# Significance of Tokenization in Promoting Cloud Based Secure Elements

Busra Ozdenizci[1], Vedat Coskun[1*], Kerem Ok[1] and Turgay Karlidere[2]
[1]NFC Lab - Istanbul, Department of Information Technologies, ISIK University, Sile, Istanbul, Turkey
[2]KocSistem Information and Communication Services Inc., Istanbul, Turkey
*Corresponding Author: vedatcoskun@isikun.edu.tr

## ABSTRACT

As a promising rather new technology, Near Field Communication (NFC) gained further appreciation with the recent advances in Cloud based Secure Element (SE). After noticing the deficiencies of existing SE alternatives, Host Card Emulation (HCE) has gained approval to provide storing, accessing, and managing private and sensitive data on the cloud instead of on the Smartphones; eventually promoting Cloud based HCE. This paper presents the significance of Tokenization method in Cloud based HCE services in terms of both technical and business aspects. In this context, a novel generic usage model is proposed for diverse HCE based NFC services such as access control, security, identification, and loyalty. Security and communication issues of the proposed model are further studied by considering major Tokenization standards and specifications.

**Keyword:** Near Field Communication, NFC, Host Card Emulation, HCE, Cloud, Secure Element.

## 1. Introduction

Near Field Communication (NFC) technology is a short-range half duplex communication protocol, which was jointly developed by Philips and Sony in late 2002. NFC is compatible with the existing infrastructure of RFID (Radio Frequency Identification) technology and contactless ISO/IEC 14443 contactless smart cards which occurs between two NFC compatible devices within few centimeters with 13.56 MHz operating frequency (Coskun, Ok, & Ozdenizci, 2012). It provides easy communication between various NFC devices on ISO/IEC 18000-3 air interface, at transfer rates ranging from 106 to 424 Kbits per second.

NFC is a new promising short-range wireless communication technology that gained appreciation as a significant contributor of several technologies such as Internet of Things (IoT), Ubiquitous Computing (Ubicomp), and Cloud Computing (Coskun, Ozdenizci, & Ok, 2015). The importance of NFC technology comes from its ease of use for triggering the functionality and seamlessly enabling a secure communication in the meanwhile. In order to engage in an NFC communication, the user needs to touch her NFC Smartphone to either an NFC tag, another NFC Smartphone, or an NFC reader.

When NFC Smartphone is touched to an NFC tag, Smartphone reads/writes data from/to an NFC tag; when touched to another NFC Smartphone, they exchange data; and when touched to an NFC reader, the reader reads the valuable and private data stored on Smartphone. An operating mode name is given to each interaction: reader/writer mode to the tag interaction, peer-to-peer mode to the Smartphone interaction, and card emulation mode to the reader interaction, as illustrated in Figure 1 (Coskun, Ozdenizci, & Ok, 2015).

