Fundamentos de 'tokenización' y el cumplimiento de PCI DSS

David Eduardo Acosta

Consultor senior de Seguridad de la Información de Internet Security Auditors

Con el objetivo de proteger la información confidencial que se almacena, se transmite y se procesa en un entorno corporativo, existen diversas alternativas técnicas para hacer ilegibles dichos datos y evitar que sean visualizados por personal no autorizado. Algunos de esos mecanismos de protección de información confidencial son:

- Uso de técnicas de *hash* (cifrado no reversible).
- Cifrado simétrico y asimétrico.
- Truncamiento.
- Trasposición.
- Enmascaramiento.
- 'Tokenización'.

Dependiendo de las necesidades específicas, se puede optar por la implementación de uno u otro mecanismo e incluso un conjunto de ellos. En este artículo se enfocará el análisis en el concepto de "tokenización" o uso de *tokens*[1]: valores únicos asociados a datos confidenciales que son empleados como reemplazo de estos últimos, con la característica diferencial de que el *token* no permite inferir el dato confidencial con el que se relaciona, minimizando de esta manera el riesgo de almacenamiento inseguro y la necesidad de implementación de controles asociados a datos confidenciales, ya que el *token* en sí no es catalogado como un dato confidencial.

El ámbito de uso de estos dispositivos es bastante amplio, permitiendo minimizar el riesgo y el entorno involucrado en la gestión de información sensible y cubriendo requerimientos legales, operativos y administrativos: protección de información de carácter personal (LOPD), datos de seguridad social, información clínica, patentes y datos de tarjetas de pago -a los que afecta la normativa *Payment Card Industry Data Security Standard* (PCI DSS)-, entre otros.

A lo largo del escrito se describirá el proceso general vinculado a la arquitectura y gestión del ciclo de vida de los componentes de la 'tokenización', así como la aplicación de esta técnica en un ambiente de cumplimiento normativo, poniendo especial énfasis en la norma de protección de datos de tarjetas de pago PCI DSS.

La arquitectura básica de un sistema de 'tokenización' contempla tres componentes principales: una base de datos maestra centralizada, un servicio de cifrado/'tokenización' e interfaces de comunicación. Veamos dichos componentes con más profundidad:

La BD maestra, centralizada

Posterior a la realización de un proceso interno de identificación y depuración de datos confidenciales dentro del entorno afectado (tal como se explicará más adelante), los datos confidenciales que tienen que ser protegidos deben ser centralizados en una única base de datos (BD) o repositorio seguro (*Data Vault*, en inglés). Junto con cada dato a ser protegido se debe relacionar su *token* correspondiente, garantizando su referencia única. Esta base de datos deberá

Nota 1: *A lo largo del texto se hará referencia a "token" y "tokenización" por ser palabras más difundidas en el ámbito técnico. Sin embargo, su uso correcto en castellano debería ser "testigos" y "uso de testigos", debido a su similitud con las carreras de relevos.*

contemplar los siguientes controles de seguridad:

- **Aislamiento:** Al ser el único elemento que almacena datos confidenciales y que contiene la referencia entre *token* y dato confidencial, la base de datos maestra requiere ser aislada del resto de componentes del sistema. Aquellas interfaces de comunicación de entrada/salida de datos de la base maestra deben ser controlados y securizados.
- **Cifrado**: Debido a que almacena datos confidenciales, la base de datos maestra debe contener esta información cifrada, cubriendo todos los controles relacionados con gestión de claves, custodia y copia de seguridad. Es importante tener en cuenta que los controles de 'tokenización' y cifrado son complementarios y no son excluyentes.
- **Alcance**: El *token* y su referencia al dato confidencial deben ser válidos únicamente para un entorno limitado. Esto garantiza que, en el caso hipotético de que el listado completo de *tokens* y referencias sea comprometido, esta información no tendrá valor alguno fuera del entorno definido. En el caso de que se requiera compartir el dato confidencial con un ente externo, se recomienda implementar rutinas de conversión *token*-a-*token* y no entregar o compartir los datos confidenciales ni permitir acceso no controlado a la base de datos maestra centralizada.
- **Autenticación, Autorización y Registro (AAA)**: Únicamente el personal aprobado por la organización debe poder tener acceso a la base de datos maestra centralizada y su autenticación debe ser robusta, para prevenir accesos no autorizados. Igualmente, es requerido implementar mecanismos de monitorización sobre la base de datos maestra con el fin de identificar cualquier actividad anómala o sospechosa en las peticiones de conversión *token*-dato confidencial y mantener trazabilidad sobre transacciones y acciones realizadas.
- **Disponibilidad**: Al convertirse en punto único de fallo y ser altamente crítica para la gestión de la organización, se debe disponer de una estrategia de disponibilidad que garantice la continuidad de la operación en caso de problemas.

