facilitar las vulnerabilidades del software.

### 6.5.1. GeneXus Consulting.

Es una empresa de consultoría que desde el conocimiento de los negocios de los

emprendedores de todo el mundo, genera soluciones de misión crítica con la herramienta de desarrollo GeneXus.

### 6.5.2. Mejores prácticas.

Algunas de las mejores prácticas que deben adoptar los programadores de los desarrollos de aplicaciones GeneXus son:

**XML Reader:** Formato de texto sencillo para intercambio de datos muy usado por los navegadores Google Chrome, Yahoo, Mozilla Firefox, entre otros. para intercambiar información entre ellos y los clientes. El cómo utilizar el objeto es responsabilidad del programador de aplicaciones. Y debe comprobar que el objeto no contenga código malicioso.

**HTTPRESPONSE:** Cuando se utiliza este objeto, GeneXus no incluye los headers habituales en las respuestas. Es responsabilidad del programador incluirlos por tanto deberá hacerlo.

**Retornar contenido estático:** Es rutinario generar contenido estático para requerimientos de Excel, pdfs, imágenes almacenadas. Todo lo que genera rutas por requerimientos http. Links(). Todo esto es conocido como Referencia directa a insegura a objetos y se ubica como el cuarto riesgo contemplado por OWASP del top 10. En vez de utilizar un link() el programador debe usar sentencias GeneXus con parámetros encriptados en donde puede amarrar un GUID como identificador y el número de la websession.

## 6.6. Security Scanner. DLL.

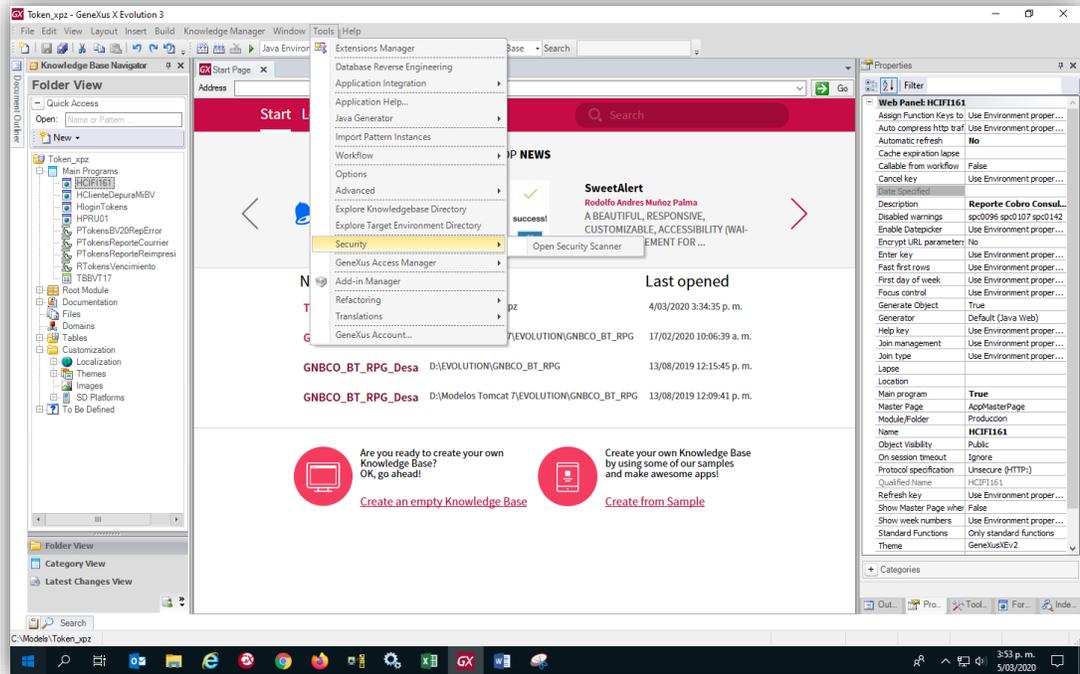
Esta extensión del Security Scanner hace la verificación de objetos dentro de la Base de Conocimiento de GeneXus versiones superiores a Gx Evolution 3 en busca de grandes problemas de seguridad confrontados con los 10 riesgos de seguridad de OWASP.

### 6.6.1. Proceso de Instalación.

Se espera que en la aplicación superior a Gx Evo 3 para desarrollar productos GeneXus aparezca la siguiente opción: en el menú de Tools, línea Security, Open Security Scanner. Que en la aplicación original no aparece. Luego el dll que se

monta adicional esta funcionalidad.

Figura. 6-10. Imagen desarrollador Gx Evolution 3



Fuente: El autor

Para ello, se deben seguir los siguientes pasos:

1.- Obtenga de la red bajando la dll: **SecurityScanner.dll**.

Desde el Marketplace :

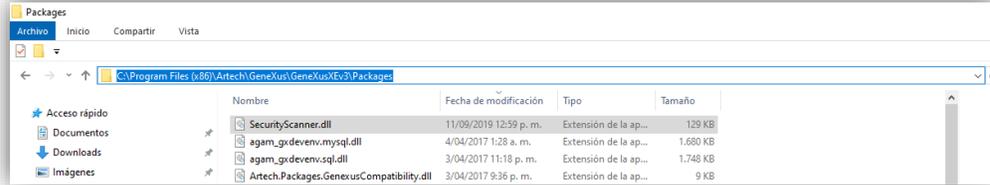
<http://wiki.gxtechnical.com/commwiki/servlet/hwikibypageid?7623>

<http://marketplace.genexus.com/product.aspx?securityscanner.es>

2.- Copie la SecurityScanner.dll ya desempaquetada en la ruta de instalación de Gx Evo 3, exactamente en:

**C:\Program Files (x86)\Artech\GeneXus\GeneXusXEv3\Packages\**

Figura. 6-11. Navegador de Windows.



Fuente: El autor

3.- Cierre el Desarrollador de Programas GeneXus.

4.- Devuélvase una carpeta y ubíquese en:

[C:\Program Files \(x86\)\Artech\GeneXus\GeneXusXEv3\](C:\Program Files (x86)\Artech\GeneXus\GeneXusXEv3\)

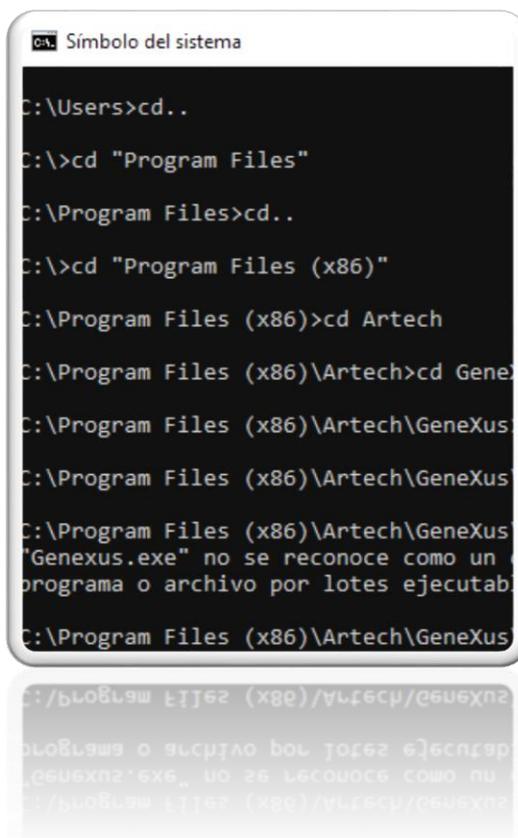
Y ejecute el comando: **Genexus.exe /install**

O sea, queda así para ser ejecutado:

[C:\Program Files \(x86\)\Artech\GeneXus\GeneXusXEv3](C:\Program Files (x86)\Artech\GeneXus\GeneXusXEv3)>**Genexus.exe /install**

Preferiblemente vaya a la ventana de DOS (símbolo del sistema) y cuando este posicionado en la carpeta, escriba: **Genexus.exe/install** presione enter, y el comando vinculará la **dll** al software para desarrollo de aplicaciones GeneXus.

*Figura. 6-12. Ejecución adición Escaner al Modelo.*



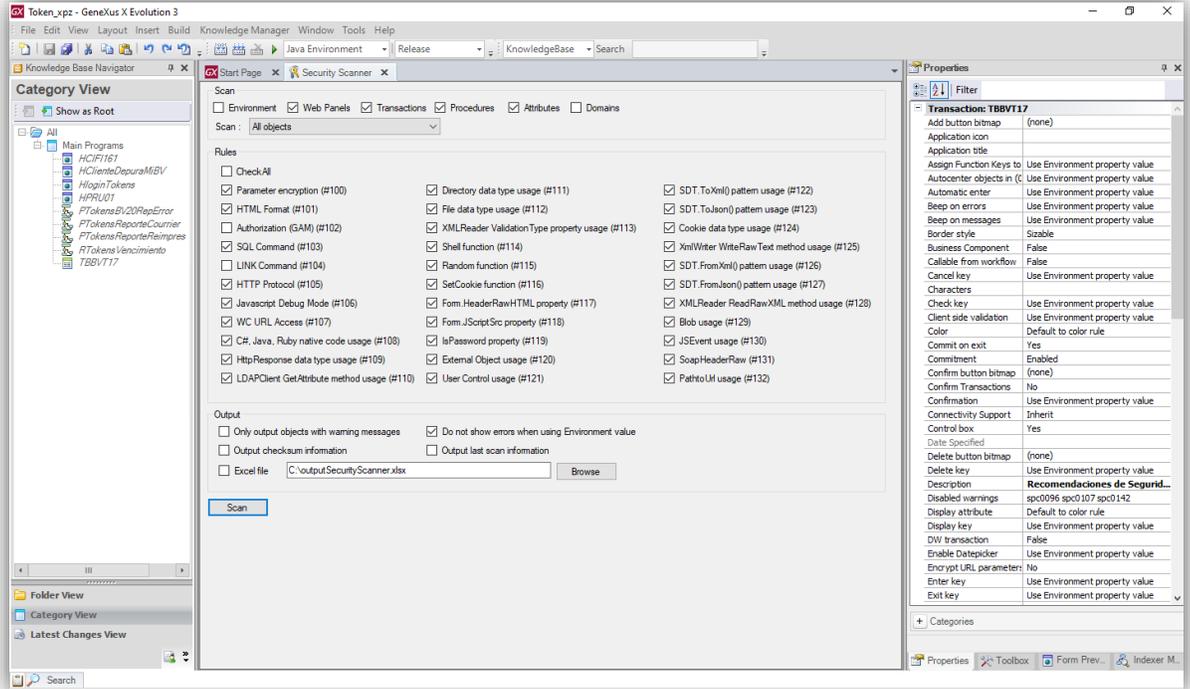
```
Simbolo del sistema
C:\Users>cd..
C:\>cd "Program Files"
C:\Program Files>cd..
C:\>cd "Program Files (x86)"
C:\Program Files (x86)>cd Artech
C:\Program Files (x86)\Artech>cd GeneXus
C:\Program Files (x86)\Artech\GeneXus>
C:\Program Files (x86)\Artech\GeneXus>
C:\Program Files (x86)\Artech\GeneXus>
"Genexus.exe" no se reconoce como un
programa o archivo por lotes ejecutab
C:\Program Files (x86)\Artech\GeneXus>
```

Figura. El Autor

5.- Luego abra de nuevo el desarrollador de aplicaciones y verifique que la función 'Open Security Scanner' haya sido vinculada en el menú de despliegue.

### 6.6.2. Pantalla herramienta Scanner en Gx-Evo 3.

Figura. 6-13. Configuración para ejecución del Escaner.



Fuente: Autor

La extensión instalada como una función más del desarrollador programador de GeneXus verifica objeto por objeto de la base de conocimientos buscando problemas de seguridad frente a los riesgos que expone OWASP.

Luego de ejecutar el arranque del scanner aparece la ventana anterior. A través de esta ventana se ejecuta toda la operación de escaneo para encontrar en todos los objetos las vulnerabilidades posibles.