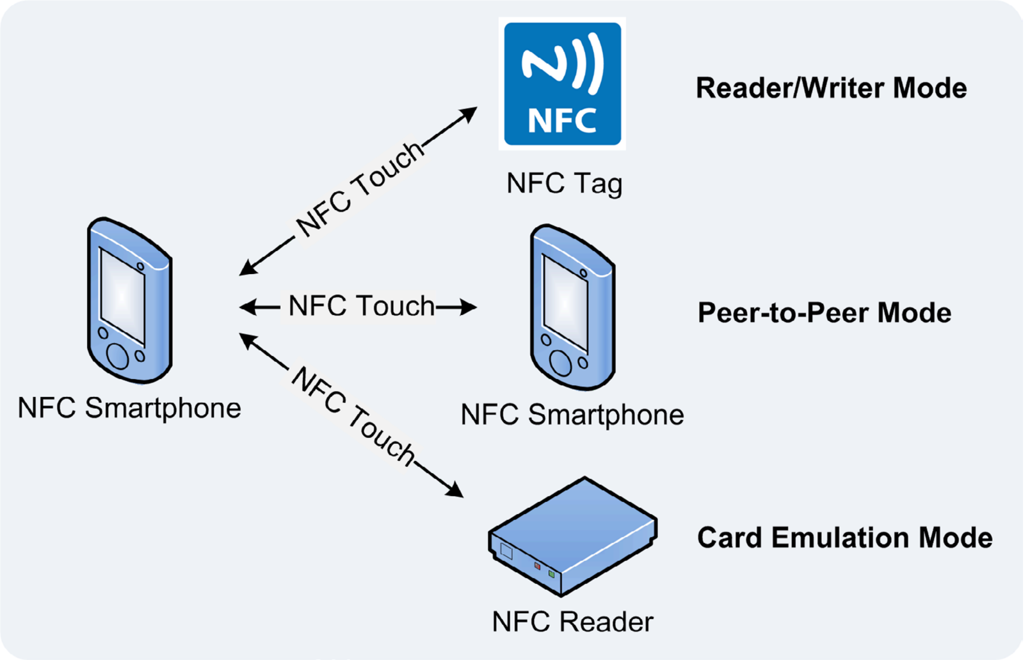

Figure 1. NFC Interactions and Operating Modes

The most promising NFC operating mode is card emulation, which enables an NFC Smartphone to behave as a contactless smart card. Card emulation mode enables realization of diverse applications such as mobile payment, ticketing, coupon, loyalty, access control, identification and so on.

In this mode, SE is the most important part of NFC Smartphones for securing the private data and mobile application executable code. According to GlobalPlatform, SE is "a tamper-resistant platform (typically a one-chip secure microcontroller) capable of securely hosting applications and their confidential and cryptographic data (e.g., key management) in accordance with the rules and security requirements set forth by a set of well-identified trusted authorities".

Up to now, several hardware based SEs including Universal Integrated Circuit Cards (UICC), embedded SEs, and SD based SEs are emerged for enabling –secure– card emulation services. However, several technical and business limitations have already been observed; and therefore offering further efficient SE alternatives for storing private data have become an important issue nowadays (Coskun, Ozdenizci, & Ok, 2015).

Cloud based SE concept emerged after the introduction of HCE (Host Card Emulation) technology in Android 4.4 (KitKat) OS by Google, which separates the card emulation functionality from the SE (Smart Card Alliance, 2014). HCE technology enables storing, accessing and managing private data on the cloud instead of on the Smartphones. The Smartphone still performs card emulation functions but the private data is stored, secured, and accessed on the cloud.

This paper aims to present the significance of Tokenization method in Cloud based HCE services in terms of technical and business aspects, and proposes a novel generic usage model which can be applied in diverse HCE based NFC services such as access control, identification, and loyalty. Furthermore, the security and communication issues of proposed model are described depending on major Tokenization standards and specifications.

## 2. Research Background

### 2.1. HCE Technology

HCE can be referred as a Software based SE, where data is stored and managed on the Cloud; whereas HCE functionality is located in libraries and APIs of mobile OS (Operating System), and these libraries and APIs are used by the application running on the host CPU as depicted in Figure 2 (Alattar & Achemlal, 2014). So, the mentioned application becomes able to exchange APDU commands with an NFC reader. HCE support is currently available on the Android OS (Android KitKat 4.4 and higher) and the BlackBerry OS.

The motivation behind HCE technology is its independence from hardware based SE alternatives. In case of hardware based SEs, the APDU commands coming from the

NFC reader are passed to the application on SE of NFC Smartphone with the help of NFC controller, so that SE handles the APDU commands in order to emulate a contactless card securely (Smart Card Alliance, 2014).

In case of HCE, received APDU commands are passed to the active NFC application by the NFC controller. On the contrary, HCE technology eliminates the need for a hardware based SE; and the private data is stored on a remote server as the Cloud. Hybrid solutions those include both SE and HCE technologies are also proposed (Smart Card Alliance, 2014). Figure 2 illustrates the communication flow of SE, HCE, as well as hybrid solutions.

As the computing / storage capacity and development complexity are considered, HCE based NFC services are more advantageous over hardware based SE (Alattar & Achemlal, 2014). Moreover, in terms of NFC ecosystem and business models, HCE based solutions are independent of mobile network operators, service providers, and trusted service managers; hence HCE technology can be considered as a game changer (Mobey Forum, 2014).

