PAPER

# An ATM Security Measure for Smart Card Transactions to Prevent Unauthorized Cash Withdrawal

**Hisao OGATA**[†,††a)], ***Member***, **Tomoyoshi ISHIKAWA**[†], **Norichika MIYAMOTO**[†], ***Nonmembers***, ***and*** **Tsutomu MATSUMOTO**[††], ***Member***

**SUMMARY** Recently, criminals frequently utilize logical attacks to install malware in the PC of Automated Teller Machines (ATMs) for the sake of unauthorized cash withdrawal from ATMs. Malware in the PC sends unauthorized cash dispensing commands to the dispenser to withdraw cash without generating a transaction. Existing security measures primarily try to protect information property in the PC so as not to be compromised by malware. Such security measures are not so effective or efficient because the PC contains too many protected items to tightly control them in present ATM operational environments. This paper proposes a new ATM security measure based on secure peripheral devices; the secure dispenser in an ATM verifies the authenticity of a received dispensing command with the withdrawal transaction evidence, which is securely transferred from the secure card reader of an ATM. The card reader can capture the transaction evidence since all transaction data flows through the card reader in a smart card transaction. Even though the PC is compromised, unauthorized dispensing commands are not accepted by the secure dispenser. As a result, the new security measure does not impose heavy burden of tighter security managements for the PCs on financial institutes while achieving stringent security for the logical attacks to ATMs.

***key words:*** *ATM, security, malware, cryptography, device*

## 1. Introduction

Attacks to Automated Teller Machines (ATMs) used to be only physical attacks such as card skimming to steal card holder data and physical crash of ATM bodies to steal cash. Recently, criminals frequently utilize logical attacks for the sake of unauthorized cash withdrawal from ATMs. Typical logical attacks are so-called "Jackpotting" [1]–[4] and "Black Boxing" [5], [6] which is a variant of "Jackpotting." "Jackpotting" is such an attack that malware in the PC of an ATM sends unauthorized cash dispensing commands to the dispenser to withdraw cash from the ATM without generating a transaction. Regarding "Black Boxing," an external computer is directly connected with the dispenser and malware on the computer sends unauthorized cash dispensing commands to the dispenser. "Jackpotting" is prevailing much more than "Black Boxing" because "Jackpotting" uses only software that attacks the vulnerable ATM platform commonly installed in the PC of ATMs. In general, an ATM consists of a PC running the Windows®*1 Operating System (OS) and peripheral devices such as a card reader and a dispenser. The ATM platform provides Financial Institute (FI)'s multi-vendor application on the PC with standardized Application Programming Interfaces (APIs) to control peripheral devices. As the APIs' specifications are open to the public and the API's are not cryptographically protected, malware frequently utilizes the ATM platform for unauthorized cash dispensing.

The existing security measures [7]–[10] primarily try to protect information property in the PC and the communication line between the PC and the dispenser to prevent those logical attacks. Measures tightly protecting information property in the PC are not so effective or efficient. For example, even whitelist-based anti-malware software does not always work well to protect files and data in the PC since it might be disabled by directly manipulating the PC. Furthermore, the PC contains too many files and data, which need to be tightly protected. While the PC contains more than twenty thousand files, a FI has to manage hundreds to thousands of ATMs. Thus the total protected files amount to more than ten million to one hundred million. Tightly protecting such huge number of files would bring quite heavy management workloads to FIs because frequent physical/logical accesses inside each ATM are required in existing ATM operations. That is, frequent physical/ logical accesses increase risks of logical attacks on ATMs. Examples of those accesses are periodical cash replenishment and collection for cash services, frequent software updates for better services, contents updates for advertisement and so on. As a result of quite heavy management workloads, the existing measures may result in breaking rules, human errors, and rather poor management by ATM operation staffs.

In this paper, we propose a new ATM security measure based not on the tightly protected PC but on secure peripheral devices; a secure dispenser verifies the authenticity of a received cash dispensing command with the withdrawal transaction evidence, which is securely transferred from a secure card reader. The card reader can capture the transaction evidence since all transaction data flows between the host computer and a smart card through the card reader in a smart card transaction. An encrypted communication between the card reader and the dispenser is newly introduced so as to securely transfer the transaction evidence.

The remainder of this paper is organized as follows. Chapter 2 addresses the issues of existing ATM systems and operations, and conditions that a security measure can effec-

Manuscript received April 16, 2018.
Manuscript revised August 27, 2018.
Manuscript publicized December 6, 2018.

