Smartphones | Contactless IC Cards | Credit Cards

Technology Reports

Credit Services for the Osaifu-Keitai Mobile Payment System on Open OS Terminals

Smartphones have grown in popularity and their OSs have been published as open-source. Although open-source software is useful, it is at risk of being attacked based on the discovery of vulnerabilities by techniques such as reverse engineering. To run credit services based on our Osaifu-Keitai mobile payment system on an open-source OS terminal, a different architecture from that of conventional feature phones must be used. We have therefore developed a credit service for open OS terminals that mitigates the impact associating with revisions of existing systems while maintaining the security level achieved by feature phones.

Credit Card Business Division
Nobuyuki Miura †
Jin Hoshino
Services Platform Department
Jin-ichi Hirose
Takashi Fukuzono ††

1. Introduction

Smartphones have grown in popularity and some of them now use OSs that have been published as open-source[*1]. Although the publication of an OS as open source software helps it to become more mature and more widely used and encourages the development of applications for this OS, open source may place the OS at risk of being attacked based on the discovery of vulnerabilities by techniques such as reverse engineering[*2]. A new system architecture is needed to ensure that credit services using our Osaifu-Keitai mobile payment system are no less secure on terminals with this sort of open-source OS than on conventional feature phones (conventional i-mode terminals in this case). Also, when adapting an existing system built for feature phones so that it can run on this sort of new architecture, it is essential to consider how to mitigate the impact of system revisions and reduce the costs and development time associated with these improvements. We have therefore developed a credit service for open OS terminals that mitigates the impact on existing systems while maintaining the security level achieved by feature phones. In this article we describe how the service was implemented.

2. Architecture of Feature Phones and Open OS Terminals

The most important function for the implementation of credit services in Osaifu-Keitai is the function for securely writing credit card information to the contactless IC chip[*3] (FeliCa®[*4] chip) in Osaifu-Keitai. When writing information, a secure communication channel between a contactless IC chip server (FeliCa server) and the FeliCa chip is used to minimize the risk of tampering or eavesdropping. After the card information has been written into the FeliCa chip, the hardware security configuration of the FeliCa chip is designed to

† Currently Frontier Services Department
†† Currently Credit Card Business Division

*1 **Open source**: A generic term for a software license that allows source code to be published while protecting the copyright of the software author, or source code that has been published in this way.

*2 **Reverse engineering**: A process of analyzing the configuration and operation of software or hardware to clarity manufacturing methods and operating principles.

prevent the credit card information from being tampered with.

Figure 1 compares the architecture of a feature phone with that of an open OS terminal. In a feature phone, the card information written to the FeliCa chip is downloaded from a server that manages card information and the like (hereinafter referred to as "card information server"), and is temporarily held in memory. Furthermore, the credit card information is also written to the FeliCa chip by issuing a write request to the FeliCa server. On the other hand, the terminal application memory on an open OS terminal is liable to be exploited in an attack, and if it is used for the temporary storage of credit card information on an open OS terminal then this information is at risk of being forged or tampered with so that unauthorized credit card information is written to the FeliCa chip. We therefore adopted an architecture where an intermediary server that is less susceptible to attack is placed between the terminal application and the FeliCa server or other server used for the storage of card information, so that the card information and other such data is temporarily stored on the intermediary server. Since the existing servers for card information and the like have been built for use with