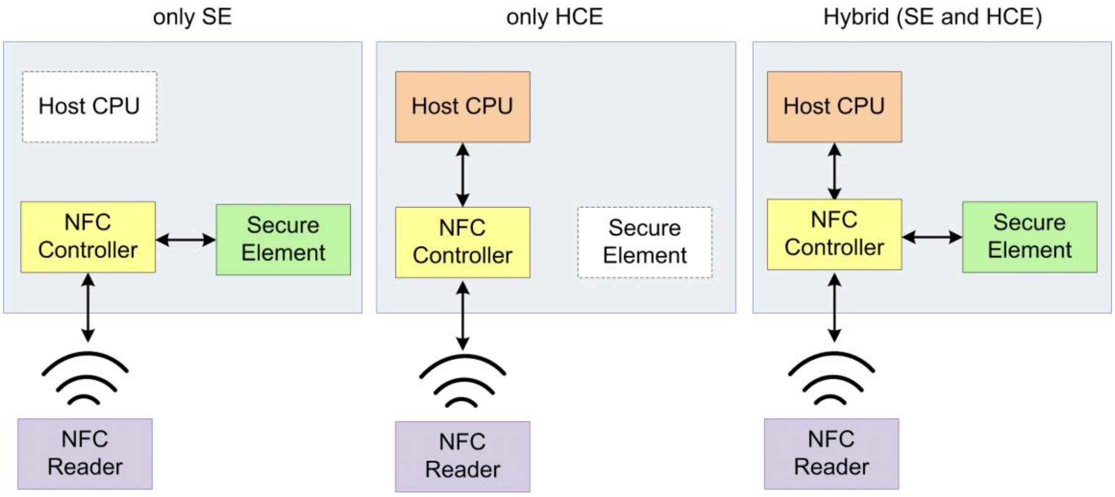

Figure 2. SE and HCE Solutions on NFC Smartphone

Among other business case alternatives, HCE technology is mostly supported on NFC based mobile payment systems over the world; important current HCE implementations are Google Wallet in US, Tim Hortons in Canada, and BBVA in Spain (Smart Card Alliance, 2014).

### 2.2. HCE Solution Alternatives

There exist two methods for performing HCE services: Full Cloud based HCE solution and Tokenization based HCE solution (Mobey Forum, 2014):

(1) *Full Cloud based HCE solution:* Card emulation is performed completely on the cloud. The mobile application on NFC Smartphone authenticates the user and enables the secure connection to the remote server (Figure 3).

NFC Smartphone aiming to obtain the credentials on the cloud needs to connect to the remote server repeatedly for each distinct transaction. As a matter of fact, this solution requires rather fast 4G or even 5G networks, which creates a network bandwidth and security limitations (Mobey Forum, 2014).