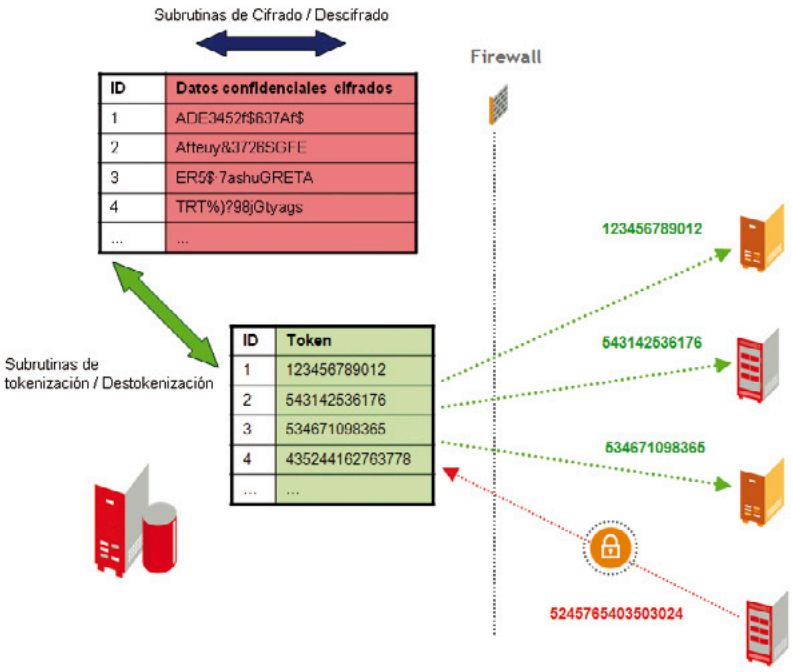

Figura 1: *Esquema de tokenización: La base de datos se encuentra aislada del resto de sistemas y mantiene mantiene una estructura de pares "dato confidencial–token", donde "dato confidencial" se almacena cifrado.*

Servicio de cifrado/'tokenización'

El proceso operativo de 'tokenización' consta de dos partes:

- Cifrado/descifrado de la información confidencial almacenada en la base de datos maestra centralizada.
- Asignación de un *token* a dicho dato cifrado.

Por lo tanto, se requiere de un servicio que provea el cifrado/descifrado de la información almacenada en la base de datos y que gestione la asignación y referencia de *tokens* uno-a-uno con dicha información en los procesos de entrada y salida de datos confidenciales.

Estos componentes (cifrado y 'tokenización') pueden estar contenidos dentro de un mismo servidor o separados, permitiendo independizar la operatividad si se considera necesario.

Interfaces de comunicación

Finalmente, se encuentran las interfaces de comunicación que engloban cualquier conexión entre componentes internos (componente de cifrado, componente de 'tokenización' y llamadas a la base de datos) y aplicaciones externas (rutinas de recuperación para aquellos procesos que lo requieren por consideraciones de negocio o legales y recepción de datos para ser 'tokenizados'). Por lo general, estas interfaces están basadas en llamadas a conectores entre componentes, *Applications Programming Internet* (API), *Web Services*, etc. Cualquier entrada o salida desde o hacia estas interfaces deben ser registradas, definiendo controles de acceso específico, seguridad en el transporte y acciones en caso de eventos sospechosos.

Teniendo en cuenta que un dato 'tokenizado' no es información confidencial per se, cualquier otro control que se aplique para la protección de esta información en el resto de la infraestructura dejará de ser obligatorio y pasará a ser recomendable.