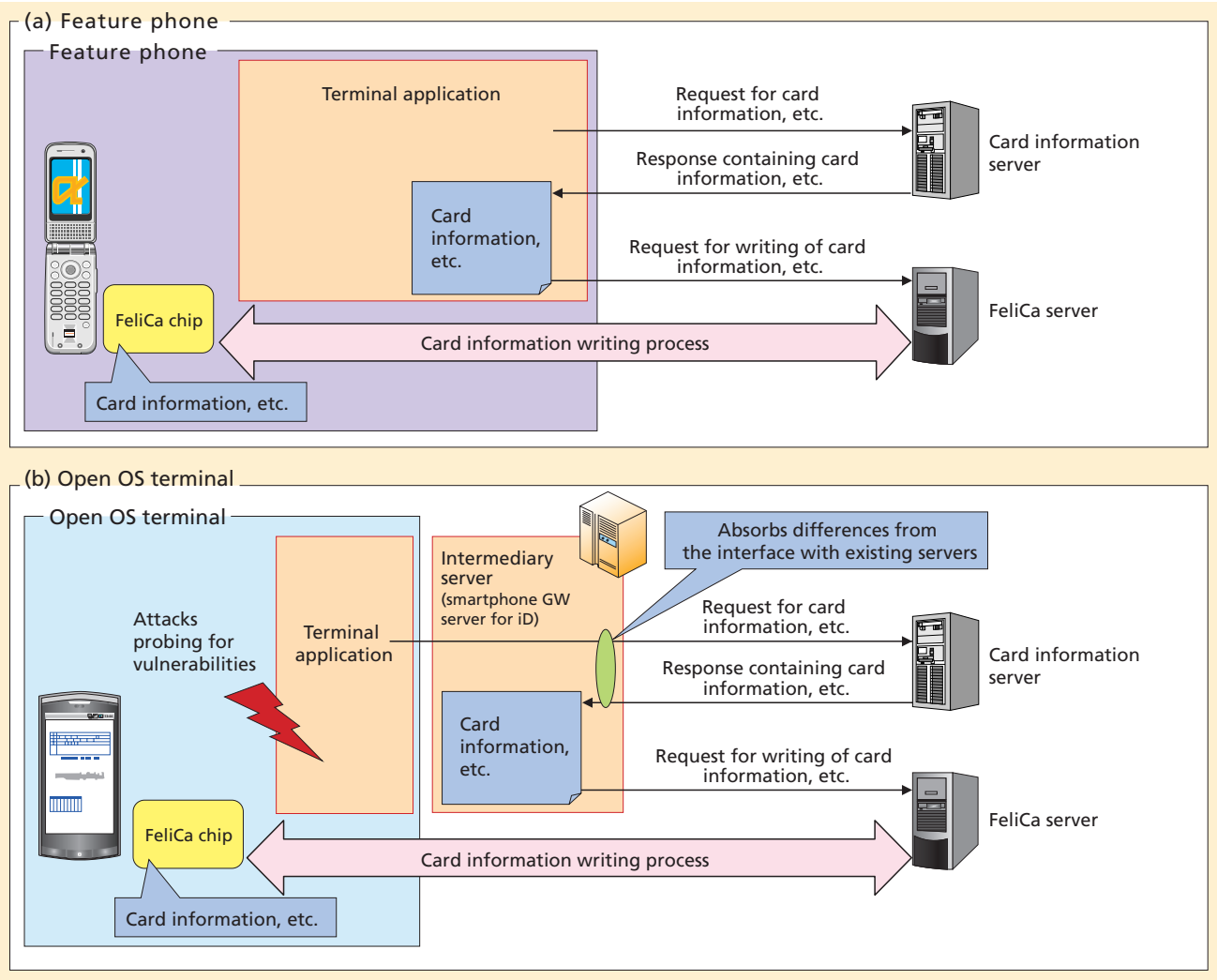


Figure 1 Comparison of the architectures of feature phones and open OS terminals

*3 **Contactless IC chip**: A semiconductor integrated circuit that exchanges information by radio communication with an IC card reader/writer.

*4 **FeliCa**®: A contactless IC card technology developed by Sony Corp. A registered trademark of Sony Corp.

feature phones, the intermediary server loads a function for ironing out the interfaces differences of terminal applications running on open OS terminals so as to minimize the impact on existing servers.

3. Implementing the Intermediary Server and Terminal Applications

3.1 Smartphone GW Server for iD

The smartphone GW server for iD [*5] is an intermediary server for securely writing credit card information to a FeliCa chip mounted in an open OS terminal. **Figure 2** shows the functional configuration of the smartphone GW server for iD.

The smartphone GW server for iD chiefly provides four functions:

1) FeliCa Chip Update/referencing Function

A framework for securely writing to the FeliCa chip is offered as a FeliCa server. In the smartphone GW server for iD, the FeliCa server is used to write credit card information directly to the FeliCa chip, thereby eliminating the deployment of credit card information into the memory of the open OS terminal and securing the information against forgery or tampering. In the same way, this framework is also used when credit card information that has been set in the FeliCa chip is read out to the server to prevent forgery and tampering.

2) Interface Conversion Function

Communication to card information servers (DOCOMO credit card system (Credit Mobile Gateway System (CREMO)), card information download center, brand download center) that is next to the smartphone GW server for iD, is all collected into a single session when card information is written to a FeliCa chip. And, interface messages between the systems are issued according to requests from the terminal application.

In the smartphone GW server for iD, interface messages from the terminal application for open OS terminals are converted into the same format as interface messages for the feature phone interface, thereby minimizing the scope of revisions that need to be made to the system that replaces the smartphone GW server for iD. In this way, we aim to reduce the costs and development time associated with introducing the new system.