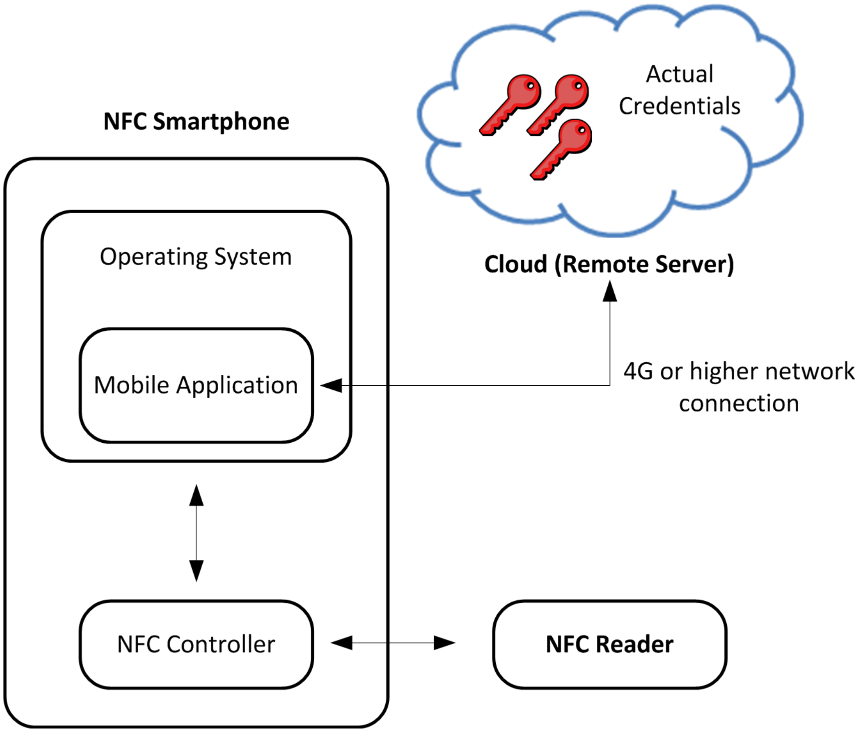

Figure 3. Full Cloud Based HCE Solution

(2) *Tokenization based HCE solution:* To mitigate technical limitations and security risks, a second option is emerged that is based on Tokenization. Tokenization opens up the possibility of enabling more secure and efficient offline transactions.

Tokenization replaces the actual data exchange by a token, which is a disguised representation of the original value (Mobey Forum, 2014; PCI DSS, 2011). Threats via brute force attack to the Tokens can be prevented by several methods such as limiting the number of transactions or limiting the validity time.

For each transaction, the mobile application on the Smartphone sends token value to NFC reader, and Service Provider of the NFC reader sends token to Token Service Provider (TSP) for getting the actual credential; after which the Service Provider may authorize the transaction (Figure 4).

The card emulation is performed by the mobile application on NFC Smartphone; there is no need for the NFC Smartphone to access to the cloud; transactions are completely based on tokens in this solution, providing more secure communication.

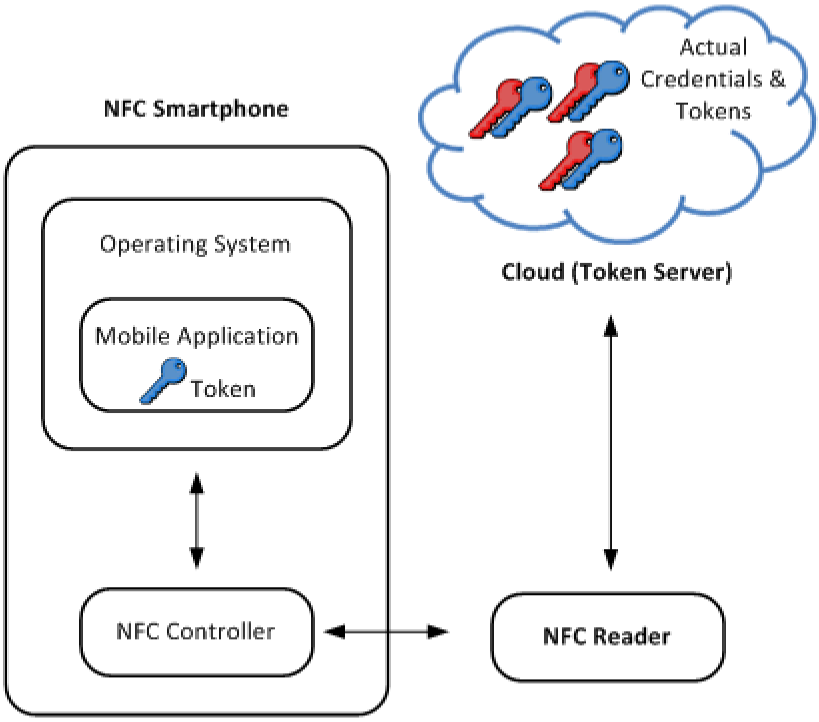

Figure 4. Tokenization based HCE Solutions

### 2.2. Tokenization

ASC X9 (Accredited Standards Committee X9), PCI DSS (Payment Card Industry Data Security Standard), Visa, and EMVCo (EuroPay, MasterCard, Visa) have several efforts to develop standards and specifications for Tokenization.

Among others, PCI DSS provided the most considerable standard on Tokenization systems for payment industry stakeholders (PCI DSS, 2011). According to the provided PCI DSS standard, Tokenization systems have four common components:

(1) Token Generation: The process of creating a token can be implemented by using several methods such as mathematically reversible cryptographic function based on strong cryptographic algorithms and keys, one-way non-reversible cryptographic functions (e.g., a hash function with strong, secret salt), or assignment through a randomly generated number. Token is referred as PAN (Primary Account Number) in payment systems.
(2) Token Mapping: The generated token further needs to be assigned to its original value.
(3) Card Data Vault: A central repository is required for storing original values and corresponding tokens.
(4) Cryptographic Key Management: The process for creating, using, managing, and protecting cryptographic keys is also an important issue; token needs to be protected with these keys on data vault.