### 6.6.2.1. Configuración de escaneo:

En la primera línea de la ventana del scanner de seguridad se configuran los elementos de la base de conocimiento a ser escaneados.

Figura. 6-14. Objetos a ser revisados

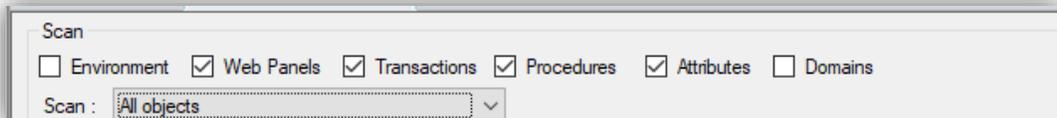


Figura. El Autor

### 6.6.2.2. Configuración de reglas:

Aquí se marcan las reglas que cada objeto debe ejecutar.

Figura. 6-15. Reglas configuradas en Escaner.

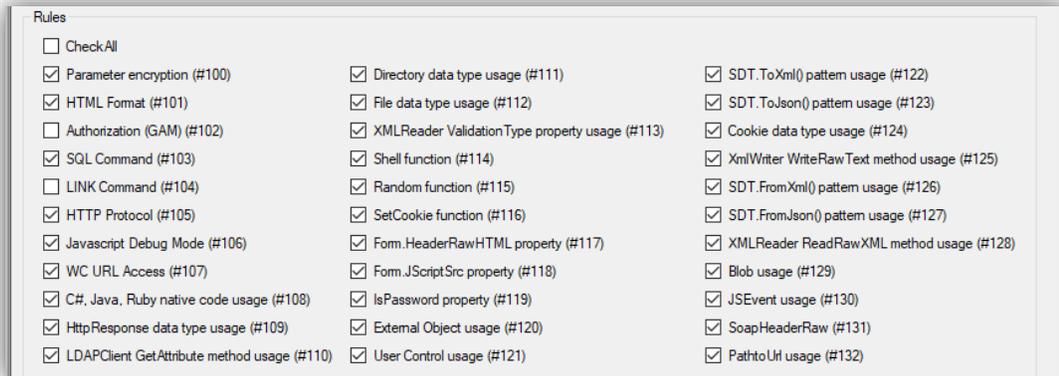


Figura. El Autor

Pueden revisarse todas las reglas con la marca en Check all.

Las reglas de Ambiente van de la #100 a la #105.

**#100 – Cifrado de parámetros:** Verifica si sus parámetros están encriptados. Si los parámetros no lo están, el mensaje que mostrará el informe es:

```
error: # 100 en WebPanel1 >> Parámetros de objeto no cifrados: # 100 en WebPanel 1 >> Parámetros de objeto no cifrados
```

**#101 – Formato HTML:** Verifica si los textos del programa están con problemas de formatos HTML. Si hay problemas muestra el siguiente mensaje:

```
error: # 101 en WWCountry >> Variable CountryName permite HTML: # 101 en WWCountry >> Variable CountryName permite HTML
```

**#102 – Autorización:** Security Scanner analiza los paneles web y las transacciones en el KB y comprueba si llaman a un programa de autorización (procedimiento). Esta regla no se aplica a las páginas maestras y los componentes web.

Si se encuentra un problema de <Autorización>, *Security Scanner* mostrará un mensaje como este:

```
error: # 102 en WebPanel1 >> No se encontró una llamada de autorización: # 102 en WebPanel1 >> No se encontró llamada de autorización
```

**#103 – Comando SQL:** Busca en los objetos los textos con comando SQL. Si encuentra problemas, se mostrará el mensaje siguiente:

```
error: # 103 en Comando >> Comando SQL encontrado en la fuente: # 103 en Comando >> Comando SQL encontrado en la fuente
```

**#104 – Comando de enlace:** Busca los enlaces y llamados (comandos link y call) con parámetros y si encuentra errores muestra el mensaje:

```
error: # 104 en WWCountry >> Se encontró el comando LINK sin parámetros
```

**#105 – Protocolo HTTP:** Busca y verifica si en los web panels y la KB (Base de conocimiento) se ha especificado el protocolo HTTPS, y si se encuentran problemas muestra el mensaje:

```
error: # 105 en WebPanel1 >> El protocolo HTTP no es seguro: # 105 en WebPanel1 >> El protocolo HTTP no es seguro
```

**#106 – En Gx X Ev1 Ajax Request Security:** Analiza el entorno de la KB y comprueba si la seguridad para Ajax se ha configurado en Alta. Mostrará el mensaje:

```
error: # 106 en WebPanel1 >> La seguridad de solicitud de Ajax no es alta: # 106 en WebPanel1 >> La seguridad de solicitud de Ajax no es alta
```

**#106 – En Gx X Ev2 JavaScript Modo depuración:** *Security Scanner* analiza la propiedad del modo de depuración de Javascript a nivel de generador; cuando está habilitado, se mostrará el siguiente mensaje:

```
error: # 106 en el generador >> El modo de depuración de Javascript está habilitado: # 106 en el generador >> El modo de depuración de Javascript está habilitado
```

**#107 – Acceso a URL de componentes WEB:** *Security Scanner* analiza los objetos KB establecidos como componentes web y comprueba si se ha habilitado el acceso

URL para ellos. Esto significa verificar si la propiedad de acceso URL se ha establecido en " Sí".

Si se encuentra un problema <WC URL Access> , *Security Scanner* mostrará un mensaje como este:

```
error: # 107 en WebPanel1 >> Componentes web con acceso URL habilitado: # 107 en WebPanel1 >> Componentes web con acceso URL habilitado
```

**#108 – Uso de código nativo C#, Java, Ruby:** analiza la sección fuente de los objetos KB y comprueba el comando java o csharp. Se muestra el siguiente mensaje:

```
error: # 108 en NativeCodeObjectSample >> Uso de código nativo encontrado en la fuente: # 108 en NativeCodeObjectSample >> Uso de código nativo encontrado en la fuente
```

**#109 – Uso del tipo de datos HttpResponseMessage:** se analiza la sección de variables de objetos KB para verificar el uso del tipo de datos HttpResponseMessage. Se muestra el siguiente mensaje:

```
error: # 109 en WebPanelSample >> Uso de tipo de datos HttpResponseMessage en variables : # 109 en WebPanelSample >> Uso de tipo de datos HttpResponseMessage en variables  
# 109: Nombre 'respuesta' Tipo 'HttpResponse'# 109 : Nombre 'respuesta' Tipo 'HttpResponse'
```

**#110 – Uso del método LDAPClient GetAttribute:** analiza la sección fuente de los objetos KB y comprueba el uso del método GetAttribute del tipo de datos de cliente LDAP.

Se muestra el siguiente mensaje:

```
error: # 110 en ProcedureSample >> Método LDAPClient GetAttribute utilizado en la fuente: # 110 en ProcedureSample >> Método GetAttribute de LDAPClient utilizado en la fuente
```

**#111 – Uso de tipo de datos Directorio:** Security Scanner analiza la sección de variables de objetos KB para verificar el uso del tipo de datos del Directorio. Se muestra el siguiente mensaje:

```
error: # 111 en WebPanelSample >> Uso de tipo de datos de directorio en variables: # 111 en WebPanelSample >> Uso de tipo de datos de directorio en variables
```

```
# 111: Nombre 'd01' Tipo 'Directorio'# 111 : Nombre 'd01' Tipo 'Directorio'
```

**#112 – Uso de tipo de datos de Archivo:** analiza la sección de variables de objetos KB para verificar el uso del tipo de datos de archivo. Se muestra el siguiente mensaje:

```
error: # 112 en WebPanelSample >> Uso de tipo de datos de archivo en variables:  
# 112 en WebPanelSample >> Archivo de datos Tipo de uso en las variables  
# 112: Nombre 'f01' Tipo 'Archivo'# 112 : Nombre 'f01' Tipo 'Archivo'
```

**#113 – Uso de la propiedad ValidationType de XMLReader :** Security Scanner analiza la sección fuente de los objetos KB y comprueba el uso de la propiedad XMLReader Data Type ValidationType. Se muestra el siguiente mensaje:

```
error: # 113 en ProcedureSample >> Propiedad XMLReader ValidationType no utiliza  
da en la fuente: # 113 en ProcedureSample >> Propiedad XMLReader ValidationType  
no utilizada en la fuente
```

**#114 – Función de Shell:** analiza la sección fuente de los objetos KB y comprueba el uso de la función Shell. Se muestra el siguiente mensaje:

```
error: # 114 en ProcedureSample >> Función de shell encontrada en la fuente: # 1  
14 en ProcedureSample >> Función Shell encontrada en fuente
```

**#115 – Función Aleatoria:** Security Scanner analiza la sección de origen de los objetos KB comprobando el uso de funciones aleatorias. Se muestra el siguiente mensaje:

```
error: # 115 en ProcedureSample >> Función aleatoria encontrada en la fuente: #  
115 en ProcedureSample >> Función aleatoria encontrada en la fuente
```

**#116 – Función SetCookie:** analiza la sección fuente de los objetos KB y comprueba el uso de la función SetCookie. Se muestra el siguiente mensaje:

```
error: # 116 en ProcedureSample >> Función SetCookie encontrada en la fuente: #  
116 en ProcedureSample >> Función SetCookie encontrada en fuente
```

**#117 – Propiedad Form.HeaderRawHTML:** analiza la sección fuente de WebPanels and Transactions y comprueba el uso de la propiedad Form.HeaderRawHTML.

Se muestra el siguiente mensaje:

```
error: # 117 en WebPanelSample >> Propiedad Form.HeaderRawHTML encontrada en la
fuente: # 117 en WebPanelSample >> Formulario . Propiedad HeaderRawHTML encontra
da en la fuente
```

**#118 – Propiedad Form.JScriptSrc:** analiza la sección fuente de WebPanels and Transactions y comprueba el uso de la propiedad Form.JScriptSrc.

Se muestra el siguiente mensaje:

```
error: # 118 en WebPanelSample >> Propiedad Form.JScriptSrc encontrada en la fue
nte: # 118 en WebPanelSample >> Formulario . Propiedad JScriptSrc encontrada en
la fuente
```

**#119 – Propiedad IsPassword:** *Security Scanner* analiza la sección fuente de WebPanels and Transactions y comprueba el uso de la propiedad IsPassword. Se muestra el siguiente mensaje:

```
error: # 119 en WebPanelSample >> Propiedad IsPassword habilitada en WebForm: #
119 en WebPanelSample >> Propiedad IsPassword habilitada en WebForm
error: # 119 en WebPanelSample >> propiedad IsPassword encontrada en la fuente e
rror : # 119 en WebPanelSample >> propiedad IsPassword encontrada en la fuente
```

```
error: # 119 en WebPanelSample >> Propiedad IsPassword encontrada en reglaserror
: # 119 en WebPanelSample >> Propiedad IsPassword encontrada en reglas
```

**#120 – Uso de objetos externos:** analiza la sección de origen de los objetos KB comprobando el uso de objetos externos.

Se muestra el siguiente mensaje:

```
error: # 120 en ProcedureSample >> Uso de objetos externos en variables: # 120 e
n ProcedureSample >> Uso de objetos externos en variables
# 120: Nombre 'myMD5' Tipo 'md5'# 120 : Nombre 'myMD5' Tipo 'md5'
```

**#121 – Uso de control de usuario:** *Security Scanner* analiza la sección WebPanels y Transacciones WebForm para el uso de los Controles de usuario. Se muestra el siguiente mensaje:

```
error : # 121 en WebPanelSample >> UserControl detectado en WebForm # 121 : Nombre 'HistoryManager' Tipo 'HistoryManager'
```

**#124 – Uso del tipo de datos de cookie:** *Security Scanner* analiza la sección de variables de objetos KB para verificar el uso del tipo de datos Cookie. Se muestra el siguiente mensaje:

```
error : # 124 en ObjectSample >> Uso de tipo de datos de cookie en variables
```

**#125 - Uso del método XmlWriter WriteRawText:** *Security Scanner* analiza la sección fuente de los objetos KB y comprueba el uso del método XMLWriter WriteRawText. Se muestra el siguiente mensaje:

```
error : # 125 en ProcedureSample >> Método XMLWriter WriteRawText utilizado en la fuente
```

**#126 - Uso del patrón SDT.FromXml ():** *Security Scanner* analiza la sección fuente de los objetos KB y comprueba el uso del método-FromXml. Se muestra el siguiente mensaje:

```
error : # 126 en WebPanel1 >> SDT. Patrón FromXml () detectado en la fuente
```

**#127 - Uso del patrón SDT.FromJson():** *Security Scanner* analiza la sección fuente de los objetos KB y comprueba el uso del método-FromJson.