Todo el proceso se puede ver en la Figura 1: Se cuenta con una base de datos centralizada, la cual se encuentra aislada del resto de sistemas en el entorno informático. Cualquier entrada o salida de datos es registrada. La base de datos mantiene una estructura de pares "dato confidencial–*token*",

donde “dato confidencial” se almacena cifrado. Internamente se realizan las subrutinas de cifrado/descifrado y generación/asignación de *tokens*. En la salida de *tokens* aplican los controles de transmisión normales conforme con las directivas de la organización, mientras que en la recepción o llegada de datos confidenciales se deben aplicar controles de cifrado, integridad y monitorización en la transmisión. Una vez presentado el esquema típico y los componentes de un sistema que emplee 'tokenización' (y cifrado), veremos en detalle el ciclo de vida del *token*, el cual pasa por diferentes procesos: generación, asignación, transmisión, almacenamiento y eliminación.

Generación

La clave en el proceso de generación de *tokens* es garantizar que el dispositivo nunca pueda ofrecer información directa ni permitir inferir los datos confidenciales con los cuales está relacionado. Dependiendo de las necesidades, el *token* puede reemplazar total o parcialmente el dato confidencial y se puede generar asociado a un dato específico de forma dinámica (por ejemplo, asociado a una transacción) o de forma estática (asociado a un dato confidencial hasta que dicho dato deja de ser válido).

El valor del *token* puede ser generado de forma pseudoaleatoria, secuencial, usando trasposición o empleando funciones criptográficas de una sola vía (*hash*) o criptografía simétrica. En cualquier caso, se debe validar que el *token* es único para cada dato confidencial a ser protegido y que no se repetirá o 'colisionará' con otro dato para asegurar la integridad en las referencias. Añadido a esto, las funciones o subrutinas generadoras de *tokens* deben ser protegidas y controladas para evitar la generación no autorizada de este tipo de dispositivos.

Igualmente –y con el fin de minimizar el impacto en el procesamiento y almacenamiento y optimizar la compatibilidad– es recomendable que el *token* mantenga el mismo tipo de formato que el dato confidencial reemplazado hasta donde sea posible: longitud (14 ó 15 para tarjetas de pago, por ejemplo), codificación (alfabético, numérico, caracteres especiales, ASCII, EBCDIC, etc.) y tipo de dato (char, varchar, blob, int, etc.), teniendo en cuenta que empleando estas restricciones se puede reducir el conjunto de posibilidades de combinación para la generación de *tokens*.

Mantener el mismo formato y longitud del dato confidencial en el *token* ya es en sí una tarea complicada, dependiendo de la forma como se haya generado dicho dispositivo. Para garantizar esta compatibilidad se ha establecido una técnica conocida como *Format Preserving Tokenization* (FPT) que elimina la necesidad de modificar aplicaciones y estructuras de almacenamiento lógicas, definiendo un subconjunto finito de caracteres a ser empleados en los procesos de generación de *tokens*, con lo cual se permite reemplazar el dato confidencial por otro valor con el mismo conjunto de caracteres y longitud.

> Un sistema de 'tokenización' contempla una BD maestra, cifrado e interfaces de comunicación

Este mismo criterio se aplica en el momento del almacenamiento de datos confidenciales cifrados y para la generación de *tokens* pseudoaleatorios empleando técnicas de criptografía. Una función tradicional de cifrado puede tomar una cadena corta de datos y generar una secuencia extensa de caracteres en hexadecimal o en binario, que puede requerir modificaciones en las estructuras de almacenamiento en bases de datos, aplicaciones y en consultas para gestionar esta nueva información con un nuevo formato.

Para solucionar esta situación, se puede hacer uso de *Format Preserving Encryption* (FPE), también conocido como *Datatype Preserving Encryption* (DPE). Estas son funciones criptográficas especiales que –al igual que FPT– permiten que los datos confidenciales tengan la misma estructura y formato que el dato original posterior al proceso de cifrado, ofreciendo altos niveles de seguridad. Un ejemplo de ello es *Feistel Finite Set Encryption Mode* (FFSEM/FFX) **[1]**, una variante de *Advanced Encryption Standard* (AES) que implementa FPE que se ha probado que es matemáticamente segura y que se encuentra en proceso de aprobación por el *National Institute of Standards and Technology* (NIST).