3) Sequence Management Function

Requests from open OS terminals are stored as session [*6] information to prevent the generation of interrupts from fraudulent terminals. When operations to write credit card information are configured by multiple interface messages from a terminal application, coordinating information about management information (IP addresses, passwords, etc.) is stored on the smart-

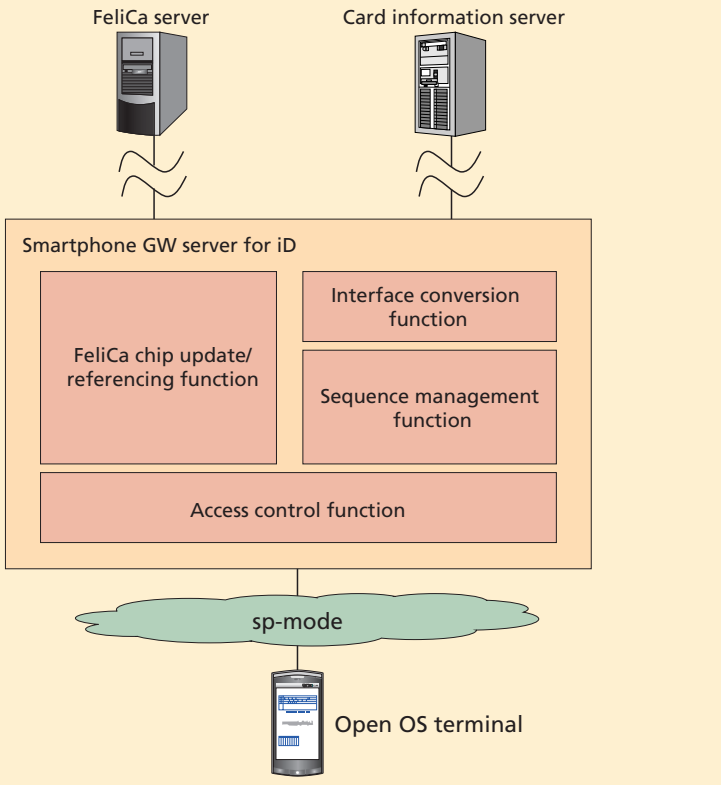


Figure 2 Configuration of smartphone GW server for iD

*5 **iD**: "iD" and the "iD" logo are trademarks or registered trademarks of NTT DOCOMO.

*6 **Session**: A meaningful episode of communication between a server and client. Here, the sequence of communication involved in writing card information is treated as a session.

phone GW server for iD, and is carried around until one session has ended. When interface messages from the terminal application are not coordinated in the prescribed order, a sequence error is deemed to have occurred and the fraudulent application is prevented from issuing interrupts.

4) Access Control Function

Since the smartphone GW server for iD is a system that can be accessed from open OS terminals, it is at risk of being subjected to Denial of Service (DoS) attacks[*7]. Although common security defenses are in place, it also has an access control blacklist function to prepare for security attacks from specific users. To allow the functionality of a service to be checked before it is launched, there is also a whitelist function that only allows access from specific open OS terminals.

3.2 Terminal Application

The terminal application is used to secure an area in the FeliCa chip and manage multiple cards. It can manage up to two cards, and has functions for tasks such as issuing areas, adding cards and deleting cards.

As mentioned in chapter 2, the characteristics of the open OS terminal application are provided via an intermediary server, and security is maintained by temporarily storing card information and the like on the intermediary server.

The terminal application has the following two main functions.

1) FeliCa Chip Update/referencing Function

With this function, users are able to settle payments with an open OS terminal by allowing the FeliCa chip to perform all access tasks such as issuing and deleting areas, adding, deleting and updating card information, updating the main card, and storing/removing card information. In the implementation of this function, card information is transmitted to the terminal in a feature phone, where the terminal application performs processing to write this information to the FeliCa chip, whereas in an open OS terminal as described in chapter 2 and section 3.1, the information stored in the terminal's memory is liable to be tampered with. Consequently, instead of transmitting credit card information to the terminal application, this information is terminated at the smartphone GW server for iD so that it can be written directly to the FeliCa chip without being deployed in the memory of the open OS terminal. The specific sequences are compared in **Figure 3** and **4**.