Se muestra el siguiente mensaje:

```
error : # 127 en WebPanel1 >> SDT. Patrón FromJson () detectado en la fuente
```

**#128 - Uso del Método XMLReaderRawXML:** analiza KB objetos sección de cheques fuente para el XMLReader Tipo de datos's ReadRawXML Método de uso.

Se muestra el siguiente mensaje:

```
error : # 128 en WebPanel1 >> Método XMLReader ReadRawXML utilizado en la fuente
```

**#129 – Uso de Blobs** *Security Scanner* analiza la sección de variables de objetos KB para verificar el uso del tipo de datos-Blob.  
Se muestra el siguiente mensaje:

```
error : # 129 en WebPanel1 >> Uso de blob detectado en el objeto WebPanel1
```

**#130 - Uso de JSEVENT:** *Security Scanner* analiza la sección fuente de los objetos KB comprobando el uso del Método JSEvent.  
Se muestra el siguiente mensaje:

```
error : # 130 en WebPanel1 >> Uso de JSEvent encontrado en la fuente
```

**#131 - SoapHeaderRaw:** *Security Scanner* analiza la sección de origen de los objetos KB y comprueba el uso de la función no estándar SoapHeaderRaw.  
Se muestra el siguiente mensaje:

```
error : # 131 en WebPanel1 >> uso de soapHeaderRaw encontrado en la fuente
```

**#132 - Uso de PathToURL:** *Security Scanner* analiza la sección fuente de los objetos KB y comprueba el uso de la función PathToURL.  
Se muestra el siguiente mensaje:

```
error : # 132 en WebPanel1 >> uso de pathToUr1 encontrado en la fuente
```

### 6.6.2.3. Configuración de salida para resultados Scanneo:

*Figura. 6-16. Datos destino informe de salida del escáner.*

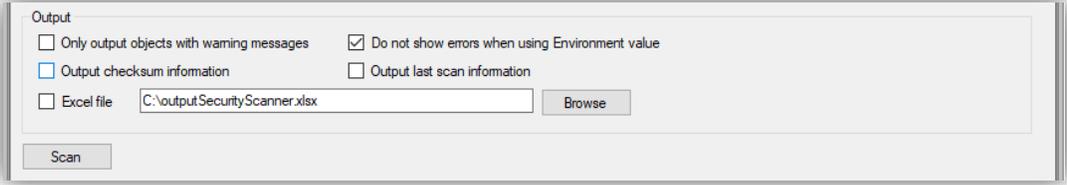


Figura. El Autor

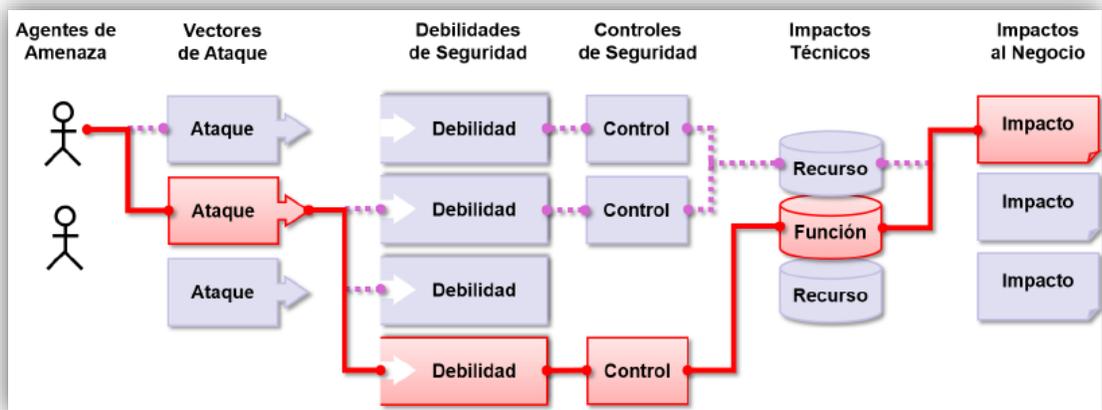
Aquí solo muestra objetos que estén en riesgo para ser advertidos. El informe genera un Excel si la casilla Excel File está marcada y el nombre del archivo y la ubicación de escritura se han ubicado con el botón Browse. Este Informe muestra las anomalías presentadas por cada objeto y sirve como base para estudiar las alternativas de protección y aplicarlas para proporcionar seguridad a la aplicación.

Por último, se utiliza el botón **Scan** para comenzar el análisis y esperar el resultado.

### 6.7. GeneXus (GX) y el OWASP Top Ten 2017.

En las aplicaciones desarrolladas los atacantes pueden utilizar diferentes caminos para cumplir con su objetivo y vulnerar las debilidades que encuentren haciendo de los productos presas fáciles y al explotar las debilidades sacar provecho propio.

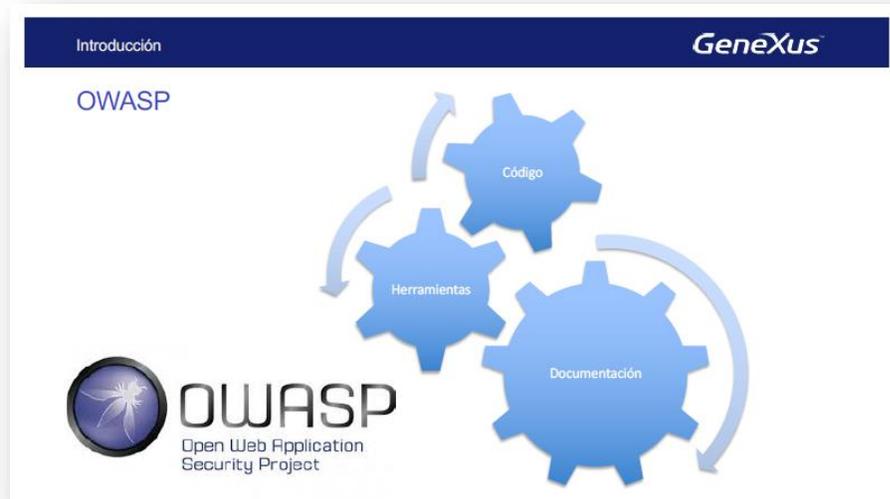
Figura. 6-17. Owasp. – Flujo ataques.



Fuente: OWASP Top 10 2017

Muchas veces las aplicaciones no se pueden franquear, pero en algunas ocasiones los atacantes logran su objetivo poniendo en riesgo los intereses de las operaciones dando lugar a grandes pérdidas e inclusive hasta la quiebra.

Figura. 6-18. Owasp. Apartes.



Fuente: GeneXus 16 y el OWASP Top 10 2017

Owasp es una fundación que trabaja para mejorar la seguridad en el software y tiene como objetivos:

- Crear conciencia para identificar los riesgos de tipo crítico que enfrentan las organizaciones.
- Proveer técnicas básicas y consejos de cómo protegerse al momento de detectar estas áreas de riesgo.

Periódicamente OWASP publica un documento estándar con el Top 10 de los principales riesgos de seguridad en las aplicaciones Web con el fin de cambiar la cultura en el desarrollo de software, en el incluye una explicación de la vulnerabilidad la forma de prevenirlo, así como diferentes escenarios de ataque.

La última publicación establece los siguientes riesgos:

### 6.7.1. A1 -Inyección.

Datos no confiables son enviados a un intérprete como parte de un comando o para hacer consultas. Estas inyecciones engañan al intérprete ejecutando comandos para acceder a datos importantes o ingresar a las aplicaciones.

Los tipos de inyecciones según el intérprete atacado:

- Inyecciones SQL
- Inyecciones LDAP
- Inyecciones XML
- Inyecciones JSON
- Inyecciones de Comandos
- Inyecciones de Código Fuente

Vamos a analizar los más importantes.

**Inyecciones SQL:** Es producida cuando un atacante logra cambiar una sentencia ejecutada por el DBMS.

GeneXus genera las consultas a la base de datos utilizando consultas con parámetros.

```
SELECT * FROM Usuarios WHERE UserName = "&UserName" and Password = "&Password"
```

```
SELECT * FROM Usuarios WHERE UserName = ? and Password = ?
setString(1,&Username)
setString(2,&Password)
```

#### **El programador hace:**

Comando SQL, comienza en el código con la palabra SQL habilitándose la escritura de comando SQL directamente. Así:

#### **Sentencia controlada:**

```
SQL SELECT COUNT(1) FROM Contratos WHERE Cliid = GAMUser.GetId()
```

#### **Sentencia sospechosa:**

```
SQL SELECT * FROM Clientes WHERE Cliid = [!&char!]
```

La variable &char es ingresada por el usuario.

```
&Sent = 'DELETE FROM CLIENTES WHERE CLICOD = 2'
SQL [!&Sent!]
```

Aquí podría cambiarse el contenido de la variable: &sent abriendo una puerta que permita ingresar a la base de datos y apoderarse del control.

#### **Mitigación:**

Debe traerse las tablas de objetos externos a través de DataViews (objeto genexus de conexión multiplataforma) y utilizar lenguaje propio de Gx para tener acceso controlado.

## **6.7.2. A2 -Autenticación Rota.**

### **Pérdida de Autenticación.**

Los atacantes pueden ganar control sobre las aplicaciones cuando usando medios manuales pueden acceder a la aplicación sobrepasando el logín de ingreso a la misma.

Se producen con frecuencia en las aplicaciones web por mal manejo de sesiones y accesos de usuario a la aplicación. Perfiles mal configurados.

Se evitan problemas manejando bien los accesos de usuarios a paginas públicas, o sea hay que tratar de exponer las aplicaciones internas en servidores públicos.

Otra forma de vulnerabilidades de este tipo es permitiendo que se hagan ataques de fuerza bruta con usuarios o combinaciones de usuarios y claves hasta que logren ingresar.

### **Tipos de Vulnerabilidades.**

De acuerdo con OWASP las vulnerabilidades se pueden presentar de varias formas

Las aplicaciones WEB contienen vulnerabilidades de autenticación rota de la siguiente manera:

- Si el atacante tiene una lista de usuarios y contraseñas y la aplicación le puede permitir ataques automáticos. Credential Stuffing.
- Permite ataque de fuerza bruta.
- Permite usuarios y contraseñas débiles: Ejemplo admin/admin. Password123, etc.
- Usa contraseñas planas, con hash débiles.
- Usa procesos de recuperación de usuarios y contraseñas bastante débiles.
- No cuenta con autenticaciones múltiples en un solo ingreso, Usuario, clave, doble clave, uso de token, biometría, etc.
- Expone identificadores de sesiones débiles.
- Expone parámetros no encriptados.
- No inactiva las sesiones después de un tiempo de inactividad.

### **Recomendaciones estándar para sortear estas vulnerabilidades:**

Implemente en los accesos a las aplicaciones las autenticaciones múltiples, como Usuario, doble password, uso de Tokens, uso de biometría, etc. con estas autenticaciones múltiples se puede bloquear los ataques de fuerza bruta.

- Aprovechone las aplicaciones con usos de sesiones por ingreso creándolas por cada acceso con un número randómico.
- No despliegue credenciales por defecto, ni siquiera para los administradores.
- Haga que las contraseñas se venzan en determinado tiempo.
- Elimine los usuarios que no acceden a la aplicación porque ya no están son activos en las empresas.
- Use obligatoriamente nomenclatura especial y compleja para la creación de usuarios y claves.
- Cuide pasar parámetros no encriptados entre llamados a objetos o nuevas estancias de la aplicación.
- Limite los intentos de sesión fallidos hasta inactivar los usuarios.