En cuanto al uso de FPE en PCI DSS, Visa ha tenido en cuenta este tipo de algoritmos dentro de sus *Best Practices for Data Field Encryption* **[2]**, aclarando que cualquier método que produzca un dato cifrado con la misma longitud y formato del dato original ha de estar sujeto a, por lo menos, una evaluación de seguridad independiente e implementación con base en las recomendaciones de dicha evaluación (incluyendo los aspectos asociados a la gestión de claves).

Asignación

El proceso de asignación de dispositivos debe asegurar la integridad en las referencias, es decir, que cada *token* debe corresponder única y exclusivamente a un dato confidencial. La existencia de un *token* referenciando a dos o más datos diferentes puede dar pie a problemas en el procesamiento y en el despliegue de información confidencial arbitraria.

Transmisión

En el proceso de la transmisión se identifican dos casuísticas, dependiendo del tipo de dato enviado o recibido:

- **Recepción y salida de *tokens***: Debido a que el *token* no es un dato

confidencial, le aplicarán los mismos controles que protegen a los datos no confidenciales definidos dentro de la organización conforme con la política de clasificación de datos.

- **Recepción (alimentación) y salida (entrega) de datos confidenciales**: Se deben establecer canales seguros para la recepción/entrega de datos confidenciales. En la entrada, el proceso se denomina "alimentación" y consiste en la recepción de datos confidenciales desde los diferentes orígenes posibles, su transmisión segura, su almacenamiento seguro y la relación con su *token* único (generación). En la salida -por condiciones operativas en ciertos escenarios- es indispensable obtener a partir de un *token* el dato confidencial relacionado. Cuando ello sucede, todos los controles de seguridad en la transmisión deben ser activados (cifrado, integridad, monitorización, etc.).

Uno de los pasos iniciales en el proceso de PCI DSS es definir el alcance de cumplimiento: 'scope'

Almacenamiento

Tal como se ha descrito anteriormente, el *token* en sí no es un dato confidencial. Sin embargo, los datos referenciados sí lo son, razón por la cual deben ser protegidos empleando cifrado o cualquier otro método que garantice su confidencialidad, integridad, disponibilidad y trazabilidad. Esto aplica tanto a datos en uso como a copias de seguridad. Es importante también que en el proceso de migración a 'tokenización' se implemente el sistema de *tokens* a datos confidenciales almacenados como históricos, con el fin de evitar su despliegue en consultas asociadas (por ejemplo, en *datawarehousing*).

Eliminación

Finalmente, la plataforma de 'tokenización' deberá controlar que en el proceso de eliminación del dato confidencial posterior a la superación del umbral de retención definido, el *token* sea eliminado, bloqueado o su referencia sea liberada para ser reutilizada si es necesario.

Cuando una organización desea implementar 'tokenización' para proteger su información a nivel interno o la que comparte con terceros, es importante la realización de los siguientes pasos de forma secuencial:

1. Identificación: Revisión de los procesos e interfaces identificando los flujos y activos en donde se encuentren presentes datos confidenciales durante la transmisión (interfaces de conexión, canales, equipos activos de red, etc.), el procesamiento (aplicaciones, esquemas *batch*, *online*, etc.) y el almacenamiento (bases de datos, ficheros de usuario, depuración, históricos, registro de eventos, copias de seguridad, almacenamiento temporal, etc.).

2. Validación: Posterior a la realización del inventario de activos que procesan, almacenan y transmiten datos confidenciales, la organización debe identificar aquellas operativas que requieren del uso exclusivo y justificado de estos datos en texto claro por condiciones del negocio, legales o administrativas y analizar si en las demás operativas el dato confidencial puede ser reemplazado por un *token* sin afectar la operación normal. Esto ayudará a determinar si la reducción en el ámbito donde reside el dato confidencial es práctica y justifica el Retorno de Inversión (*Return On Investment* -ROI-).

3. Análisis de viabilidad: Tras la identificación de dichos flujos, es necesario realizar un análisis de impacto y viabilidad de emplear o no 'tokenización', dependiendo de la cantidad de información confidencial existente, su distribución, su operativa y el riesgo potencial al que se enfrenta, así como los desarrollos necesarios para integrar la solución en el entorno actual. Si se considera que una solución de 'tokenización' no es viable, se puede optar por trabajar con cualquiera de las demás opciones descritas al principio del artículo.