[†]The authors are with Hitachi-Omron Terminal Solutions, Corp., Tokyo, 141–8576 Japan.
[††]The authors are with Yokomaha National University, Yokohama-shi, 240–8501 Japan.
a) E-mail: hisao_ogata@hitachi-omron-ts.com

DOI: 10.1587/transinf.2018EDP7136

tively protect cash in an ATM. Chapter 3 presents the proposed security measure. Chapter 4 describes application of the proposed measure to a ticketing device. Chapter 5 is the conclusion.

## 2. The Issues of Existing ATM Systems and Operations

### 2.1 Overview of an ATM System and a Cash Withdrawal Transaction

An overview of an ATM system is depicted in Fig. 1. An ATM is composed of a PC and peripheral devices controlled by the PC. While the PC is connected with each peripheral device through a communication line, the PC is also connected with the host computer through a wide area network, namely the FI's intranet. The PC is logically constituted with three layers: multi-vendor application, ATM platform to control the peripheral devices, and Windows®*1 OS. The ATM platform provides international standardized interfaces to multi-vendor application: Comité Européen de Normalisation / eXtensions for Financial Services (CEN/XFS) APIs [11]. The ATM platform architecture was established in 90s and the primary concepts are interoperability and compatibility. The APIs' specifications are open to the public and the APIs are not cryptographically protected. Encrypting PIN Pad (EPP) is a peripheral device used for an ATM user's Personal Identification Number (PIN) entry to show proof of identity. The EPP itself outputs an encrypted PIN called Enciphered PIN Block (EPB), which is cryptographically protected in conformity with the EMV®*2 (EuroPay, MasterCard International and Visa International) specifications [12], the Payment Card Industry (PCI) requirements [13], [14] and ISO 9564 [15]. Figure 2 outlines an example of an existing cash withdrawal transaction using a smart card. It is supposed that the multi-vendor application includes transaction application (AP) to process transaction messages and dispensing AP to control a dispenser. A session to create an EPB is the EPP and the host computer in conformity to the PCI requirements. The transaction sequence is described as follows:

S1: A user inserts a smart card (FI's cash card) into the card reader of an ATM to read the Primary Account Number (PAN) on the card. The PAN was preliminarily assigned to the card by the FI to identify the user. After that, the user inputs the PIN with the EPP and the EPP outputs an EPB to the transaction AP.
S2: The user selects "cash withdrawal" from the menu on the screen and inputs a withdrawal amount to ATM.
S3: The transaction AP creates a transaction request message based on the user's request and sends the message and the EPB to host computer.
S4: The host computer verifies the authenticity of the message, extracts the PIN from the EPB to authenticate the user's identity, and confirms the user's account balance or the credit. Then the host computer sends the transaction response message back to the transaction AP.
S5: The transaction AP provides the dispensing AP with the cash dispensing request in accordance with the transaction response message. Then the dispensing AP sends a cash dispensing command to the dispenser.

The detailed data flow of (S3) to (S5) is illustrated in Fig. 3. (S3) to (S4) conform to the EMV specifications [12], [16] and PCI DSS [17]. The brief summary of the EMV specifications is as follows. A smart card and the host computer must be a tamper-proof secure device. Transaction messages and corresponding Message Authentication Codes (MACs) to verify the authenticity and the integrity of a message are exchanged between the smart card and the host computer. The smart card generates a MAC to a transaction request message created by the transaction AP and then verifies a MAC to a transaction response message received from the host computer.

A master key to generate a session key for a MAC has been installed in a smart card and the host computer through a card personalization process before issuing the card. The master key is linked with the user's PAN. A unique session key for each transaction is generated from the master key and a transaction counter output from the smart card according to the EMV specification [16]. The host computer also