2) Unique Information Display Functions for Each Card Issuer

It is possible to manage the information of up to two cards in the terminal application, but the messages, company logos and the like displayed

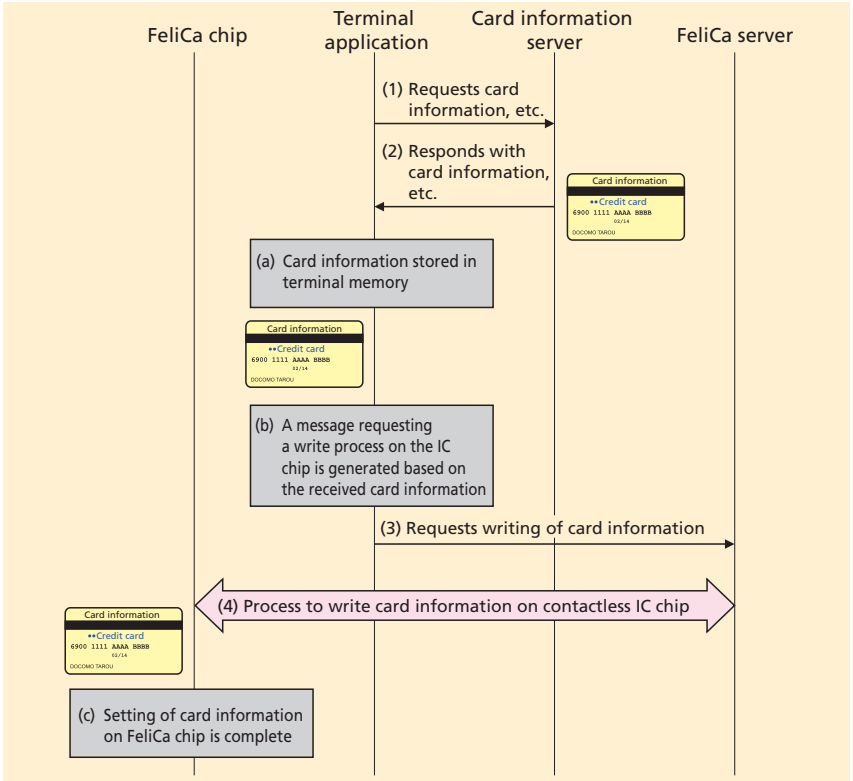


Figure 3 Card information writing sequence in a feature phone

*7 **DoS attack**: A malicious attack that can cause a service to stop.

in the terminal application must be changed for each card issuer. The function for displaying different information for each card issuer is implemented by acquiring data such as messages and images specific to each card issuer from the card information server when the card information is set. To make the best possible use of systems that have already been provided for feature phones, we suppressed the scope of revisions to the bare minimum by converting the format of interface messages from the application into the same format as interface messages for feature phones as described in section 3.1 2).

3.3 Other Security-related Issues

To configure iD from a terminal application, it currently has to be accessed via an sp-mode 3G network.

Access from other bearers [*8] such as wireless LAN is restricted because applications for open OS terminals operate via an intermediary server (smartphone GW server for iD). If a Domain Name System (DNS) [*9] server spoofs the IP address of this intermediary server, then a rogue server will be able to steal passwords and the like that could be used to set fraudulent credit card information.