4. Elección de la solución: Para proceder con la implementación, se puede optar por el desarrollo de un sistema de 'tokenización' a nivel interno o de una solución comercial que cumpla con las premisas analizadas anteriormente en el proceso de generación, asignación, almacenamiento y eliminación de *tokens*. De igual forma –y dependiendo del tipo y lugar de almacenamiento de los datos confidenciales–, se puede analizar una opción instalada sobre una base de datos ya existente o una solución tipo *appliance* o equipo dedicado. En ambos casos, los controles de aislamiento deberán ser implementados.

5. Migración: Identificadas las entradas y salidas de los procesos que tratan datos confidenciales, se procede a implementar el sistema de 'tokenización' de forma escalonada con el fin de minimizar los potenciales impactos que puedan tener en la organización y gestionar aquellas excepciones no registradas en el proceso de identificación inicial. Se define un momento N de corte, se empieza con la implementación del proceso en nuevos datos, se continúa con los datos operativos hasta el momento de corte y se finaliza con los datos históricos. Cuanto menor sea el umbral de migración en estos tres núcleos, menor será el riesgo asociado con la información confidencial almacenada, siendo lo más óptimo una migración paralela de toda la información. El objetivo es centralizar los datos confidenciales en un único lugar y gestionar las interfaces de conexión con aplicativos mediante rutinas securizadas de reemplazo por *tokens*.

6. Depuración: Cualquier dato confidencial que, por consideraciones legales u operativas, no deba ser

La plataforma de 'tokenización' deberá controlar que en el proceso de eliminación del dato confidencial posterior a la superación del umbral de retención definido, el 'token' sea eliminado, bloqueado, o su referencia, liberada

almacenado o datos redundantes que deben ser eliminados.

7. Mantenimiento: Después de la puesta en marcha de la solución, se requiere un mantenimiento preventivo y correctivo, ya que la base de datos maestra centralizada se convierte en punto único de fallo de todo el proceso. Por ello, son imprescindibles consideraciones de disponibilidad y monitorización.

Tokenización en PCI DSS

El estándar PCI DSS **[3]** aplica sobre aquellos activos que procesan, almacenan o transmiten el *Primary Account Number* (PAN). Asimismo, prohíbe el almacenamiento de datos sensibles posterior a la autorización (CVV2, Banda magnética, PIN/PINBLOCK) y define controles especiales sobre datos personales del titular de la tarjeta, código de servicio y fechas de expiración cuando esta información es almacenada en conjunto con el PAN.

Uno de los pasos iniciales en el proceso de implementación de los controles de PCI DSS es definir el alcance de cumplimiento (*scope*), que permite identificar cualquier activo relacionado de forma directa con el tratamiento de datos de tarjeta o que ofrece servicios a dicha plataforma. Estos activos deben cumplir de forma completa con todos los controles descritos por el estándar con el fin de proteger los datos confidenciales que fluyen a través de él.

En una organización compleja, la implementación de PCI DSS es una labor larga y costosa, debido a la gran cantidad de aplicaciones, equipos activos de red, bases de datos, ficheros y plataformas operativas involucradas. Al ser el dato de PAN la clave en muchos procesos operativos, los riesgos son incontables: almacenamiento no autorizado en ficheros temporales, de depuración, históricos, registros, correos electrónicos, copias impresas, almacenamiento removible, imágenes, etc.

Frente a esto, el PCI *Security Standards Council* (PCI SSC) **[4]** recomienda el uso de segmentación y aislamiento de los activos dentro del *scope* para minimizar el riesgo y el impacto en el cumplimiento y facilitar las labores de implementación y mantenimiento. De no realizar una segmentación correcta, toda la red se encontrará dentro del alcance (red plana). Es precisamente en este punto en el que las funcionalidades de la 'tokenización' proveen beneficios en el ámbito de PCI DSS: reemplazar el dato del *Personal Account Number* (PAN) con un *token* excluye de forma directa a cualquier activo que almacene, procese o trasmita dicho dato 'tokenizado'. La reducción del alcance –dependiendo de la complejidad de la organización– es drástica y permite delimitar e identificar puntualmente los activos e interfaces por los cuales fluye el PAN y, por lo tanto, son susceptibles de cumplir con PCI DSS. Otra alternativa puede ser que, en vez de reemplazar el PAN, se 'tokenicen' aquellas transacciones que hagan referencia a este dato.

