

# GENERACIÓN DE TOKENS PARA PROTEGER LOS DATOS DE TARJETAS BANCARIAS

TRABAJO TERMINAL No. 2017-B008

PRESENTAN

DANIEL AYALA ZAMORANO

DAZ23AYALA@GMAIL.COM

LAURA NATALIA BORBOLLA PALACIOS

LN.BORBOLLA42@GMAIL.COM

RICARDO QUEZADA FIGUEROA

QF7.RICARDO@GMAIL.COM

DIRECTORA

DRA. SANDRA DÍAZ SANTIAGO

CIUDAD DE MÉXICO, 9 DE MAYO DE 2018

ESCUELA SUPERIOR DE CÓMPUTO  
INSTITUTO POLITÉCNICO NACIONAL



# CONTENIDO

Planteamiento del problema

# CONTENIDO

Planteamiento del problema

Planteamiento de la solución

# CONTENIDO

Planteamiento del problema

Planteamiento de la solución

Algoritmos generadores de *tokens*

# CONTENIDO

Planteamiento del problema

Planteamiento de la solución

Algoritmos generadores de *tokens*

Trabajo a futuro

# CONTENIDO

Planteamiento del problema

Planteamiento de la solución

Algoritmos generadores de *tokens*

Trabajo a futuro

# UN INICIO TORMENTOSO

- ▶ En la década de los 80 y 90, el comercio en línea comenzó a crecer y tomar importancia.
- ▶ Las empresas no estaban preparadas para el impacto que tuvieron y los fraudes relacionados con el comercio electrónico aumentaron rápidamente.
  - ▶ Visa y Mastercard reportaron, entre 1988 y 1998, pérdidas de 750 millones de dólares.
  - ▶ En 2001, se reportaron pérdidas de 1.7 miles de millones de dólares. y 2.1 miles de millones de dólares al año siguiente.

# UN ESTÁNDAR PARA GOBERNARLOS A TODOS

- ▶ A inicios del 2000, las grandes compañías (¿emisoras de tarjetas?) comenzaron a publicar, individualmente, *buenas prácticas* de seguridad.
- ▶ Las empresas intentaron adoptar las prácticas, pero era tremendamente complicado y costoso.
- ▶ Se aliaron las compañías y, en 2004, publicaron un estándar unificado: PCI-DSS (Payment Card Industry - Data Security Standard).
  - ▶ Se hizo obligatorio para quienes realizasen más de 20K transacciones al año.



# CAMBIO DE ESTRATEGIA

- ▶ Hasta ahora, el enfoque era proteger los datos sensibles donde sea que se encuentren y por donde sea que transiten.
- ▶ Surge un nuevo enfoque: cambiar la información valiosa, por *valores representativos* (tokens); es decir, la tokenización de la información.
- ▶ En 2011, el PCI-SSC publicó las primeras guías para los procesos de tokenización.
  - ▶ Aunque indica lo que debe satisfacer el sistema tokenizador, no dice cómo generar los tokens.
  - ▶ Tiene un gran número de requerimientos (y subrequerimientos), por lo que es difícil de satisfacer.

# PERO ¿POR QUÉ?

A pesar de ser una práctica extendida, la tokenización sigue estando rodeada de desinformación y desconfianza.

- ▶ Se busca combatir la desinformación al estudiar e implementar cinco algoritmos tokenizadores, compararlos y mostrar los resultados.
- ▶ Hacer notar que la criptografía y la tokenización no están peleadas; pues la tokenización puede verse como una aplicación de la criptografía.

# CONTENIDO

Planteamiento del problema

Planteamiento de la solución

Objetivos del proyecto	9
Metodología del proyecto	10
Prototipos	11
Prototipo de generación de tokens	12
Especificaciones técnicas del desarrollo	13

Algoritmos generadores de *tokens*

Trabajo a futuro

# OBJETIVOS DEL PROYECTO

Lo que se busca con este proyecto es implementar un programa generador de *tokens* que provea confidencialidad a los datos de las tarjetas bancarias.