### **Autenticación:**

Caracterizada en:

- Inicio de sesión
- Gestión de contraseñas
- Recordar usuario
- Mecanismos de recuperación de contraseñas
- Actualización de las cuentas
- Validación de sesión
- Fortaleza de contraseñas
- Mensajes de error

### **GeneXus:**

- Proporciona un módulo de seguridad llamado, GAM GeneXus Access Manager (GeneXus X Evolution 2 o superior). Y que entra a gestionar la seguridad de usuarios que acceden a la aplicación.
- Utiliza mejores prácticas de la industria.
- Los tipos de autenticación soportados por la versión son: local, externa, personalizada, Facebook, twitter, google.

### **Que debe hacer el desarrollador, programador:**

- Se recomienda que utilice GAM para configurar las funcionalidades de accesos seguros.
- Implementar un módulo propio siguiendo las configuraciones de accesos seguros que recomienda OWASP.
  - Fortaleza en definición de usuarios y contraseñas.
  - Gestión de contraseñas, vencimientos, complejidad, doble contraseña, etc.

- Uso de Token si es posible.
- Actualización de cuentas.
- Creación de accesos con números de sesiones.
- Uso de mecanismos seguros de recuperación de contraseñas.

**Inicio de sesión:**

Validación que verifica la identidad de quien ingresa y detecta que es quien debe ser.

**Con el uso de:**

- Usuarios / Contraseñas.
- Tokens, USB, etc.
- Certificados digitales.
- Biometría (Huellas, Lectura iris, Lectura facial).

**Consideraciones:**

- Se recomienda que el nombre usuario no diferencie entre mayúsculas ni minúsculas.
- No permitir la enumeración de usuarios.
- Bloqueo de usuario después de un número de intentos.
- Mantener un Log de usuario, sobre todo después de cada intento de validación al ingreso. Guardando:  
Ip origen, Usuario, Contraseña, Resultado.

**Administración de contraseñas:**

**Fortaleza:**

Una contraseña es fuerte cuando es muy difícil adivinarla. Debe tenerse en cuenta la longitud y la complejidad.

El Módulo GAM de GeneXus provee políticas para contraseñas como:

- Tiempo máximo entre cambio de contraseñas (días)
- Tiempo mínimo de espera entre cambio de contraseñas (minutos)
- Largo mínimo de las contraseñas
- Mínima cantidad de caracteres numéricos en la contraseña
- Mínima cantidad de caracteres mayúscula en la contraseña
- Mínima cantidad de caracteres especiales en la contraseña
- Número máximo de contraseñas en la historia
- Largo (mínimo y máximo)
- Juegos de caracteres (a-z, A-Z,0-9, especiales)
- Histórico

**Gestión y Recuperación de Contraseñas:**

Utilizar políticas de gestión como:

- Utilizar la implementación GAM.

- La contraseña debe tener fecha de vencimiento.
- Cuando se vaya a vencer debe estar avisando a cada ingreso dentro de los x días para expirar.
- Solicitar la contraseña actual para efectuar el cambio.

**Otras consideraciones:**

- Vencimiento de cuenta de usuario.
- Mensajes de error para administración de cuenta y contraseña.

**Administración de Sesiones:**

La sesión es la instancia que se crea en el momento que el usuario hace un ingreso al sistema, puede identificarse con un número aleatorio que crea el ingreso y se guarda en una tabla del servidor donde reside la aplicación.

Los manejos de Cookies se hacen por sesiones. Las cookies mantienen el valor de variables durante la vida de la sesión de la aplicación. Con esto evitamos el paso de parámetros por objetos que facilitarían los ataques de los enemigos.

GeneXus colabora con el objeto websession para ayudar a los manejos de variables de sesión y dar utilización a estas formas de seguridad.

**Exposición de identificadores de sesión:**

**Mitigación (1)**

No transmitir o almacenar el identificador de sesión.

Propiedad Id del tipo de datos websession no debe ser almacenado o transmitido.

**Mitigación (2)**

Utilizar cookies con vencimiento para el identificador de la sesión

Debe tener las propiedades:

- SECURE – Indica a los navegadores que deben enviar esta cookie solamente a través de conexiones seguras (HTTPS).
- HTTP-ONLY – Indica a los navegadores que esta cookie no debe ser accesible por scripts (ej.: JavaScript o VBScript).

- DOMINIO – El dominio de una cookie, le indica al navegador que la misma es solamente enviada a dicho dominio o subdominios.
- PATH – El path de una cookie, le indica al navegador que la misma es solamente enviada a dicha aplicación dentro del dominio o subdominios.
- SAME SITE – Indica a los navegadores que deben enviar la cookie solamente si el pedido surgió del mismo sitio.

Valores: strict, lax, (vacía).

### **Mitigación (3)**

Deshabilitar en el servidor web los mecanismos de intercambios de identificadores de sesión no admitidos para la aplicación.

### **Fijación del identificador de sesión**

Una forma habitual de secuestro de sesión, es tratar de fijar el identificador de sesión de nuestra víctima.

### **6.7.3. A3 -Exposición de datos sensibles.**

Los atacantes roban claves y datos. Ejecutan ataques (Man in the Middle) Hombre en el medio o roban datos en texto plano directamente del servidor, en tránsito o desde el mismo cliente. Los ataques pueden ser de tipo manual, pero pueden utilizarse bases de datos con hashes que se hicieron públicas para poder obtener las contraseñas originales.

Este es un ataque de gran impacto en los últimos tiempos. El gran error, es no cifrar los datos sensibles. Deben utilizarse también al cifrar algoritmos de muy buena complejidad para evitar que los atacantes sean capaces de descifrar los datos de acceso y lograr penetrar los puntos de acceso a los programas y aplicaciones.

Las grandes fallas comprometen información especial, como información personal sensible, datos de salud, datos personales, registros bancarios, datos de tarjetas de crédito, etc.

En la transmisión de datos no protegida se corre el riesgo de ser interceptada y que el atacante logre obtener toda esta información que puede aprovechar para hacer daño de todo tipo, financiero, personal, confidencial, etc.

Por esto debe empezar por revisarse el reglamento general de protección de datos, y desde esta base realizar las labores que lleven a resguardar con encriptación compleja la protección de los datos de todo tipo reglamentarios.

### **Forma de hacer la prevención:**

Debe seguirse las siguientes recomendaciones como: (OWASP Recomendaciones)

- Clasifique los datos procesados, almacenados o transmitidos por el sistema. Identifique qué información es sensible de acuerdo a las regulaciones, leyes o requisitos del negocio y del país.
- Aplique los controles adecuados para cada clasificación.
- No almacene datos sensibles innecesariamente. Descártelos tan pronto como sea posible o utilice un sistema de tokenización que cumpla con PCI DSS. Recuerde, los datos que no se almacenan no pueden ser robados.
- Cifre todos los datos sensibles cuando sean almacenados.
- Cifre todos los datos en tránsito utilizando protocolos seguros como TLS con cifradores que utilicen Perfect Forward Secrecy (PFS), priorizando los algoritmos en el servidor. Aplique el cifrado utilizando directivas como HTTP Strict Transport Security (HSTS).
- Utilice únicamente algoritmos y protocolos estándares y fuertes e implemente una gestión adecuada de claves. No cree sus propios algoritmos de cifrado.
- Deshabilite el almacenamiento en cache de datos sensibles.
- Almacene contraseñas utilizando funciones de hashing adaptables con un factor de trabajo (retraso) además de SALT, como Argon2, scrypt, bcrypt o PBKDF2.
- Verifique la efectividad de sus configuraciones y parámetros de forma independiente.

#### **6.7.4. A4 -Entidades Externas XML (XXE).**

Con documentos XML que pueden ser ejecutados porque contienen códigos hostiles ubicándolos en lugares expuestos en donde se pueden insertar estos documentos y ser corridos logrando explotar vulnerabilidades.

Con estas ejecuciones se pueden extraer datos, ejecutar una solicitud remota desde el servidor, escanear sistemas internos, realizar ataques de denegación de servicio causando gran impacto en las entidades del negocio.

#### **Forma de hacer la prevención:**

La experiencia del programador es muy importante para que detecte en las aplicaciones donde pueden darse este tipo de ataques y bloquear las posibilidades. Además prevenir estos ataques por XXE requieren de:

- De ser posible, utilice formatos de datos menos complejos como JSON y evite la serialización de datos confidenciales.
- Actualice los procesadores y bibliotecas XML que utilice la aplicación o el sistema subyacente. Utilice validadores de dependencias. Actualice SOAP a la versión 1.2 o superior.

- Deshabilite las entidades externas de XML y procesamiento DTD en todos los analizadores sintácticos XML en su aplicación, según se indica en la hoja de trucos para prevención de XXE de OWASP.
- Implemente validación de entrada positiva en el servidor ("lista blanca"), filtrado y sanitización para prevenir el ingreso de datos dañinos dentro de documentos, cabeceras y nodos XML.
- Verifique que la funcionalidad de carga de archivos XML o XSL valide el XML entrante, usando validación XSD o similar.
- Las herramientas SAST (examinan el código fuente (en reposo) para detectar y reportar las debilidades que pueden conducir a vulnerabilidades) pueden ayudar a detectar XXE en el código fuente, aunque la revisión manual de código es la mejor alternativa en aplicaciones grandes y complejas.
- Si estos controles no son posibles, considere usar parcheo virtual, gateways de seguridad de API, o Firewalls de Aplicaciones Web (WAFs) para detectar, monitorear y bloquear ataques XXE.

#### **6.7.5. A5 -Perdida de control de acceso.**

Esta falla es detectable manualmente al ingresar a las aplicaciones y consiste en la falta de cuidado de los programadores para procurar con dificultad que los usuarios puedan acceder a las aplicaciones; uso de usuarios privilegiados y perfilados con contraseñas obligadas para catalogarse como usuarios registrados en el uso de las aplicaciones.

#### **Forma de hacer la prevención:**

Teniendo las recomendaciones siguientes según OWASP:

- Con la excepción de los recursos públicos, la política debe ser denegar de forma predeterminada.
- Implemente los mecanismos de control de acceso una vez y reutilícelo en toda la aplicación, incluyendo minimizar el control de acceso HTTP (CORS) (El CORS regula el acceso a contenidos de servidores ajenos.).
- Los controles de acceso al modelo deben imponer la propiedad (dueño) de los registros, en lugar de aceptar que el usuario puede crear, leer, actualizar o eliminar cualquier registro.
- Los modelos de dominio deben hacer cumplir los requisitos exclusivos de los límites de negocio de las aplicaciones.
- Deshabilite el listado de directorios del servidor web y asegúrese que los metadatos/fuentes de archivos (por ejemplo de GIT) y copia de seguridad no estén presentes en las carpetas públicas.
- Registre errores de control de acceso y alerte a los administradores cuando corresponda (por ej. fallas reiteradas).

- Limite la tasa de acceso a las APIs (conjunto de reglas (código) y especificaciones que las aplicaciones pueden seguir para comunicarse entre ellas) para minimizar el daño de herramientas de ataque automatizadas.
- Los tokens deben ser invalidados luego de la finalización de la sesión por parte del usuario.
- Los desarrolladores y el personal de QA (analista pruebas de calidad) deben incluir pruebas de control de acceso en sus pruebas unitarias y de integración.

#### **6.7.6. A6 -Configuración de seguridad incorrecta.**

Para obtener acceso o conocimiento del sistema del negocio los atacantes muchas veces intentarán explotar vulnerabilidades sin acceder a cuentas por defecto, páginas no utilizadas y archivos o directorios que no se hayan protegido.