EMVCo has issued a Tokenization framework to describe the requirements for creation and use of payment tokens in the context of digital transactions (EMVCo, 2014). The framework introduces a third party entity called Token Service Provider (TSP) that generates and resolves tokens.

## 3. Generic Model for HCE based NFC Services

In accordance with the PCI DSS and EMVCo standards, we propose a novel generic usage model for HCE based non-payments NFC services including loyalty and couponing, access control, identification and security applications. The proposed model aims to provide secure data service on the cloud by also promoting HCE based NFC services. Valuable data of the NFC Smartphone users are securely stored on a remote server of trusted entity.

### 3.1. Actors and Roles

Our proposed model includes three main actors (Figure 5):
(1) NFC Smartphone users who need to own HCE enabled Smartphones,
(2) Service Provider that supplies HCE enabled NFC service(s),
(3) TSP that provides token generation, secure data service, and token mapping processes.

As the NFC Smartphone stores some data on the server of the TSP, the corresponding token is replied by the TSP which is then stored on the NFC Smartphone. For each NFC transaction, Service Provider requests authorization from TSP for the token value received from NFC Smartphone. TSP performs token mapping (i.e., de-tokenization) process, and then sends an authorization response including the original data to Service Provider in case of affirmative evaluation result.

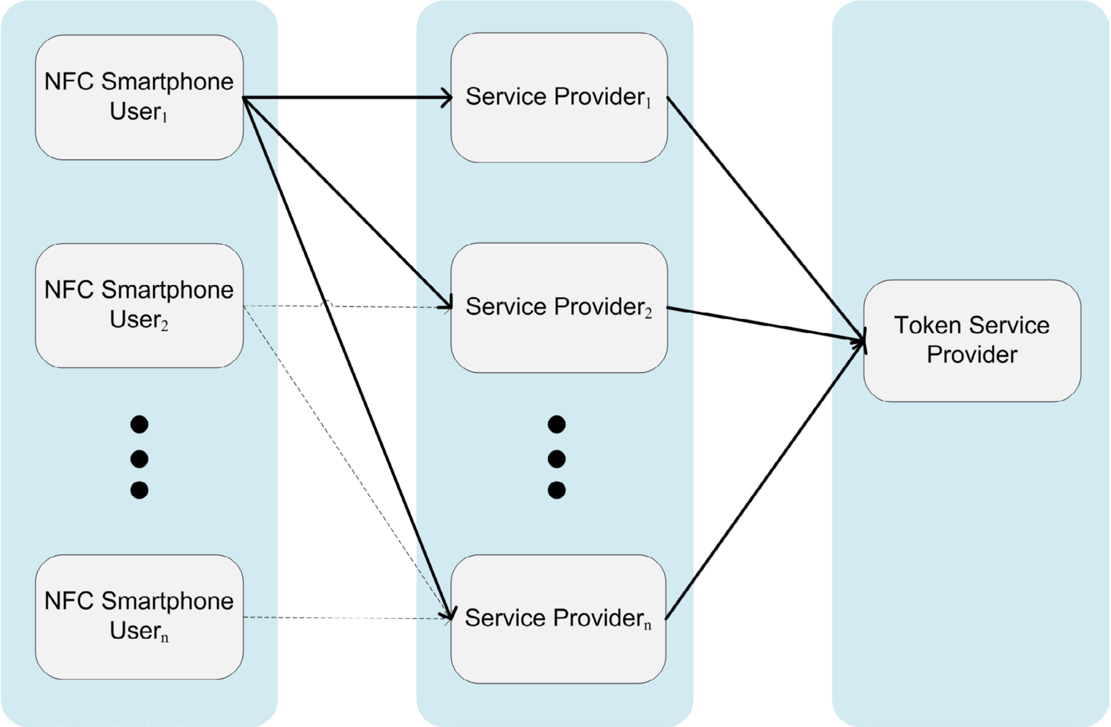

Figure 5. Conceptual Model of System Actors

### 3.2. Communication and Security Issues

The communication model uses a two-phased Tokenization for providing secure interaction between entities. In this context, the following security and communication issues are considered for the proposed model:

(1) *User Identity Management:* The proposed model aims to provide user authentication to the system. NFC Smartphone stores a token value as *userToken* that refers to the user's identity data (i.e., ID, first name, last name, etc.). The *userToken* value is generated by TSP and transferred on the user's Smartphone. The identity data of the user and *userToken* values are securely stored on the data server of the TSP; so that unauthorized parties cannot access them. Table 1 presents an example of *userToken* table on data server.

