3rd International Conference on Creative Technology
19 - 21 August 2015, Thailand

A Secure Communication Model for HCE based NFC Services

Busra Ozdenizci[1], Vedat Coskun[1*], Kerem Ok[1] and Turgay Karlidere[2]

[1]NFCLab Istanbul, Department of Information Technologies, ISIK University
Istanbul, Turkey
[2]KocSistem Information and Communication Services Inc.,
Istanbul, Turkey
*Corresponding Author: vedatcoskun@isikun.edu.tr

ABSTRACT

Near Field Communication (NFC) is a new promising short-range wireless communication technology that provides ease of use by triggering the communication with a simple touch and making the user feel secure by short distance communication. Most promising functionality of NFC technology is via card emulation operating mode that enables an NFC Smartphone to behave like a contactless smart card. By this way, using card emulation functionality enables realization of diverse applications such as mobile payment, ticketing, coupon, loyalty, access control, identification and so on. In this context, Secure Element (SE) is the important part of NFC Smartphones for securing the private data (e.g. credit and debit card numbers) or mobile application executable codes. On the other hand, Cloud based SE concept emerged especially after the introduction of Host Card Emulation (HCE) technology with Android 4.4 (KitKat) OS by Google which adds completely new functionality for card emulation based NFC services. HCE introduces new methods for storing, accessing, and managing private data on the Cloud instead of on the Smartphones. The purpose of this study to present a novel model for HCE based NFC services, which performs a secure communication using Tokenization method. Tokenization method opens up the possibility of enabling secure offline transactions for NFC services as well. The importance of Tokenization method for our proposed model is initially explained, and the system requirements of the proposed model are presented thereafter. The communication model is based on two-phased Tokenization that aims to provide secure interaction between system actors, and presented through an HCE based NFC access control use case.

Keywords

Near Field Communication, NFC, Secure Element, SE, Host Card Emulation, HCE, Tokenization.

INTRODUCTION

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 [2, 3].

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. As the 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 [3].

Card emulation is the most challenging operating mode, which enables an NFC Smartphone to behave as a contactless smart card by using ISO/IEC 14443 standard. 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. 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 [3].

Cloud based SE concept emerged after the introduction of HCE technology in Android 4.4 (KitKat) OS by Google, which separates the card emulation functionality from the SE [6]. 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.

HOST CARD EMULATION (HCE)

HCE can also be referred as Software based SE which is currently available on the Android OS (Android KitKat 4.4 and higher) and the BlackBerry OS. HCE enables storing and managing data on the Cloud and provides independence from Hardware based SE alternatives. 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 [1]. So, the mentioned application becomes able to exchange APDU commands with an NFC reader.

In case of Hardware based SEs, the APDU commands coming from the NFC reader are passed to the application on the 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 [6].

As the computing / storage capacity and development complexity are considered, HCE based NFC services are more advantageous over hardware based SE [1]. 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 [4].

There exist two methods for performing HCE services [4]: Full Cloud based HCE, and Tokenization based HCE. In case of 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.

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 create a network bandwidth and security requirements [4].

On the other side, Tokenization based HCE solution 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 [4, 5]. 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 (Figure 1).

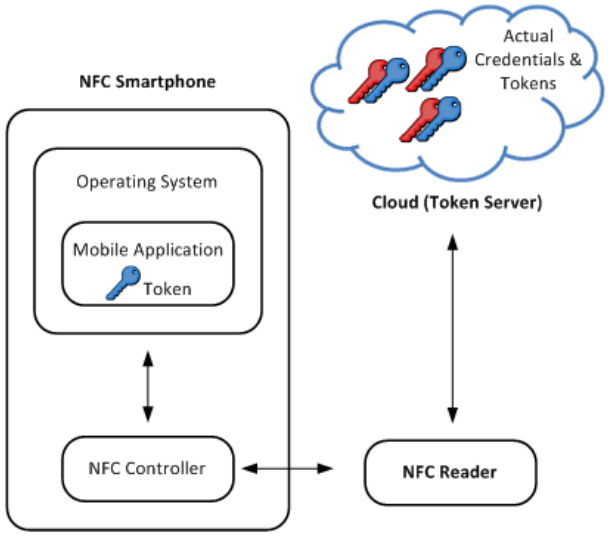

Figure 1. Tokenization based HCE solution.

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.

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.

SECURE COMMUNICATION MODEL FOR HCE BASED NFC SERVICES

A novel generic usage model is proposed for HCE based non-payment 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.

Two-Phase Tokenization Model

The communication model performs a two-phased Tokenization operation for providing secure interaction between system actors.

There exist three main actors: *NFC Smartphone users* who need to own HCE enabled Smartphones; *Service Provider* that supplies HCE enabled NFC service(s); *TSP* that provides token generation, secure data service, and token mapping processes.

In this context, two important security issues are considered for our 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.

(2) Application Identity Management: Another important 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, which is also stored on the data server of the TSP.

HCE based Access Control Use Case

The proposed model can be used for security, identification, access control, loyalty, and couponing applications.