#### **Forma de hacer la prevención:**

Implementense proceso de instalación que incluyan:

- Proceso de fortalecimiento reproducible que agilice y facilite la implementación de otro entorno asegurado. Los entornos de desarrollo, de control de calidad (QA) y de Producción deben configurarse de manera idéntica y con diferentes credenciales para cada entorno. Este proceso puede automatizarse para minimizar el esfuerzo requerido para configurar cada nuevo entorno seguro.
- Use una plataforma minimalista sin funcionalidades innecesarias, componentes, documentación o ejemplos. Elimine o no instale frameworks (entorno de trabajo o marco de trabajo) y funcionalidades no utilizadas.
- Siga un proceso para revisar y actualizar las configuraciones apropiadas de acuerdo a las advertencias de seguridad y siga un proceso de gestión de parches. En particular, revise los permisos de almacenamiento en la nube (por ejemplo, los permisos de buckets S3) (son contenedores de objetos para almacenar permisos en Amazon).
- La aplicación debe tener una arquitectura segmentada que proporcione una separación efectiva y segura entre componentes y acceso a terceros, contenedores o grupos de seguridad en la nube (ACLs) (listas de control de acceso).
- Envíe directivas de seguridad a los clientes (por ej. cabeceras de seguridad).
- Utilice un proceso automatizado para verificar la efectividad de los ajustes y configuraciones en todos los ambientes.

#### **6.7.7. A7 -Cross-Site Scripting (XSS) .**

Agujero de seguridad. Permite que se vulneren las aplicaciones inyectando código javascript en las páginas visitadas por los usuarios.

XSS es un vector de ataque que puede ser utilizado para robar información delicada, secuestrar sesiones de usuario, y comprometer el navegador, subyugando la integridad del sistema. Las vulnerabilidades XSS han existido desde los primeros días de la Web

#### **Forma de hacer la prevención:**

Para hacer esta prevención deben separarse los datos no confiables del contenido activo del navegador. OWASP recomienda:

- Utilizar frameworks seguros que, por diseño, automáticamente codifican el contenido para prevenir XSS, como en Ruby 3.0 o React JS.
- Codificar los datos de requerimientos HTTP no confiables en los campos de salida HTML (cuerpo, atributos, JavaScript, CSS, o URL) resuelve los XSS Reflejado y XSS Almacenado. La hoja de trucos OWASP para evitar XSS tiene detalles de las técnicas de codificación de datos requeridas.
- Aplicar codificación sensitiva al contexto, cuando se modifica el documento en el navegador del cliente, ayuda a prevenir DOM XSS. Cuando esta técnica no se puede aplicar, se pueden usar técnicas similares de codificación, como se explica en la hoja de trucos OWASP para evitar XSS DOM.
- Habilitar una Política de Seguridad de Contenido (CSP) es una defensa profunda para la mitigación de vulnerabilidades XSS, asumiendo que no hay otras vulnerabilidades que permitan colocar código malicioso vía inclusión de archivos locales, bibliotecas vulnerables en fuentes conocidas almacenadas en Redes de Distribución de Contenidos (CDN) o localmente.

#### **6.7.8. A8 -Deserialización Insegura.**

Este ataque no es tan fácil de hacer ya que los exploits distribuidos funcionan raramente. Si no hay ayuda humana este mecanismo de ataque no es muy efectivo, pero dice OWASP que lo incluye por datos de una encuesta de la industria que no tiene datos cuantificables. La idea en este ataque es lograr cambiar código dentro de la aplicación que cambie el comportamiento de la misma.

#### **Forma de hacer la prevención:**

El único patrón de arquitectura seguro es no aceptar objetos serializados de fuentes no confiables o utilizar medios de serialización que sólo permitan tipos de datos primitivos. Si esto no es posible, considere alguno de los siguientes puntos:

- Implemente verificaciones de integridad tales como firmas digitales en cualquier objeto serializado, con el fin de detectar modificaciones no autorizadas.

- Durante la deserialización y antes de la creación del objeto, exija el cumplimiento estricto de verificaciones de tipo de dato, ya que el código normalmente espera un conjunto de clases definibles. Se ha demostrado que se puede pasar por alto esta técnica, por lo que no es aconsejable confiar sólo en ella.
- Aísle el código que realiza la deserialización, de modo que se ejecute en un entorno con los mínimos privilegios posibles.
- Registre las excepciones y fallas en la deserialización, tales como cuando el tipo recibido no es el esperado, o la deserialización produce algún tipo de error. • Restrinja y monitoree las conexiones (I/O) de red desde contenedores o servidores que utilizan funcionalidades de deserialización.
- Monitoree los procesos de deserialización, alertando si un usuario deserializa constantemente.

#### **6.7.9. A9 -Uso de Componentes con Vulnerabilidades Conocida.**

Se refiere al uso de componentes tales como librerías, frameworks y otros módulos de software, que en muchas ocasiones funcionan con todos los privilegios y que se dejan libres dentro de las aplicaciones.

#### **Forma de hacer la prevención:**

Se previene a través de:

- Identificar las versiones que están usando los componentes, incluyendo dependencias, en este caso podría no usar componentes sin codificar.
- Revisar la seguridad del componente en bases de datos públicas CVE y NVD, lista de corros del proyecto y lista de correo de seguridad.
- Mantener actualizados los componentes.
- Establecer políticas de seguridad, pasar test de seguridad, licencias aceptables etc.
- Agregar capas de seguridad alrededor del componente para deshabilitar funcionalidades sin utilizar.
- Identificar los componentes de acuerdo con la versión, incluyendo dependencias.
- Actualizar constantemente los componentes en bases de datos públicas. Listas de correo del proyecto y de seguridad.
- Adoptar políticas de seguridad que regulen el uso de componentes.
- Tener la opción de agregar capas de seguridad al componente para deshabilitar funcionalidades no utilizadas.

#### **6.7.10. A10 - Registro y Monitoreo Insuficientes.**

Es la base de la mayoría de los incidentes de seguridad. Los ataques dependen de la falta de monitoreo y respuesta oportuna para lograr sus objetivos sin ser detectados. Para verificar el estatus puede examinar los registros después de las pruebas de penetración.

#### **Forma de hacer la prevención:**

Según el riesgo de los datos OWASP aconseja:

- Asegúrese de que todos los errores de inicio de sesión, de control de acceso y de validación de entradas de datos del lado del servidor se pueden registrar para identificar cuentas sospechosas. Mantenerlo durante el tiempo suficiente para permitir un eventual análisis forense.
- Asegúrese de que las transacciones de alto impacto tengan una pista de auditoría con controles de integridad para prevenir alteraciones o eliminaciones.
- Asegúrese que todas las transacciones de alto valor poseen una traza de auditoría con controles de integridad que permitan detectar su modificación o borrado, tales como una base de datos con permisos de inserción únicamente u similar.
- Establezca una monitorización y alerta efectivos de tal manera que las actividades sospechosas sean detectadas y respondidas dentro de períodos de tiempo aceptables.
- Establezca o adopte un plan de respuesta o recuperación de incidentes, tales como NIST 800-61 rev.2 o posterior. Existen frameworks de protección de aplicaciones comerciales y de código abierto tales como OWASP AppSensor, firewalls de aplicaciones web como ModSecurity utilizando el Core Rule Set de OWASP, y software de correlación de registros con paneles personalizados y alertas.

## 7. RESULTADOS Y PRODUCTOS OBTENIDOS.

El desarrollo de los pasos sistémicos que llevan a feliz término el análisis de seguridad del proyecto, desglosa al detalle un informe generalizado que muestra las debilidades encontradas en los objetos que exponen las capacidades que tienen para arriesgar la información que se maneja dentro del sistema de administración de tokens bancarios.

El mismo, incluye en referencia al top 10 de OWASP un paralelo entre cada uno de los problemas de seguridad que generalmente se encuentran en las aplicaciones y que las hacen fácilmente vulnerables. Algunos de ellos relacionados pueden tocar la pérdida de control de acceso en la validación de ingreso, otro como la facilidad de inyectar código malicioso, entre otras.

### 7.1. Nomenclatura de objetos.

Al realizar la migración del modelo desarrollado con GeneXus 9.0 así como todos sus objetos fueron renombrados por GeneXus, debido a que en versiones más recientes no es posible tener dos objetos con el mismo nombre, aunque sean diferentes los tipos de objeto. (Sin tener en cuenta la modularidad).

Hay que tener esto en cuenta para poder empatar los reportes asociados a cada uno de los modelos o aplicaciones que se quieran scannear.

Algunos ejemplos son: Para los Web Panels se agregó el prefijo “H” para los procedimientos el prefijo “P”.

Figura. 7-1. Gx 90 – Gx Evo 3.

GeneXus 9		GeneXus X Evolution 3	
Name	Type ^	Name /	Type
 ClienteBV02	Web Panel	 HClienteBV02	Web Panel

Fuente. El Autor.

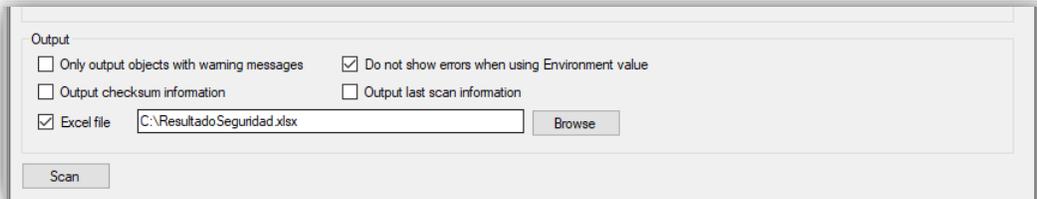
### 7.2. Scannear la Aplicación de Administración de Tokens.

Una vez montada la aplicación de Administración de Tokens, actualizada a la nueva

versión de Gx Evo 3, y superada la instalación de la dll, se procede a realizar con el buscador las vulnerabilidades de seguridad.

Marcamos la opción de salida generar Excel con el nombre: ResultadoSeguridad.xls

Figura. 7-2, Archivo destino.

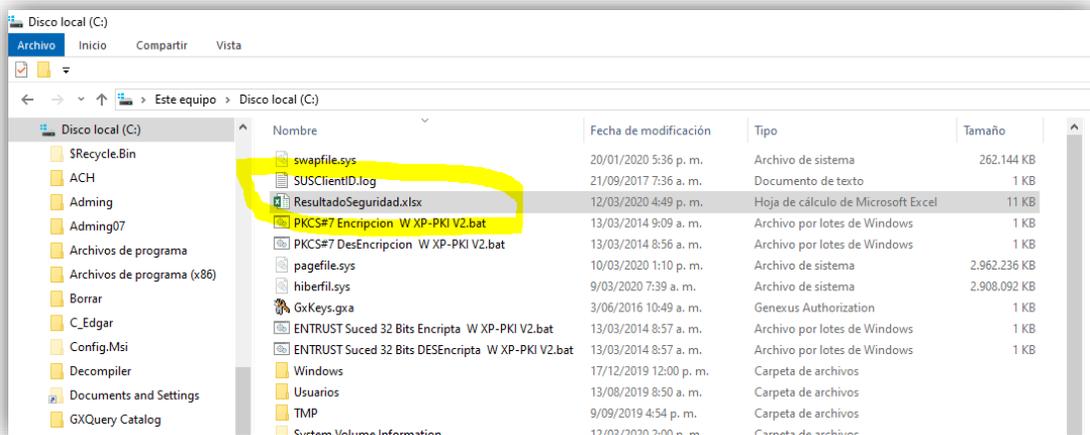


Fuente: El Autor

Y presionamos el botón Scan.

Dura unos cuantos minutos escaneando todos los objetos de la aplicación y finalmente produce el archivo Excel: **C:\ResultadoSeguridad.xlsx**

Figura. 7-3. Excel destino.



Fuente. El Autor.

### 7.3. Tabla Referencia del escáner de seguridad OWASP top 10 2017.

Esta tabla muestra la referencia entre la nomenclatura que marca OWASP y la que exhibe el Scanner de GeneXus.

Por ejemplo OWASP marca: A1 y El Scanner: #103, #108, #113, #114, #120, #126, #127, #128, #129

Figura. 7-4. Transcripción de códigos Owasp.