Además, con el afán de disminuir la desinformación existente sobre la tokenización, se busca obtener una comparativa de los algoritmos implementados.




# METODOLOGÍA DEL PROYECTO

Para el desarrollo de este proyecto se está usando una metodología de prototipos, en la que cada prototipo usa como metodología interna SDL (*Security Development Lifecycle*).

SDL es una metodología especializada para software de seguridad desarrollada por Microsoft, que se caracteriza por continuas fases de verificación de seguridad y una primera etapa de estudio de los temas acordes al proyecto.

# PROTOTIPOS

Este proyecto está dividido en 3 prototipos, los cuales son:

 <b>Prototipo de generación de tokens. ✓</b>	 <b>Prototipo de servicio Web.</b>	 <b>Prototipo de tienda en línea.</b>
<p>Revisar e implementar diversos algoritmos generadores de tokens para hacer un programa tokenizador, así como realizar pruebas comparativas entre estos algoritmos.</p>	<p>Diseñar e implementar una API web capaz de comunicar al programa tokenizador con al menos una tienda en línea con el fin ofrecer el servicio de tokenización.</p>	<p>Implementar una tienda en línea que utilice la API web para poder revisar el correcto funcionamiento del servicio.</p>

## Prototipos del trabajo terminal.

# PROTOTIPO DE GENERACIÓN DE TOKENS

En TT1 se planeó terminar el primer prototipo dado que es la parte central del proyecto; situación que se consiguió, llegando a implementar 5 algoritmos distintos y realizar pruebas comparativas entre estos. Además, en los algoritmos pertinentes, se realizaron pruebas de aleatoriedad con el fin de respaldar la seguridad de la implementación.

# ESPECIFICACIONES TÉCNICAS DEL DESARROLLO

La implementación de los algoritmos generadores de *tokens* se hizo en lenguaje C++, dado que combina mantenibilidad y rendimiento.

Para las implementaciones que hacen uso de una base de datos, se utilizó el gestor MariaDB, el cual es una bifurcación de MySQL.



# CONTENIDO

Planteamiento del problema

Planteamiento de la solución

Algoritmos generadores de *tokens*

Clasificación 15

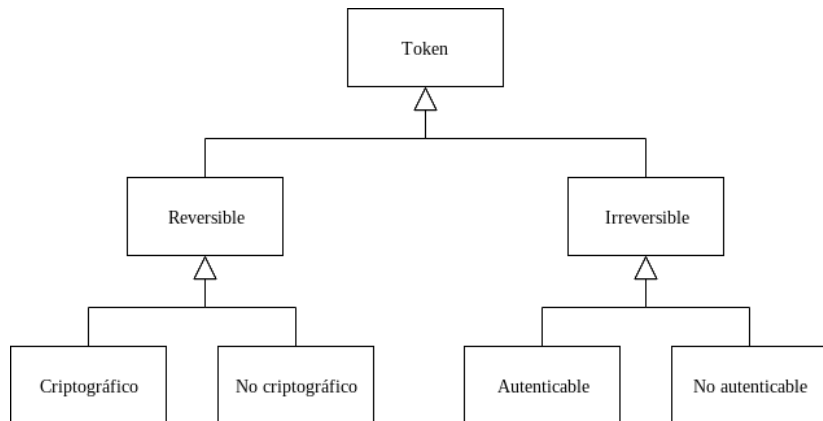
Métodos reversibles 18

Métodos irreversibles 18

Comparación de desempeño 18

Trabajo a futuro

# CLASIFICACIÓN DEL PCI SSC



Clasificación de algoritmos tokenizadores [1].

## Generación de tokens para proteger los datos de tarjetas bancarias

- Algoritmos generadores de *tokens*

- Clasificación

- Clasificación del PCI SSC



Los irreversibles no pueden ser reconvertidos al PAN (de ninguna manera, mas que con fuerza bruta). Los autenticables funcionan como una función Hash: si tienes el PAN y el token, se puede validar que ese token es el par de ese PAN. Los no autenticables no pueden validar esto último.