4. Service Image

Figure 5 shows the service image of credit services implemented on an

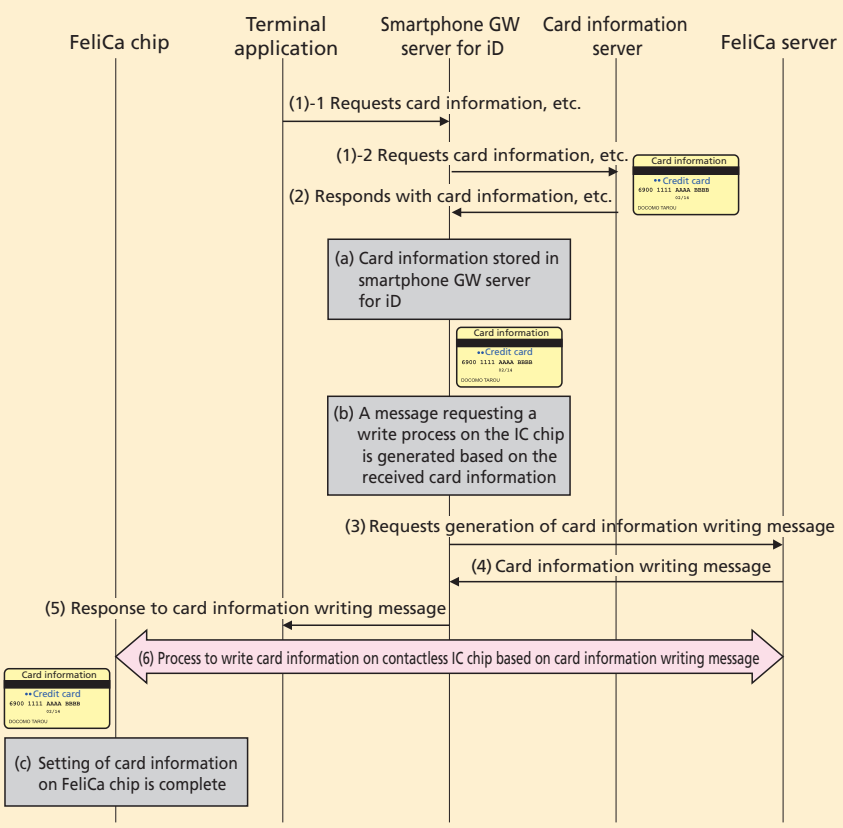


Figure 4 Card information writing sequence in a smartphone

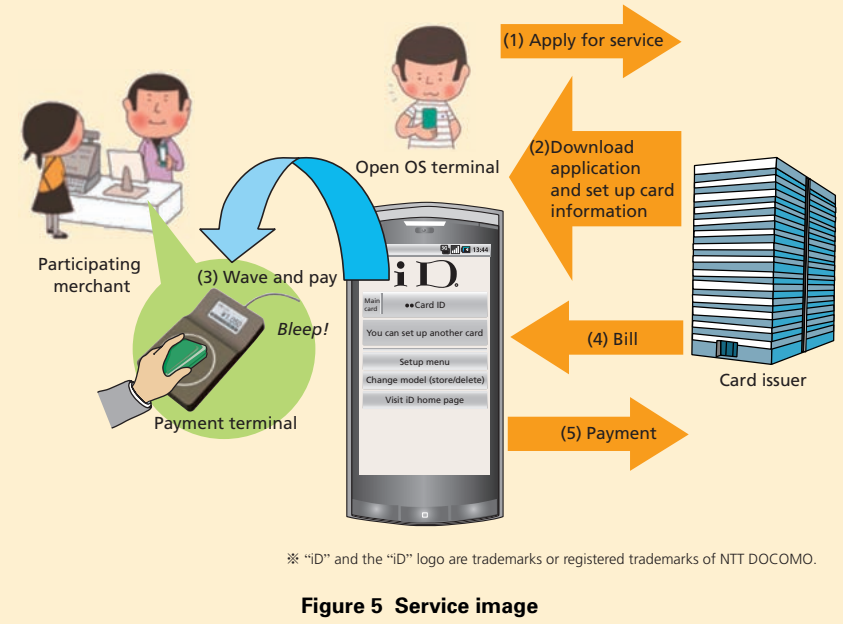
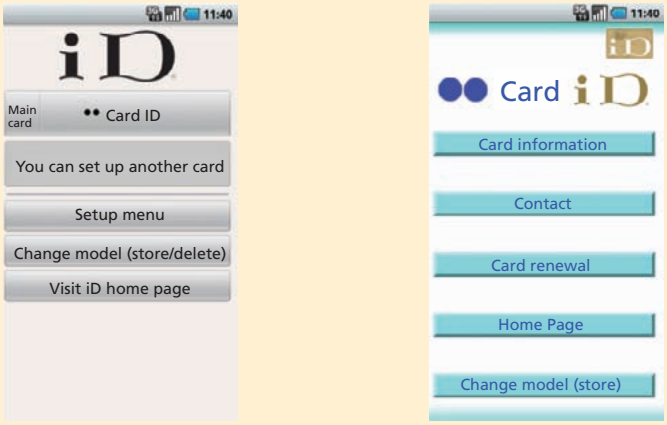


Figure 5 Service image

*8 **Bearer**: A communication circuit that transmits information.
*9 **DNS**: A system that associates host names and IP addresses on IP networks.

open OS terminal, and **Figure 6** shows a screenshot of the terminal application. People are able to use this contactless IC credit service in exactly the same way regardless of whether they are using a feature phone or an open OS terminal.



(a) Terminal application top page

(b) Different information displayed for each card issuer

Figure 6 Terminal application screenshots

5. Conclusion

In this article, we have described the development of a credit service compatible with open OS terminals that mitigates the impact of upgrading existing systems while maintaining the same level of security offered by feature phones. In the future, we hope to study and implement services that exploit the flexibility of applications on open OS terminals.