	#100	#101	#102	#103	#104	#105	#106	#107	#108	#109	#110	#111	#112	#113	#114	#115	#116	#117	#118	#119	#120	#121	#124	#125	#126	#127	#128	#129	#130	#131	#132	#133
A1				✓					✓					✓	✓						✓				✓	✓	✓	✓				
A2	✓					✓	✓	✓								✓																
A3						✓				✓																						
A4																																✓
A5	✓		✓			✓	✓	✓				✓	✓															✓				
A6							✓																									
A7		✓								✓								✓	✓			✓	✓									
A8														✓											✓	✓	✓					
A9																					✓	✓										
A10																																

Fuente. El Autor.

### 7.4. Resultado General del Scaneo con Seguridad para GeneXus.

Luego de lanzar el scaneo se produce una tabla de 365 registros de cada uno de los objetos que presentan posibles debilidades frente a la seguridad web que exponen en la aplicación. Esta es la lista de los objetos y su clasificación.



101	Attribute	BBT16Msj	Attribute Format allows HTML
101	Transaction	TBBVD91	HTML Attributes detected in WebForm (Name 'BBV91CdPro' Type 'HTML'. Name 'BBV91Estad' Type 'HTML'. )
101	Transaction	TBBVD94	HTML Attributes detected in WebForm (Name 'BBV94TipSo' Type 'HTML'. Name 'BBV94EstSo' Type 'HTML'. )
101	Transaction	TBBVD97	HTML Attributes detected in WebForm (Name 'BBV97EstAr' Type 'HTML'. )
101	Transaction	TBBVD98	HTML Attributes detected in WebForm (Name 'BBV98EstRg' Type 'HTML'. )
101	Transaction	TBBVT16	HTML Attributes detected in WebForm (Name 'BBT16TipB' Type 'HTML'. Name 'BBT16Msj' Type 'HTML'. Name 'BBT16Ind' Type 'HTML'. Name 'BBT16InIoE' Type 'HTML'. )
101	Transaction	TBVD016A	HTML Attributes detected in WebForm (Name 'BDoTipIpeC' Type 'HTML'. )
101	Transaction	TBVD067A	HTML Attributes detected in WebForm (Name 'BV067TipD1' Type 'HTML'. Name 'BV067TipD2' Type 'HTML'. )
101	Web Panel	HCIFI161	HTML Textblock detected in WebForm (Name 'txtMessages' Type 'HTML'. Name 'htmlTxtEmptyEvent' Type 'HTML'. Name 'txtScriptsArea' Type 'Raw HTML'. )
101	Web Panel	HClienteDepuraMiBV	HTML Attributes detected in WebForm (Name '&wSeguroDesAutorizar' Type 'HTML'. )
101	Web Panel	HColConfSolServToka	HTML Attributes detected in WebForm (Name '&BBV87TipD1' Type 'HTML'. Name '&BBV87TipD2' Type 'HTML'. )
101	Web Panel	HTokensBV01	HTML Attributes detected in WebForm (Name '&bBDO14Marca' Type 'HTML'. Name '&bBDO14Tipo' Type 'HTML'. )
101	Web Panel	HTokensBV23	HTML Attributes detected in WebForm (Name '&ChkTipo' Type 'HTML'. Name '&bBDO14Marca' Type 'HTML'. Name '&bBDO14Tipo' Type 'HTML'. )
101	Web Panel	HTokensLoad	HTML Textblock detected in WebForm (Name 'formulario' Type 'HTML'. Name 'archivo' Type 'HTML'. Name 'boton' Type 'HTML'. )
101	Web Panel	HTokensBV02	HTML Attributes detected in WebForm (Name '&bBBV87TipBn' Type 'HTML'. Name '&bBBV87Estad' Type 'HTML'. )
101	Web Panel	HTokensBV03	HTML Attributes detected in WebForm (Name '&bBBV87TipBn' Type 'HTML'. )
101	Web Panel	HTokensBV04	HTML Attributes detected in WebForm (Name '&bBDO14Marca' Type 'HTML'. )
101	Web Panel	HTokensBV05	HTML Attributes detected in WebForm (Name '&bBDO14Marca' Type 'HTML'. Name '&bBDO14Tipo' Type 'HTML'. )
101	Web Panel	HTokensBV06	HTML Attributes detected in WebForm (Name '&bBDO14Marca' Type 'HTML'. )
101	Web Panel	HTokensBV07	HTML Attributes detected in WebForm (Name '&bBDO14Marca' Type 'HTML'. Name '&bBDO14Estad' Type 'HTML'. )
101	Web Panel	HTokensBV08	HTML Attributes detected in WebForm (Name '&bBBV87TipBn' Type 'HTML'. )
101	Web Panel	HTokensBV10	HTML Attributes detected in WebForm (Name '&bBDO14Marca' Type 'HTML'. )

105	Web Panel	HTokensBV20DetaRe	HTTP protocol is not Secure
105	Web Panel	HTokensReenvio	HTTP protocol is not Secure
105	Web Panel	HTokensReporteRein	HTTP protocol is not Secure
105	Web Panel	HTokensReporteVto	HTTP protocol is not Secure
105	Web Panel	HTokensReporteVtoP	HTTP protocol is not Secure
108	Procedure	PBBVI127i	Native Code usage found in source
108	Web Panel	HloginTokens	Native Code usage found in source
113	Procedure	PBBVI127	XMLReader ValidationType property not used in source
113	Procedure	PBBVI127v	XMLReader ValidationType property not used in source
113	Procedure	PBBVI127i	XMLReader ValidationType property not used in source
115	Procedure	PBDIO007	Random function found in source
117	Web Panel	HCIFI161	Form.HeaderRawHTML property found in source
119	Web Panel	HloginTokens	IsPassword property found in source
122	Web Panel	RecentLinks	SDT.ToXml() pattern detected in source
122	Web Panel	RwdRecentLinks	SDT.ToXml() pattern detected in source
126	Web Panel	RecentLinks	SDT.FromXml() pattern detected in source
126	Web Panel	RwdRecentLinks	SDT.FromXml() pattern detected in source
130	Web Panel	HCIFI161	JSEvent usage found in source
130	Web Panel	HColConfSolServToka	JSEvent usage found in source
130	Web Panel	HTokensRechazar	JSEvent usage found in source
130	Web Panel	HTokensDevolucion	JSEvent usage found in source
130	Web Panel	HTokensBV21AprMas	JSEvent usage found in source
130	Web Panel	HTokensReenvio	JSEvent usage found in source

## Resumen de resultados de vulnerabilidades encontradas:

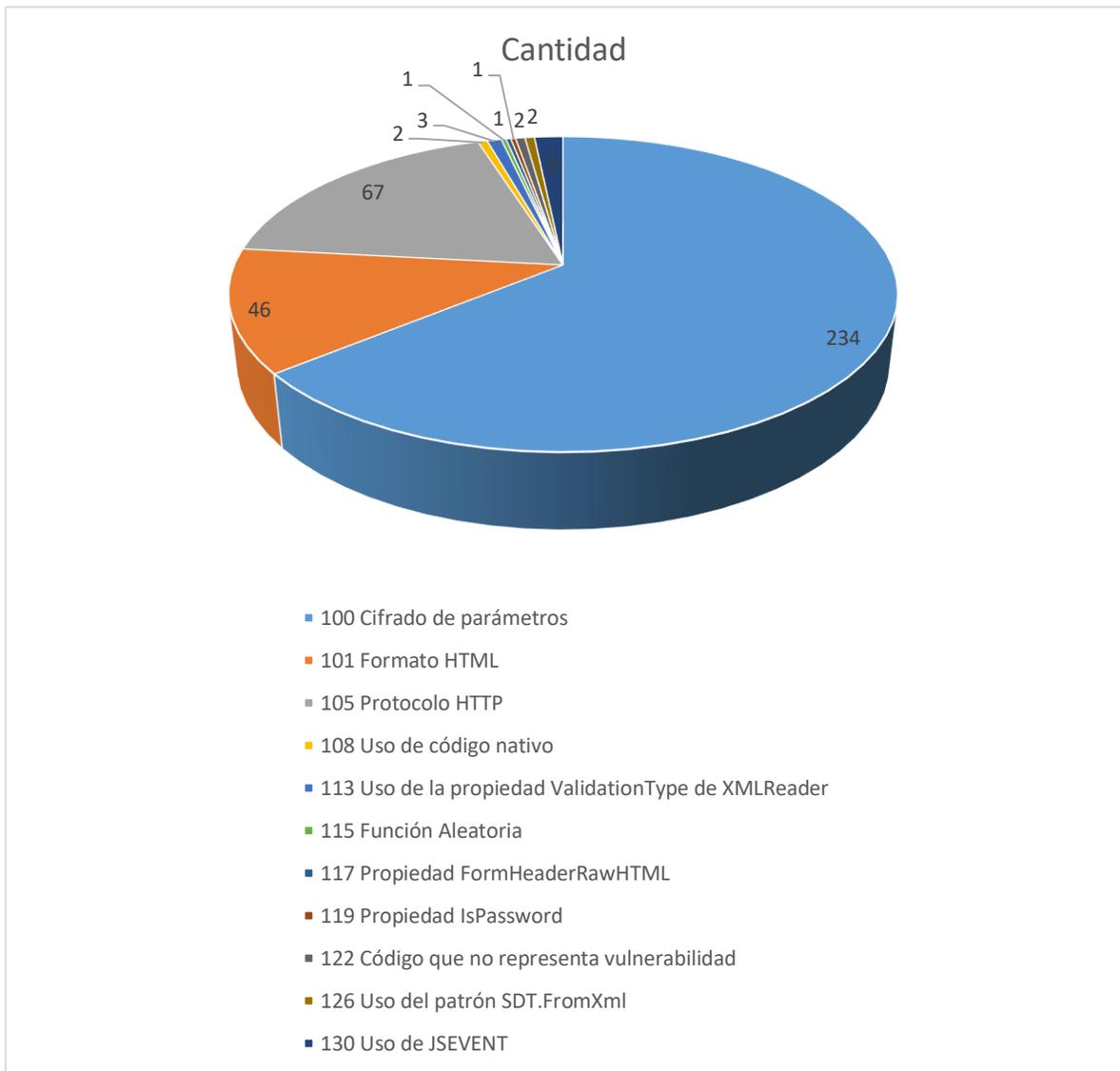
Figura. 7-5. Resumen vulnerabilidades.

Código	Nombre vulnerabilidad	Cantidad
100	Cifrado de parámetros	234
101	Formato HTML	46
105	Protocolo HTTP	67
108	Uso de código nativo	2
113	Uso de la propiedad ValidationType de XMLReader	3
115	Función Aleatoria	1
117	Propiedad FormHeaderRawHTML	1
119	Propiedad IsPassword	1
122	Código que no representa vulnerabilidad	2
126	Uso del patrón SDT.FromXml	2
130	Uso de JSEVENT	6
	<b>Total...</b>	<b>365</b>

Fuente. El Autor.

## Comportamiento gráfico de los objetos o puntos vulnerables:

Figura. 7-6. Composición vulnerabilidades.



Fuente. El Autor.

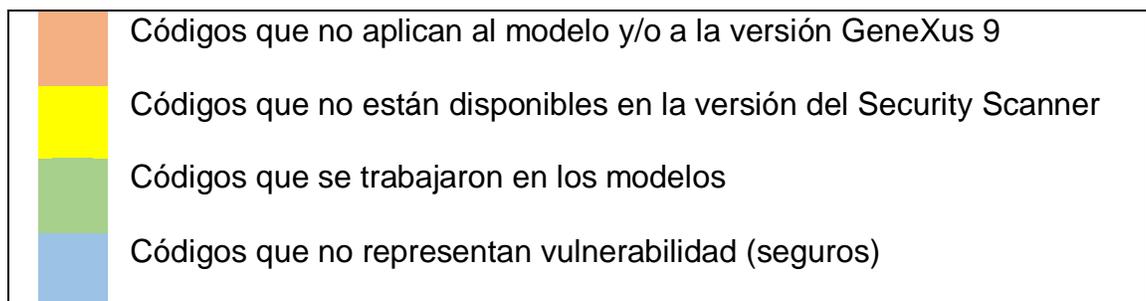
Bastante importante el número de casos presentados en la exposición de objetos de tipo Cifrado de parámetros con 234, el segundo en importancia con 67 casos es la exposición de protocolos HTTP y luego con 46 objetos los de Formato HTML; ya los demás casos son con uno, dos y tres objetos expuestos.