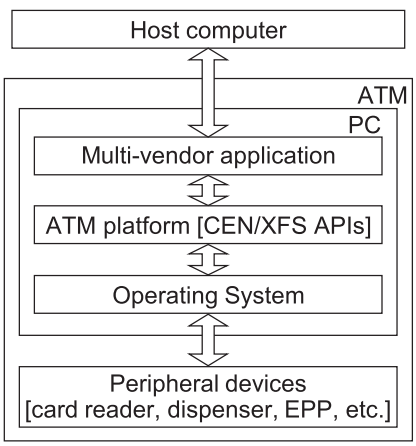


**Fig. 1** An overview of an ATM system.

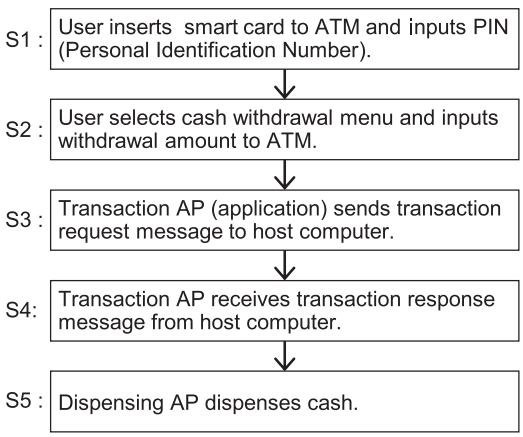


**Fig. 2** An outline of cash withdrawal transaction example.

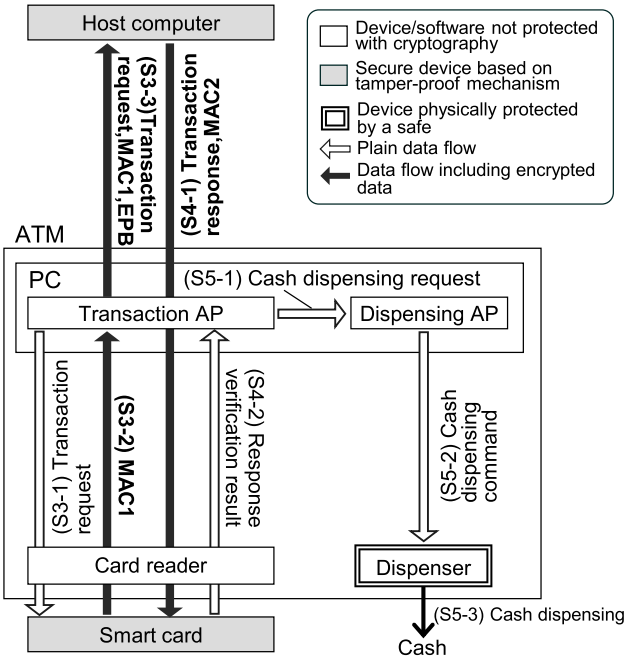


**Fig. 3** Data flow of existing cash withdrawal transaction.

shares the same session key conforming to the specification. The dispenser is secure against unauthorized physical manipulation because it is physically protected by a safe. The ATM platform and the OS are omitted in the figure. A sequence of the detailed data flow and data processes is described below. "-number" shows a sub-step of each step in Fig. 2.

S3-1: The transaction AP creates a transaction request message from the PAN and the withdrawal amount. After that, the transaction AP sends the message to the smart card through the card reader using a CEN/XFS API.

S3-2: The smart card generates a MAC to the message called "MAC1" and then sends the MAC back to the transaction AP.

S3-3: The transaction AP sends the transaction request message, "MAC1", and the EPB received in S1 to the host computer.

S4-1: The host computer verifies the PIN extracted from the EPB, the authenticity of the transaction request message with "MAC1", and checks the user's account balance or the credit. Then the host computer creates a transaction response message and "MAC2" for it, and the host computer sends them back to the transaction AP and the AP forwards them to the smart card through the card reader using a CEN/XFS API.

S4-2: The smart card verifies the authenticity of the received message with "MAC2" and returns the response verification result to the transaction AP. The value of the verification result varies in accordance with the host computer's decision. It is noted that the response verification result is plain data as the smart card and the transaction AP do not share any cryptographic keys.

S5-1: The transaction AP provides the dispensing AP with the cash dispensing request including the withdrawal amount in accordance with the response verification result.

S5-2: The dispensing AP specifies the bill denomination according to the user's selection, and sends a cash dispensing command to the dispenser to dispense cash through another CEN/XFS API.

S5-3: The dispenser dispenses cash following to the received cash dispensing command.

### 2.2 Issues of ATM Systems and Operations

As explained in Sect. 2.1, (S3-2), (S3-3), and (S4-1) are securely protected with cryptographic technologies compliant with the EMV specifications and PCI requirements while other device, software, data are not. Hence criminals can perpetrate "Jackpotting" and "Black Boxing" by attacking vulnerable (S5-1), (S5-2) and dispensing AP. The existing measures [7]–[10] try to protect information property in the PC against "Jackpotting" and try to cryptographically protect the communication line for (S5-2) against "Black Boxing" [7], [8]. Some ATM vendors and FIs also try to cryptographically protect the communication line for (S3-1) and (S4-2). Those communication lines' protection also relies on protecting information property in the PC and the peripheral devices, which include cryptographic keys and cryptographic processing modules. As a result, the whole ATM, which consists of the PC, the peripheral devices, and the communication lines between the PC and the peripheral devices, should be tightly protected in the existing measures. However, such measures are not so effective or efficient by virtue of the following situations of existing ATM systems and operations. The most critical issue is the PC's protection.