PCI DSS contempla los procesos de 'tokenización' dentro del requerimiento 3.4 de la normativa ("Haga que el PAN quede, como mínimo, ilegible en cualquier lugar donde se almacene..."). Es también recomendable que los controles de enmascaramiento (reemplazo por asteriscos, por ejemplo) se sigan implementando independientemente de que se trate de un PAN real o un PAN 'tokenizado', siguiendo las directrices del requerimiento 3.3 ("Oculte el PAN cuando aparezca -los primeros seis y los últimos cuatro dígitos es la cantidad máxima de dígitos que aparecerá-…").

El proceso de 'tokenización' del PAN puede ser completo (reemplazo total del PAN por un *token*) o parcial (dejando los primeros seis y últimos cuatro dígitos iguales y reemplazando por un *token* los datos intermedios, por ejemplo), dependiendo de la operativa de la organización (ver Figura 2).

Identificadas las entradas, salidas y operativas que requieren del PAN en texto claro (por ejemplo, procesos de fraude o atención a incidencias), se definen interfaces de comunicación seguras y se procede con la centralización de datos confidenciales y asignación de *tokens*. Cualquier activo que procese, almacene o transmita estos dispositivos se encontrará fuera del alcance de cumplimiento, con todas las ventajas que ello provee: minimi-

Figura 2: *Proceso de 'tokenización' parcial de PAN (Personal Account Number), que consiste en dejar los primeros seis y cuatro últimos dígitos iguales, reemplazando por un token los datos intermedios.*

zación del riesgo, ahorro en costes de implementación y mantenimiento, etc. (Ver Figura 3).

Por otra parte, dentro del proceso de implementación y migración hacia 'tokenización' existirá un punto de convivencia de PAN reales y 'tokenizados'. Para distinguir los unos de los otros se puede hacer uso del algoritmo de Luhn **[5]**. Cualquier PAN válido cumplirá con este algoritmo, por lo que se puede configurar que la función de generación de *tokens* produzca valores que no cumplan con esta condición, facilitando las labores de identificación de datos y evitar la "contaminación" de datos reales con datos 'tokenizados' sin control.

Dentro de la apuesta por la 'tokenización' que realizan las marcas de pago, Visa ha publicado una serie de mejores prácticas para 'tokenización' ("*Best practices for the tokenisation of cardholder information*") **[6]**, que sirven como guía a cualquier organización que se esté planteando implementar este sistema dentro de sus procesos internos como soporte a la normativa PCI DSS.

Mayor control

A pesar que el concepto de "tokenización" en el ámbito de seguridad de la información es bastante reciente, por necesidades propias o por cumplimiento de normativas se está empezando a implementar en ambientes que requieren protección de información confidencial. El reemplazo de un dato confidencial por otro no confidencial que garantice la misma operativa minimiza el riesgo y los costes de implementación y mantenimiento de controles y restringe el ámbito afectado. Empleando técnicas de conservación de formato de datos en subrutinas de cifrado (FPE) y 'tokenización' (FPT), se puede mantener el mismo formato y longitud del dato confidencial en el *token* haciendo transparente la migración y evitando cambios de infraestructura.