## 7.5. Síntesis del análisis y extracción de lo vulnerable e importante.

El anterior análisis general de todos los objetos nos mostraba todos los puntos que están expuestos a vulnerabilidades calificadas en el marco de OWASP. Ahora vamos a realizar un análisis más especializado viendo lo que verdaderamente debe ser modificado para resolver las exposiciones de acuerdo a estudios y la manera como se van desarrollando las aplicaciones realizadas, en especial esta de la Administración de Tokens Bancarios.

Para ello vamos a aplicar colores de acuerdo con la importancia de la exposición, de la siguiente manera:

*Figura. 7-7. Códigos y colores vulnerabilidades.*



Fuente. El Autor.

Exhibimos a continuación una relación de todas las reglas OWASP dentro de su enmarcación de vulnerabilidades para poder extraer las que se presentan dentro de nuestro análisis de la aplicación de Administración de Tokens.

*Figura. 7-8. Resumen más explícito vulnerabilidades.*

EXCLUIDOS	Componentes con vulnerabilidades conocidas	Deserialización Insegura	XSS	Configuración de seguridad incorrecta	Pérdida de Control de Acceso	XXE External Entity	Exposición de datos sensibles	Pérdida de Autenticación	Inyecciones
122	108	126	101	106	102	113	105	100	103
123	120	127	109		104	133	108	102	110
128	121	134	117		107		109	105	113
131		135	118		108		119	107	114
132			120		111		121	108	120
			121		112		116	116	121
			130		115		120	120	125
					120		121	121	126
					121		124	124	127
									129

Fuente. El Autor.

Para los 2 primeros colores (naranja y amarillo) para los objetos extractados no tenemos ninguna coincidencia. Los de color azul, 122 y 123 no representan códigos inseguros o no son de vulnerabilidad importante. Además los excluimos junto con el 128, 131 y 132 puesto que no se presentan. Los que están subrayados no se han presentado en el scanneo.y finalmente los mostrados en rojo, son los que en realidad deben ser revisados o mostrados en el resumen final.

Figura. 7-9. Resumen del resumen de vulnerabilidades.

Componentes con vulnerabilidades conocidas	Deserialización Insegura	XSS	Pérdida de Control de Acceso	XXE External Entity	Exposición de datos sensibles	Pérdida de Autenticación	Inyecciones
108	126	101	115	113	105	100	113
		117			119	105	126
		130					

Fuente. El Autor.

## 7.6. Vulnerabilidades de características especiales y su tratamiento.

En el modelo se encontraron diferentes posibles vulnerabilidades, de las cuales algunas por el contexto y la forma de operación, hace que esto esté controlado y no represente un alto riesgo. Por ejemplo el uso de accesos a través de usuario de Servidor AS400, el uso y aplicación de perfiles, la segmentación de la aplicación, la jerarquización de la misma que permite restringir acceso y limitar privilegios, etc.

<b>Possible vulnerabilidad</b>	<b>Parameterless LINK command found [ #100 ].</b>
<b>Objetos involucrados y/o relacionados</b>	<b>HClienteBV05, HClienteBV08, HClienteBV09, HClienteBV15, HClienteBV17, HClienteBV18, HClienteBV19, HClienteBVRep07, HClienteBV20, HClienteBV21A, HClienteBV21B, etc.</b>
<b>Problema A</b>	<b>Pérdida de Control de Acceso:</b> No se valida si el usuario final está autenticado y/o autorizado para descargar el archivo.
<b>Solución A</b>	Establecer algún método de validación antes de descargar el archivo solicitado. <ul style="list-style-type: none"> <li>• Se puede utilizar un Webservice, una consulta a la base de datos o utilizar una variable del tipo WebSession.</li> </ul>
<b>Problema B</b>	Este proceso no controla la eliminación del archivo posterior a su uso.
<b>Solución B</b>	Dar al proceso el control de eliminar el archivo posterior a su uso. <ul style="list-style-type: none"> <li>• Se puede utilizar la función sleep() y posterior eliminar el archivo descargado, utilizando una variable del tipo file.</li> </ul>
<b>Objeto Externo Relacionado</b>	<b>DOWNLOAD.</b>

<b>Posible vulnerabilidad</b>	<b>HTML Textblock detected in WebForm. [ #101 ].</b>
<b>Objetos involucrados y/o relacionados</b>	<b>HTokensLoad</b>
<b>Problema A</b>	Crea un formulario manual (etiquetas HTML), donde incluye 3 campos ‘ocultos’, uno de estos campos incluye el PATH/Ruta del archivo.
<b>Solución A</b>	No incluir campos ocultos con información sensible como la ruta del archivo. Utilizar otros métodos para pasar la información y/o cambiar la funcionalidad del programa externo para que este controle la ruta del archivo.
<b>Solución B</b>	<b>Utilice varios métodos para pasar información sensible:</b> <ul style="list-style-type: none"> <li>• Utilizar una variable del Tipo WebSession en GeneXus y guardar la información. Para recuperar la información en el programa externo se debe consumir y utilizar la clase apropiada distribuida por GeneXus para obtener la información.</li> <li>• Crear y consumir un WS/API donde se pase el archivo en base64 y la información adicional cifrada.</li> </ul>
<b>Objeto Externo Relacionado</b>	<b>SERVLETUPLOAD</b> – Programa Java que luego se convierte en una librería que consume llamados para alojar archivos de Base de Datos de subida al AS400 a través de una ubicación dentro del servidor de aplicaciones.

## 7.7. Análisis de resultados e impactos.

Lo mencionado anteriormente expone el reto de adoptar unas buenas prácticas para el desarrollo de los nuevos productos y la corrección de los existentes logrando la conciencia de proteger los intereses de los clientes a través de muchas alternativas, empezando por la adopción de mejores y más seguros desarrollos para las aplicaciones.

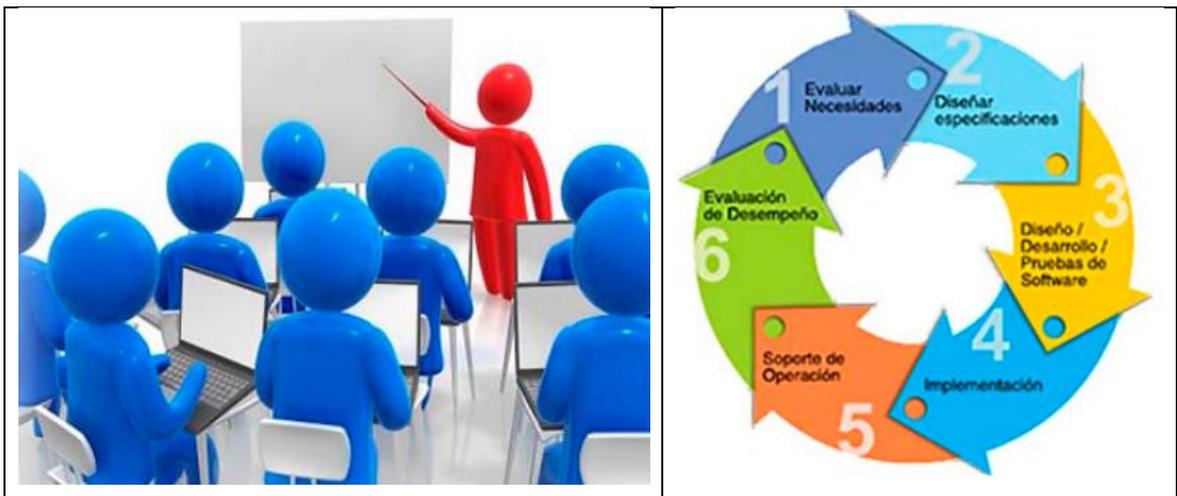
Estas buenas prácticas deben trascender a todos los desarrollos de las áreas de tecnología e inclusive penetrar en desarrollo que no sean desprendidos y apoyados por el generador GeneXus (Gx). Además recomendar la creación de la fábrica de software que promueven las grandes compañías para hacer software genérico y masivo para las diferentes industrias; entre ellas la industria del dinero y el manejo de las finanzas.

Las fábricas de software tienen como objetivo el desarrollo de productos a la medida, que al final se elaboran con los conocimientos de negocio que posee el cliente, es decir el cliente es quien dirige el desarrollo de sus aplicaciones, y la fábrica le proporciona conocimientos, agilidades, profesionalismo metodológico en el manejo de proyectos y personal para que los desarrolle. El cliente finalmente

queda como dueño de la propiedad intelectual de los desarrollos. En estos desarrollos es donde la metodología y mejores prácticas tanto de logística como de seguridad de la información juegan un plus destacado; porque suman el aseguramiento informático a la calidad para la puesta en producción de mejores y más actualizados productos.

En resumen, la fábrica de software debe propender la adopción de metodología con el uso de buenas prácticas de desarrollo seguro para que, al generar productos en masa impulse el aseguramiento y la calidad de los productos a entregar y se hagan los desarrollos cada vez menos artesanal y más profesionales.

Figura. 7-10. Resultados e impactos.



Fuente: IT Consultores. <http://www.itconsultores.com.co/services/software1-2?lang=en>

### 7.8. Debilidades en el uso de Tokens criptográficos.

Aunque la orientación del objetivo de éste estudio no es directamente el Token y sus debilidades, vamos a expresar algunas de ellas muy puntuales.

Exponer la información de los tokens en los logs de gestión. Los logs de gestión están ubicados en las áreas de máquina virtual y si un atacante logra penetrar hasta esta área, puede conocer el serial y la identificación del cliente asignado al token,

la cual es bastante información de seguridad del dispositivo.

Secuestro de Tokens facilitados por acceder a una sesión y poder tomar el control a través de cross-site ó scripting, usando inyección de código java-script o html se puede tomar el control de una aplicación y de sus tokens de seguridad, aunque la posibilidad es mínima.

Por todas estas posibilidades deben generarse Tokens robustos contratados con empresas serias y reconocidas que cumplan con todos los sellos de calidad que sean acreditados y superen las expectativas esperadas.

Los tokens y su sistema de verificación deben generar números aleatorios de difícil adivinación en su algoritmo de secuencia. Además acompañar la validación y verificación de autenticidad con otros identificadores como IP, cédula del cliente, tipo de identificación, etc.

También generar protección en el uso de los tokens a través de no transmitir información de los tokens en la Url, hacerlo a través de cookies y en el código interno de la aplicación, evitar sesiones simultaneas con la misma identificación del cliente, no hacer visible la digitación de los seriales de identificación, inactivar las sesiones después de tiempos medidos de inoperabilidad, evitar sesiones simultaneas de un mismo cliente, es decir manejar número de sesión para cada ingreso e inactivar los anteriores, además de todas las medidas que se aconsejan a nivel de desarrollo seguro.

Tanto la semilla como el inventario pueden ser tomados por atacantes para luego de tener su dominio poder hacer cambios en los estados de las mismas; como por ejemplo, una semilla de token que esté inactiva, alterar su estado y activarla para ponerla en uso y poder operar el sistema. Poder también cambiar a una semilla activa su identificación para que la pueda usar otro cliente. Poder alterar cualquier dato del inventario y secuestrar la información o volver loco el sistema; afortunadamente hay un buen control de acceso que puede limitar las acciones de los atacantes y el poner la aplicación en un servidor de aplicaciones internas ayudaría muchísimo en la seguridad; pero desafortunadamente muchas veces se hace residir un aplicativo en servidores expuestos porque no hay presupuesto y se toman acciones equivocadas como esta.

## 8. CONCLUSIONES

En todos los procesos de gestión tanto en las empresas como en la vida cotidiana se incluye la tecnología; está visto que darse a conocer al mundo exterior es exponerse a las posibilidades de ser intersectados vía internet y quedar expuestos a que los ataques sean mal intencionados y sean aprovechables para perder tanto oportunidades de negocio como capital económico.

Es por esto que deben estudiarse todas las posibilidades de ser atacados y establecer todos los mecanismos defensivos que puedan adoptarse para resguardar todos los intereses de una compañía, y más si es de tipo financiero.