(2) *Application Identity Management:* Another important communication and security issue is the identity management of Service Provider's application. Each Service Provider may have one or more HCE enabled NFC services. To distinguish and identify applications of all Service Providers in this centralized

model, a token value for each application –being *appToken*– is used. The *appToken* value is generated by the TSP and uploaded to the Service Provider's backend system. Table 2 presents an example of *appToken* table on data server.

Table 1. *userToken* Table Example

| userToken | ID | First Name | Last Name |
|---|---|---|---|
| user1Token | 1001 | Vedat | Coskun |
| user2Token | 1002 | Busra | Ozdenizci |
| user3Token | 1003 | Kerem | Ok |

Table 2. *appToken* Table Example

| appToken | Company | Application Name |
|---|---|---|
| app1Token | X Company | Access Control |
| app2Token | X Company | ID Card |
| app3Token | Y Company | Membership Card |
| app4Token | Z Company | Loyalty Application |

(3) *User Token and Application Token Matrix Management:* The private data of the NFC Smartphone user are stored and managed on the data server of the TSP as illustrated in Table 3.

Table 3. *userToken* and *appToken* Matrix Table Example

| | appToken | | | | |
|---|---|---|---|---|---|
| userToken | app1Token | app2Token | app3Token | app4Token | ... |
| user1Token | 1010 | Null | Null | Null | ... |
| user2Token | 1020 | Null | Null | 4959 0059<br>0172 3389 | ... |
| user3Token | Null | 1111 | Null | 4959 0059<br>9375 3390 | ... |

### 3.3. Generic Usage Model

The usage model of the proposed scheme is illustrated in Figure 6 and further described step by step hereunder:

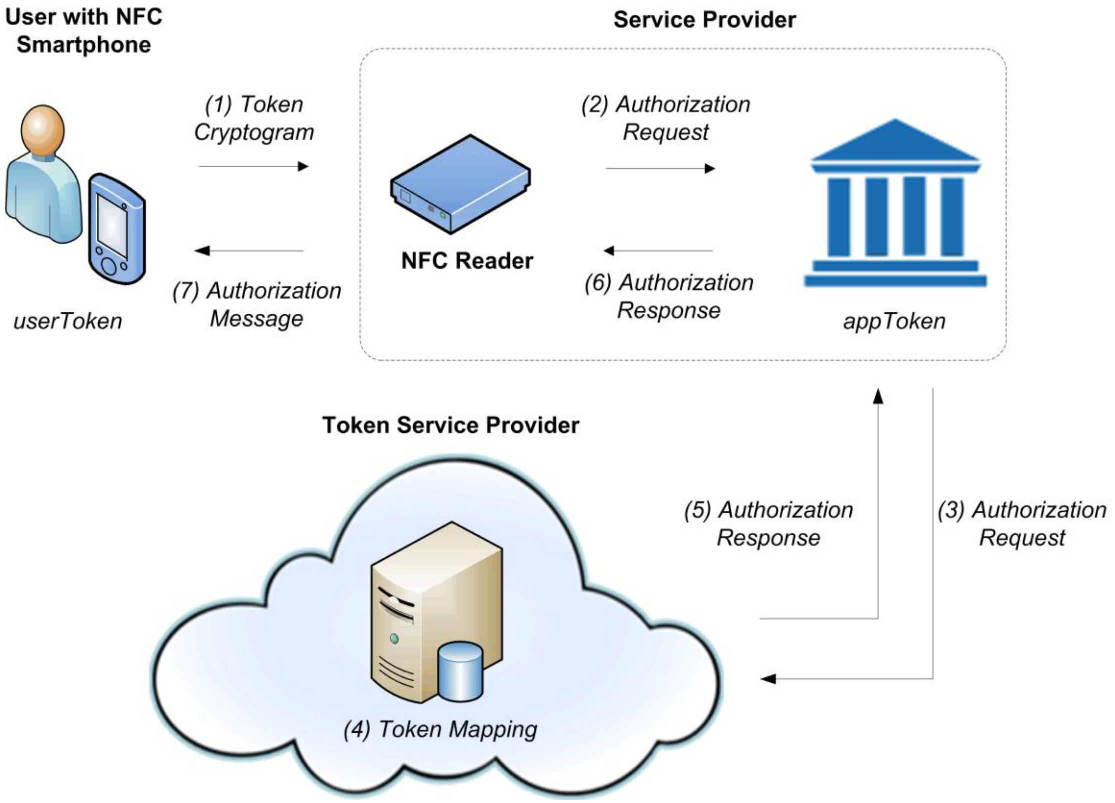
User with NFC Smartphone
Service Provider
(1) Token Cryptogram
(2) Authorization Request
NFC Reader
(7) Authorization Message
(6) Authorization Response
userToken
appToken
Token Service Provider
(5) Authorization Response
(3) Authorization Request
(4) Token Mapping