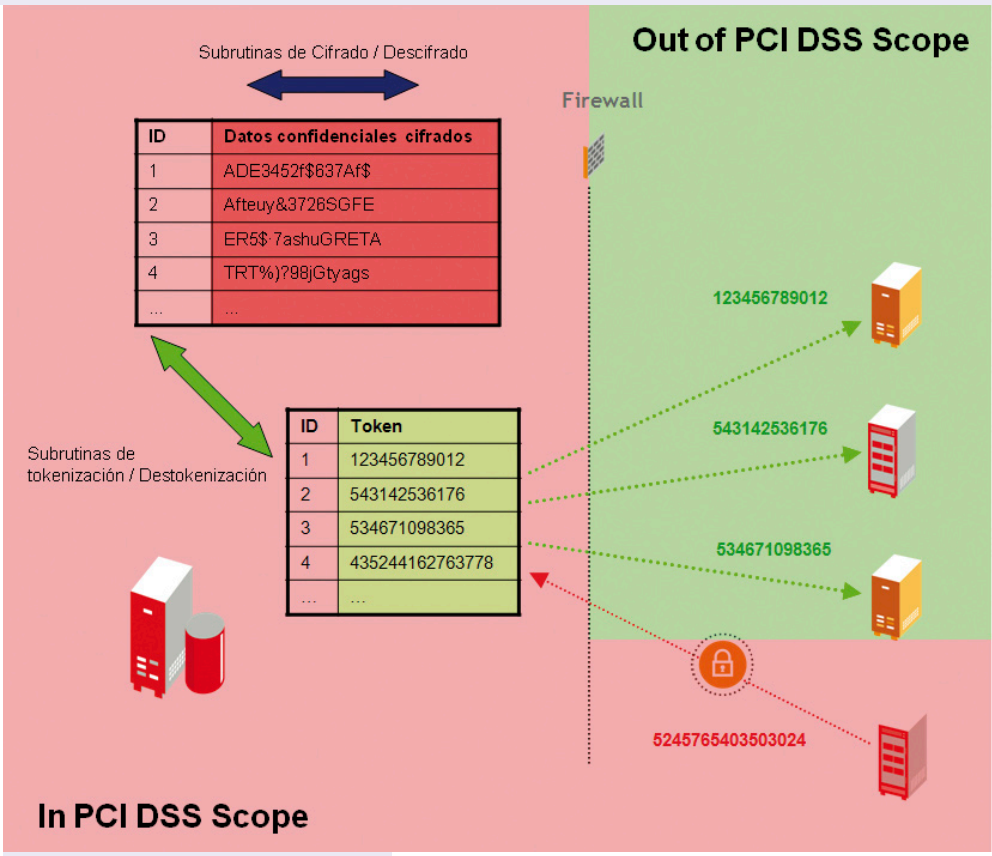

Figura 3: *Cualquier activo que procese o transmita tokens se encontrará fuera del alcance de cumplimiento, con todas las ventajas que ello provee, como la minimización del riesgo o el ahorro en costes de implementación.*

Sin embargo, la base de datos maestra centralizada de *tokens* se convierte en un punto único de fallo, que requiere de protección robusta y de disponibilidad con el fin de evitar potenciales problemas dentro de la organización.

Con el uso de esta técnica, estándares como PCI DSS (orientado a protección de datos en tarjetas de pago), LOPD, Ley de Portabilidad y Responsabilidad del Seguro Médico (HIPPA, por sus siglas en inglés), y otros, minimizan su alcance y el riesgo relacionado con la información que debe ser ser protegida.

David Eduardo Acosta Rodríguez *es ingeniero de Sistemas de la Universidad Distrital "Francisco José de Caldas" en Bogotá D.C. (Colombia). Màster en Seguretat de les Tecnologies de la Informació (MSTI) y Master in Project Management (MPM) de la Universidad de La Salle–Ramón Llull (Barcelona, España). Cuenta con las certificaciones CISSP, CISM, CISA, BS25999 Lead Auditor, CHFI, OPST, CCNA y PCI QSA. Es miembro de la IEEE, de ISMS Forum Spain y trabaja actualmente como consultor senior en Seguridad de la Información en IsecAuditors. Las referencias que el autor destaca en este artículo son las siguientes:*

[1] *The FFX Mode of Operation for Format-Preserving Encryption http://csrc.nist.gov/groups/ST/toolkit/BCM/documents/proposedmodes/ffx/ffx-spec.pdf. Febrero de 2010.*

[2] *VISA Best Practices: Data Field Encryption Version 1.0 http://corporate.visa.com/_media/best-practices.pdf. 5 de octubre de 2009.*

[3] *PCI Data Security Standard http://es.pcisecuritystandards.org/minisite/en/pci-dss.html. Septiembre de 2010.*

[4] *PCI Security Standards Council http://es.pcisecuritystandards.org/minisite/en/index.html. Septiembre de 2010.*

[5] *Luhn algorithm http://en.wikipedia.org/wiki/Luhn_algorithm. Septiembre de 2010.*

[6] *PCI Security Standards Council "Best practices for the tokenisation of cardholder information" http://usa.visa.com/download/merchants/tokenization_best_practices.pdf?13 de julio de 2010. http://www.visaeurope.com/en/businesses__retailers/payment_security/idoc.ashx?docid=9706b57b-9238-4370-8dca-838813e9cbf1&version=-1. Septiembre de 2010.*