Las compañías de tipo financiero no solo exponen sus propios capitales y recursos, si no el de todos sus clientes que han depositado la confianza en ellas; y también han encargado sus ahorros para que sean manejados y obtengan intereses que hagan crecer sus capitales. Además, también grandes capitales y movimientos de que abarcan los intereses de clientes empresariales para las entidades financieras, cubriendo sus nóminas y facilitando los pagos de las obligaciones en las que están comprometidos en donde todo este tipo de operaciones deben llevar un componente de seguridad de alto compromiso.

Los tokens como una unidad de seguridad brindan confianza a los usuarios para acceder con tranquilidad a las aplicaciones financieras y es por eso que su administración debe sumar toda la seguridad que se necesita para afianzar su uso obligatorio.

OWASP es la entidad que resume todas las posibles vulnerabilidades que pueden atacar en todos los escenarios posibles a los accesos que comprometen la red de internet.

OWASP crea conciencia de la necesidad de andar protegiéndose a toda hora de los posibles ataques de piratas informáticos y además provee las técnicas y herramientas para contrarrestar y preveer el riesgo.

La aplicación de Administración de Tokens tiene tres grandes procesos que implican ser bien administrados y protegidos; el proceso de cargue de semillas al inventario de tokens, la creación automática de las solicitudes de tokens desde la banca virtual en el momento de la operación de registro de los clientes o la reposición de los

tokens y por último la asignación de los tokens a las solicitudes por parte de los clientes.

El desarrollo de programas realizado en el pasado no contemplaba la aplicación de protocolos de desarrollo seguro, puesto que para ese entonces no había tanta exposición a intrusos y no revestía mucha importancia como la que se exige ahora por el incremento del uso masificado de internet.

Los riesgos más significativos que presenta el estudio de esta aplicación son, la inyección de código, la exposición de parámetros sin encriptar, la exposición de código html para facilitar accesos, y la autenticación rota que facilita acceder a los servicios de las aplicaciones.

Las medidas a tomar para mitigar los riesgos son: en el momento de conectar objetos que involucren parámetros, estos deben ir encriptados o pasarlos a través de variables de sesión con cookies, no incluir campos ocultos con información sensible como las rutas de los archivos, cuando se descargan archivos establecer métodos de validación antes de realizar las acciones, etc.

Cuidar de campos digitados por el usuario ya que a través de ellos puede accederse inyectando instrucciones sql o código html de significada importancia como vulnerabilidad aprovechable; utilice validaciones en la digitación de tal manera que solo se capture información bien parametrizada y validada.

De todas estas experiencias se recoge que ellas multiplican conocimiento para ser aplicado a los nuevos desarrollos de tal manera que queden estándares probados para implantarse y hacer de las aplicaciones sitios más seguros.

Las experiencias vividas y aprendidas durante el desarrollo de este trabajo se suman para enriquecer los estándares a ser determinados con los conceptos de fábricas de software y establecer manuales y directivas de procesos aplicando desarrollo seguro en las futuras aplicaciones.

## 9. BIBLIOGRAFIA

ASOBANCARIA. (02 de 08 de 2018). Canales y seguridad. Recuperado el 02 de 08 de 2018, de ASOBANCARIA: <http://www.asobancaria.com/sabermassermas/home/consumidor-informado/mas-acerca-de-los-bancos/canales-y-seguridad/>

Avila Forero, R. (13 de 08 de 2018). Revista Dinero. Recuperado el 21 de 09 de 2018, de ¿Bancarizar o no bancarizar?: <https://www.dinero.com/Item/ArticleAsync/260869>

Dacchan T., J. C. (21 de 09 de 2018). Ley de Delitos Informáticos en Colombia. Recuperado el 21 de 09 de 2018, de DELTA Asesores: <https://www.deltaasesores.com/ley-de-delitos-informaticos-en-colombia/>

Economia Simple.Net. (02 de 08 de 2018). Definición de Banca electrónica. Recuperado el 02 de 08 de 2018, de Economía Simple.Net: <https://www.economiasimple.net/glosario/banca-electronica>

GRUPO SANTANDER S.A. (2018). ¿Qué es el Token de Seguridad? Recuperado el 07 de 09 de 2018, de GRUPO SANTANDER S.A.: <https://www.santanderrio.com.ar/banco/online/personas/pagar-y-transferir/token-de-seguridad/faq>

Mercado, I. (23 de 04 de 2018). Internet Society - Capitulo Republica Dominicana. Recuperado el 21 de 09 de 2018, de SEGURIDAD DE LAS TRANSACCIONES ELECTRÓNICAS: <https://isoc-rd.org.do/publicaciones/recursos/seguridad-de-las-transacciones-electronicas/>

Pabón Cadavid, J. A. (02 de 08 de 2018). La criptografía y la protección a la información digital. Revista U Externado, <https://revistas.uexternado.edu.co/index.php/propin/article/view/2476/3636>.  
Obtenido de La criptografía y la protección a la información digital: <https://revistas.uexternado.edu.co/index.php/propin/article/view/2476/3636>

Periódico el colombiano. (12 de 04 de 2018). Colombia, el sexto país con más ciberataques en 2017. Recuperado el 02 de 06 de 2018, de El Colombiano: <http://www.elcolombiano.com/colombia/ciberataques-en-colombia-sexto-pais-mas-vulnerable-en-la-region-AB8535174>

Revista Dinero. (07 de 07 de 2016). Bancos se preparan para la nueva era de transacciones móviles. Recuperado el 21 de 09 de 2018, de Revista Dinero: <https://www.dinero.com/edicion-impresatecnologia/articulo/bancos-se-preparan-para-la-nueva-era-de-transacciones-moviles/225415>

Revista Dinero. (02 de 02 de 2017). El apetitoso negocio del cibercrimen. Recuperado el 02 de 07 de 2018, de Revista Dinero: <https://www.dinero.com/edicion-impresatecnologia/articulo/las-cifras-que-mueven->

el-ciberdelincuencia-a-nivel-global/241593

Revista Dinero. (02 de 02 de 2017). [www.dinero.com](http://www.dinero.com). Recuperado el 20 de 08 de 2018, de [www.dinero.com](http://www.dinero.com): <https://www.dinero.com/edicion-impresita/tecnologia/articulo/las-cifras-que-mueven-el-ciberdelincuencia-a-nivel-global/241593>

Revista Dinero. (13 de 03 de 2018). Internet le roba terreno a las oficinas a la hora de hacer trámites financieros. Recuperado el 15 de 08 de 2018, de Revista Dinero: <https://www.dinero.com/economia/articulo/operaciones-financieras-en-colombia-en-2017/256283>

Seguridadensistemascomputacionales.zonalibre.org. (04 de 02 de 2011). Encriptación. Recuperado el 07 de 09 de 2018, de [seguridadensistemascomputacionales.zonalibre.org](http://seguridadensistemascomputacionales.zonalibre.org/): <http://seguridadensistemascomputacionales.zonalibre.org/>

Superintendencia Financiera de Colombia. (s.f.). Obtenido de <https://www.superfinanciera.gov.co/publicacion/20148>

Superintendencia Financiera de Colombia. (02 de 07 de 2018). Glosario. Recuperado el 02 de 07 de 2018, de Superintendencia Financiera de Colombia: <https://www.superfinanciera.gov.co/jsp/Glosario/user/main/letra/B/f/0/c/00>

Superintendencia Financiera de Colombia. (07 de 09 de 2018). [www.superfinanciera.gov.co](http://www.superfinanciera.gov.co). Obtenido de <https://www.superfinanciera.gov.co/jsp/loader.jsf?lServicio=Publicaciones&lTipo=p-publicaciones&lFuncion=loadContenidoPublicacion&id=11268&dPrint=1>

Textos Científicos. (09 de 11 de 2006). Encriptación. Recuperado el 21 de 09 de 2018, de Textos Científicos: <https://www.textoscientificos.com/redes/redes-virtuales/tuneles/encriptacion>

Welive Security. (20 de 05 de 2017). Cambios en la norma para gestionar la seguridad de la información. Recuperado el 02 de 09 de 2018, de Welive Security: <https://www.welivesecurity.com/>

XARXA AFIC El portal del Comerciante. (07 de 09 de 2018). LA SEGURIDAD EN LAS TRANSACCIONES. Recuperado el 07 de 09 de 2018, de XARXA AFIC El portal del Comerciante: <https://www.portaldelcomerciante.com/es/articulo/la-seguridad-transacciones#arriba>

Yañez, C. (08 de 11 de 2017). CEAC Planeta Formación y Universidades. Recuperado el 21 de 09 de 2018, de TIPOS DE SEGURIDAD INFORMÁTICA: <https://www.ceac.es/blog/tipos-de-seguridad-informatica>

GeneXus Community Wiki. 2017. Managing OWASP Top 10 2017 in GeneXus Applications. <https://wiki.genexus.com/commwiki/servlet/wiki?39917,A1%3A+2017+--+Injection>  
GeneXus Wiki

<https://wiki.genexus.com/commwiki/servlet/wiki?18702,Security+Scanner+extensio>

n+user+manual,  
GeneXus Security Scanner.

<https://wiki.genexus.com/commwiki/servlet/wiki?39951,Genexus+Security+Scanner+Documentation>

---

## REFERENCIAS

- <sup>i</sup>1 La Opinión: Revista. <https://www.laopinion.com.co/colombia/mas-del-55-de-los-casos-de-ciberdelito-son-cuentas-bancarias-180597#OP>
- <sup>ii</sup>2 Revista Dinero. Sistema Financiero. 3/13/2018  
<https://www.dinero.com/economia/articulo/operaciones-financieras-en-colombia-en-2017/256283>
- <sup>iii</sup>3 Colombia. Tecnología <https://www.colombia.com/tecnologia/internet/colombia-es-el-sexto-pais-en-latinoamerica-con-mayor-numero-de-ciberataques-188870>
- <sup>iv</sup>4 ASOSEC. Asociación Colombiana de Seguridad.  
<https://asosec.co/2018/05/colombia-en-el-ranking-de-paises-con-mas-ciberataques-en-latinoamerica/>
- <sup>v</sup>5 Policía Nacional. Delitos Informáticos. <https://www.policia.gov.co/ciberseguridad>
- <sup>vi</sup>6 Conceptos tomados de las referencias:  
-Superintendencia financiera de Colombia (Conformación del Sistema Financiero Colombiano, consultado el 16 de septiembre de 2018, (Superintendencia Financiera de Colombia, 2018),  
-“Citibank (2018) Que son las sociedades fiduciarias?”, consultado el 2018/09/07.  
-“Asobancaria (2016) Que es leasing?”, consultado el 2018/09/07,  
<http://www.asobancaria.com/sabermassermas/que-es-leasing/>
- <sup>vii</sup>7 Publicación econectia Fuente: <https://www.econectia.com/blog/que-es-encryptacion-de-datos>
- <sup>viii</sup>8 Fuente: <https://www.genbeta.com/desarrollo/que-son-y-para-que-sirven-los-hash-funciones-de-resumen-y-firmas-digitales>
- <sup>ix</sup>9 Fuente: <https://es.wikipedia.org/wiki/Biometr%C3%ADa>
- <sup>x</sup>10 Superfinanciera de Colombia.  
<https://www.superfinanciera.gov.co/jsp/Publicaciones/publicaciones/loadContenidoPublicacion/id/10097769/f/0/c/00>
- <sup>xi</sup>11 Revista Dinero. Ley de Habeas Data. 14/08/2009.  
<https://www.dinero.com/economia/articulo/beneficios-ley-habeas-data/76430>
- <sup>xii</sup>12 Delta Asesores. <https://www.deltaasesores.com/ley-de-delitos-informaticos-en-colombia/>

---

<sup>xiii</sup>13 Banco de la República. <http://www.banrep.gov.co/es/inclusion-financiera-informe-especial-estabilidad-financiera>

<sup>xiv</sup>14 OWASP. <https://www.dragonjar.org/owasp-top-ten-project-en-espanol.xhtml>