Los reversibles permiten obtener el PAN a partir del token. Los no criptográficos ocupan funciones pseudoaleatorias y una base de datos para guardar las relaciones PAN-token. Los criptográficos ocupan un esquema de cifrado tradicional: un PAN mas una llave permiten obtener un token; la llave y el token pueden ser ocupados para obtener el PAN. No se ocupa una base de datos.

# CLASIFICACIÓN DEL PCI SSC

¿«No criptográficos»?

La clasificación anterior presenta los siguientes problemas:

- Los casos de uso que el PCI SSC prevé en [1] para los irreversibles resultan artificiosos.

# Generación de tokens para proteger los datos de tarjetas bancarias

- └ Algoritmos generadores de *tokens*
  - └ Clasificación
    - └ Clasificación del PCI SSC

La clasificación anterior presenta los siguientes problemas:

- Los casos de uso que el PCI SSC prevé en [1] para los irreversibles resultan artificiales.

Por ejemplo, la justificación para los no autenticables es para dar soporte a aplicaciones obsoletas que necesitan un formato de PAN válido. Esto se puede lograr con los no criptográficos sin guardar nada en la base; o pasando puros ceros en el campo del PAN.

El caso para los autenticables permite verificar la tarjeta del cliente en una compra cuando este perdió el comprobante. En est caso no resulta claro por qué la tienda (o el sistema tokenizador) no guardaría la transacción original.

# CLASIFICACIÓN DEL PCI SSC

¿«No criptográficos»?

La clasificación anterior presenta los siguientes problemas:

- ▶ Los casos de uso que el PCI SSC prevé en [1] para los irreversibles resultan artificiosos.
- ▶ A pesar del nombre, los *no criptográficos* ocupan diversas aplicaciones de la criptografía para generar *tokens*.

# Generación de tokens para proteger los datos de tarjetas bancarias

- └ Algoritmos generadores de *tokens*

- └ Clasificación

- └ Clasificación del PCI SSC

La clasificación anterior presenta los siguientes problemas:

- Los casos de uso que el PCI SSC prevé en [1] para los irreversibles resultan artificiales.
- A pesar del nombre, los *no criptográficos* ocupan diversas aplicaciones de la criptografía para generar tokens.

El problema con el PCI es que parecen pensar que la criptografía se limita a esquemas tradicionales, en donde hay una llave. La generación de números pseudoaleatorios seguros es también una aplicación de la criptografía.

# CLASIFICACIÓN PROPUESTA



# CONTENIDO

Planteamiento del problema

Planteamiento de la solución

Algoritmos generadores de *tokens*

Trabajo a futuro



# BIBLIOGRAFÍA I



Payment Card Industry Security Standards Council.  
*Tokenization Product Security Guidelines – Irreversible  
and Reversible Tokens*. 2015. URL:  
[https://www.pcisecuritystandards.org/documents/  
Tokenization\\_Product\\_Security\\_Guidelines.pdf](https://www.pcisecuritystandards.org/documents/Tokenization_Product_Security_Guidelines.pdf)  
(vid. págs. 18, 20, 21).

# GENERACIÓN DE TOKENS PARA PROTEGER LOS DATOS DE TARJETAS BANCARIAS

TRABAJO TERMINAL No. 2017-B008

PRESENTAN

DANIEL AYALA ZAMORANO

DAZ23AYALA@GMAIL.COM

LAURA NATALIA BORBOLLA PALACIOS

LN.BORBOLLA42@GMAIL.COM

RICARDO QUEZADA FIGUEROA

QF7.RICARDO@GMAIL.COM

DIRECTORA

DRA. SANDRA DÍAZ SANTIAGO

CIUDAD DE MÉXICO, 9 DE MAYO DE 2018

ESCUELA SUPERIOR DE CÓMPUTO  
INSTITUTO POLITÉCNICO NACIONAL