**System aspects:**

(1) Vulnerable CEN/XFS APIs

The primary specifications of CEN/XFS APIs were established in 90s and their security functionality is rather poor than that of the current standards. Hence malware frequently utilizes CEN/XFS APIs for "Jackpotting." Many FI's multi-vendor application have been developed on CEN/XFS APIs. Even though secure CEN/XFS APIs are newly developed, it may take a long time to make such secure APIs common since a lifetime of ATMs is usually 7 to 10 years.

(2) Complicated logical structures of ATM software

FIs must provide ATM users with various kinds of ATM services, e.g. not only transactions within the FI but also transactions with other FIs. Logical structures and data processing of ATM software are very complicated and there are so many software components more than twenty thousands in each ATM. Even a tiny change of ATM software may bring a serious ATM system trouble in some cases because it is quite difficult to completely

confirm all software components and configurations in an ATM before releasing it. It is further difficult to completely confirm all software components in cases of OS updating for security patch and OS hardening since an OS is the base of all layers above.

**Operational aspects:**

(3) Frequent physical access inside an ATM
Frequent physical accesses inside an ATM are required according to the reasons listed below.

- Replenishment of bills in the dispenser
- Replenishment of receipt paper sheets
- Periodical cleaning of bill dust in the dispenser
- Removal of bill jam for troubleshooting
- Replacement of parts for troubleshooting
- Off-line system updating and log data collection due to poor network performance in some cases.

Accessing the PC is also allowed during such physical accesses inside an ATM.

(4) Frequent and unprotected system updating
FIs frequently must update ATM systems in order to improve services, to update advertisement contents, to patch the OS and so forth. System updating is not tightly controlled if it is not covered by the EMV specifications and the PCI requirements. Tightly controlled system updating may impact timely launching services as it would take a very long time to completely confirm integrity and compatibility of all software in an ATM.

### 2.3 Conditions to Effectively Prevent "Jackpotting"

Tightly protecting the whole ATM is not so practical due to the situation of existing ATM systems and operations described above. The most critical issue is protecting the PC containing a lot of information property in the ATM. Therefore, we focus on preventing "Jackpotting" compromising the PC and a security measure should satisfy the following three conditions to cope with the existing situations.

**(A) A security measure should not significantly impact management workloads of existing ATM operations.**
The existing measures recommend tight access controls to the PC in each ATM with a unique login password, a unique physical lock and additional tight management measures. On the other hand, frequent physical and logical accesses inside several thousands of ATMs are required during ATM operations in some cases. Such tight and frequent access controls to ATMs may result in a heavy burden of managing so many ATMs. One idea is to enclose the PC with a tamper-proof box to protect it from unauthorized physical access, however it would take a long turnaround time to fix the PC for troubleshooting. As one of the most breakable devices in an ATM is hard disk drives in the PC, such a long turnaround time could not be accepted for FIs. Consequently, the condition (A) is required.

**(B) A security measure should not significantly impact ATM system availability.**
The existing guidelines have required FIs to update and harden the OS of ATMs in the aim of patch for the vulnerability. However, FIs may hesitate to conduct them since occasional ATM system troubles accompanying OS updating is not allowed as a social infrastructure. As an OS is the base of all software layered above, it is quite difficult for FIs to comprehensively test the compatibilities of so many software components within a limited time to keep the OS up to date. Consequently, the condition (B) is required.

**(C) "Jackpotting" cannot be successful even though integrity of all software related to dispensing commands is not assured.**
Taking into consideration the conditions explained above, it is quite difficult to completely assure integrity of all software and data on the PC related to dispensing commands in existing ATM operations. Furthermore, as the primary objectives of vulnerable CEN/XFS APIs are interoperability and compatibility for multi-vendor applications, it is not practical to drastically change the APIs specifications for a security objective. Different approaches are needed to cope with the situation. Consequently, the condition (C) is required.

We pick up the EUROPOL's guidance and recommendations [7] here as a representative of the existing security measures. The eventual objective of the EUROPOL's requirements to prevent "Jackpotting" is to tightly protect information property in the PC. It is obvious that those requirements do not meet the conditions (A) to (C).