Figure 6. Generic Usage Model

Step (1): NFC Smartphone user first touches an NFC reader of Service Provider (e.g., an access or security point, loyalty POS terminal and so on). NFC reader requests identity of the user; after which *userToken* value on the NFC Smartphone is sent to NFC reader.

Step (2): NFC reader passes the *userToken* value to its backend system.

Step (3): Service Provider concatenates the corresponding application's *appToken* value with the *userToken* value coming from NFC reader, and then performs authorization request from TSP.

Step (4): TSP performs token mapping process of *userToken* and *appToken* values, and obtains the private data of the user from the data server.

Step (5): TSP sends an authorization response together with the secured data to Service Provider.

Step (6): Service Provider transfers the authorization response to its NFC reader.

Step (7): NFC reader sends a verification / authorization message to the NFC Smartphone of user.

## 4. Conclusion

Cloud based SE for NFC services is a popular concept nowadays with the introduction of HCE technology. In this context, Tokenization as a security method has important contributions for promoting HCE based NFC services. There are diverse standardization efforts (i.e., ASC X9, PCI DSS, Visa, EMVCo) on Tokenization method especially for payment service domain.

In accordance with these standards, we propose a novel generic usage model for HCE based NFC services such as loyalty and couponing, access control, identification and security applications. The proposed model aims to provide secure data service on the cloud for promoting HCE based NFC services, and uses two-phased Tokenization for providing secure communication between actors. Security and communication essentials of the proposed model are presented as well.

## Acknowledgment

This work is founded by Bilim Sanayi ve Teknoloji Bakanligi and KocSistem Information and Communication Services Inc. under SAN-TEZ project number 0726.STZ.2014.

## REFERENCES

Alattar, M. and Achemlal, M., Host-based Card Emulation: Development, Security and Ecosystem Impact Analysis, *Procedings of the IEEE International Conference on High Performance Computing and Communications,* August 20-22, 2014, Paris.

Coskun, V., Ok, K., and Ozdenizci, B. 2012. Near Field Communication (NFC): From Theory to Practice, 1st ed.; John Wiley and Sons: London, UK.

Coskun, V., Ozdenizci, B., and Ok, K. 2015. The Survey on Near Field Communication, *Sensors*, 15, 13348-13405.

Mobey Forum, 2014. The Host Card Emulation in Payments: Options for Financial Institutions, White Paper. Available Online: http://www.mobeyforum.org/whitepaper/the-host-card-emulation-in-payments-options-for-financial-institutions/.

PCI DSS, 2011. Tokenization Guidelines Version 2.0. Available Online: https://www.pcisecuritystandards.org/documents/Tokenization_Guidelines_Info_Supplement.pdf.

Smart Card Alliance Mobile and NFC Council, 2014. Host Card Emulation (HCE) 101, White Paper. Available Online: http://www.smartcardalliance.org/wp-content/uploads/HCE_Webinar_FINAL_061815.pdf.