The usage of the proposed model is presented and described below through an access control use case step by step hereunder (Figure 2):

Step 1: NFC Smartphone user firstly touches an NFC reader to a turnstile (e.g., security or access point). NFC reader requests identity of the user, after which *userToken* value on the NFC Smartphone is sent to NFC reader. For example, the *userToken* has *user1Token* value, which refers for identity data of Vedat Coskun (Table 1).

Step 2: NFC reader passes *user1Token* 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. In our example, the *appToken* has *app2Token* value, which is for Y Company Access Control application (Table 2).

Table 1. Example for *userToken* table.

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

Table 2. Example for *appToken* table.

appToken	Application Name
app1Token	X Company Access Control
app2Token	Y Company Access Control
…	…

Step 4: TSP performs token mapping process of received *userToken* and *appToken* values, which are *user1Token* and *app2Token*. TSP obtains the company ID of the user from Table 3 (i.e., 200101), which is stored on the data server.

Table 3. Token mapping table *userToken* and *appToken.*

	app1Token	app2Token
user1Token	Null	200101
user2Token	100101	Null
user3Token	100102	Null

Step 5: TSP sends an authorization response together with the secured data (i.e., ID of the user) to Service Provider.

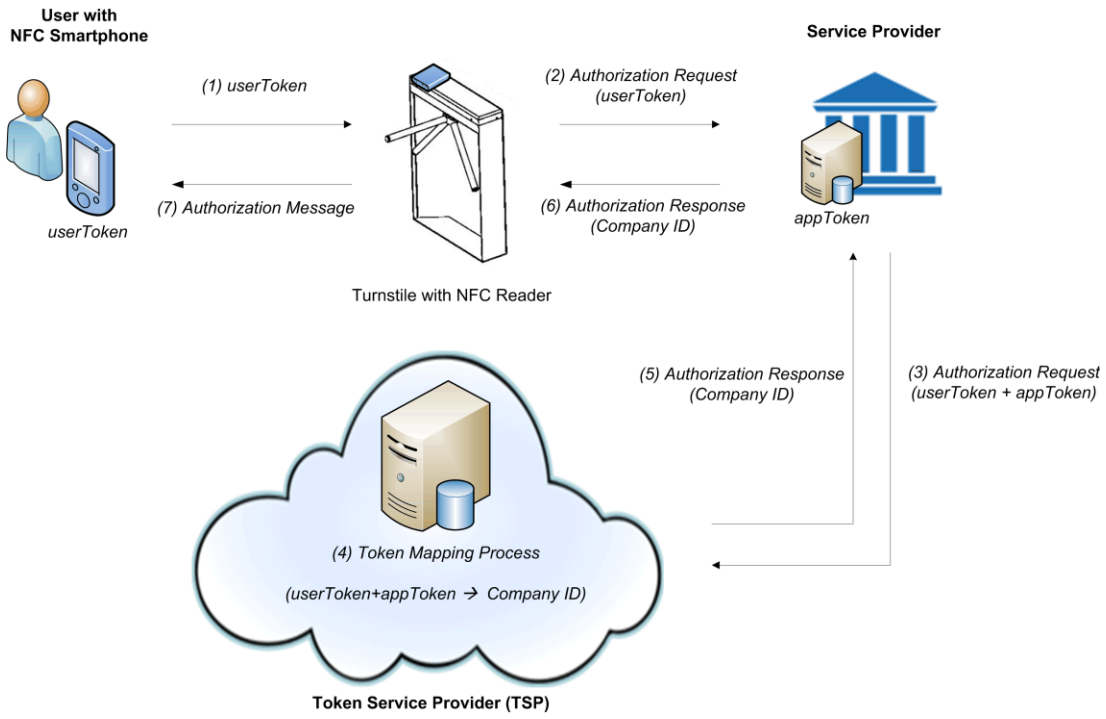

Figure 2. Usage model for HCE based NFC access control service.

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.

CONCLUSION

Tokenization as a security method has important contributions for promoting HCE based NFC services. A novel generic usage model for HCE based NFC services is proposed in this study which can be used for 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 benefits from two-phased Tokenization model for providing secure communication between system actors. The model is presented and explained briefly through an HCE based NFC access control service with its communication essentials.

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

1. Alattar, M. and Achemlal, M., Host-based Card Emulation: Development, Security, and Ecosystem Impact Analysis. In *Proc. of the IEEE International Conference on High Performance Computing and Communications 2014*.
2. Coskun, V., Ok, K., and Ozdenizci, B. Near Field Communication (NFC): From Theory to Practice, 1st ed.; John Wiley and Sons: London, UK, 2012.
3. Coskun, V., Ozdenizci, B., and Ok, K. The Survey on Near Field Communication, *Sensors* (2015), 15, 13348-13405.
4. Mobey Forum, 2014. The Host Card Emulation in Payments: Options for Financial Institutions. http://www.mobeyforum.org/whitepaper/the-host-card-emulation-in-payments-options-for-financial-institutions/.
5. PCI DSS, 2011. Tokenization Guidelines Version 2.0.https://www.pcisecuritystandards.org/documents/Tokenization_Guidelines_Info_Supplement.pdf.
6. Smart Card Alliance Mobile and NFC Council, 2014. Host Card Emulation (HCE) 101. http://www.smartcardalliance.org/wp-content/uploads/HCE_Webinar_FINAL_061815.pdf.