## 3. A Measure Based on the Secure Peripheral Devices

### 3.1 Primary Idea of the Proposed Measure

As shown in Fig. 4 (a), the PC totally controls the dispenser on the basis of a master-slave model in an existing ATM as well as other device control systems. Such control system works securely if the PC is secure while it is not always secure in the actual situations. The objective of the existing measures is to protect information property in the PC so that

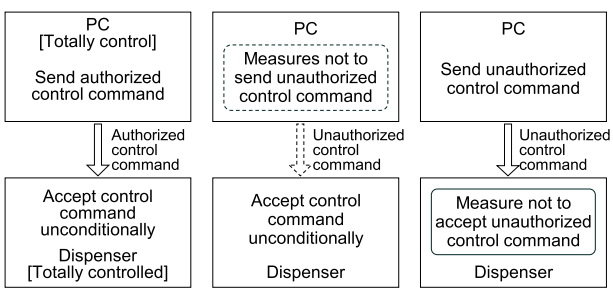


**Fig. 4** Objective of existing measures and the proposed measure.

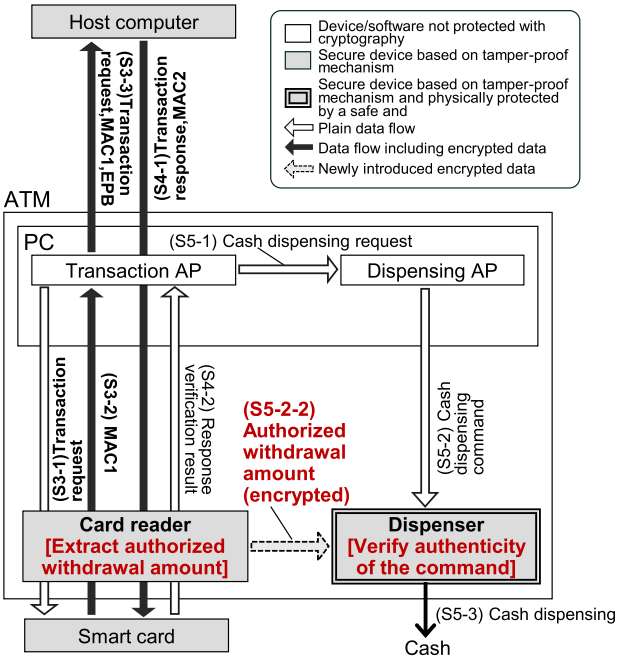


**Fig. 5** Primary idea of the proposed measure.

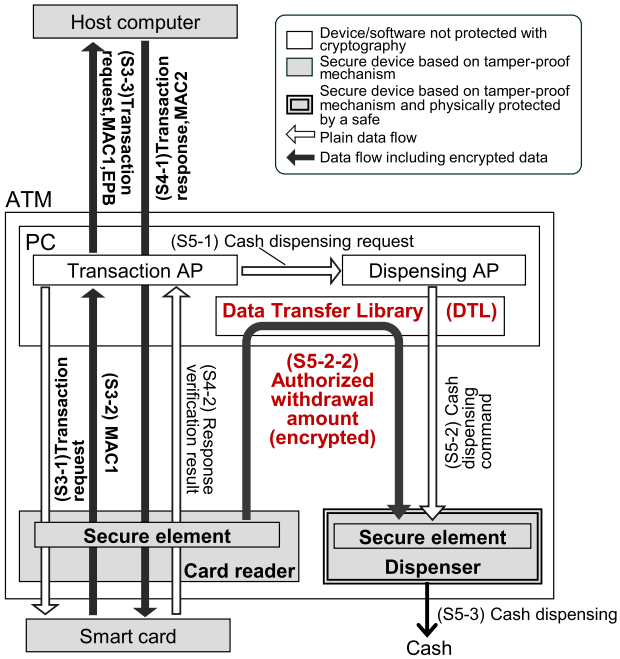


**Fig. 6** Implementation of the proposed measure.

the PC does not send unauthorized cash dispensing commands to the dispenser (Fig. 4 (b)). On the other hand, the objective of the proposed measure is to implement a function verifying the authenticity of a received command into the dispenser so that the dispenser does not accept unauthorized commands (Fig. 4 (c)).

The primary idea of the proposed measure is shown in Fig. 5. The existing dispenser on the totally controlled side does not have any information to verify the authenticity of a cash dispensing command. The encrypted data flow: (S5-2-2) authorized withdrawal amount is newly introduced between the card reader and the dispenser so that the dispenser can get information verifying the authenticity of a command. Two secure peripheral devices are introduced for the objective: a proposed card reader to extract an authorized withdrawal amount from the withdrawal transaction data flows, a proposed dispenser to verify the authenticity of the command with the authorized withdrawal amount (S5-2-2).

The authenticity of the command is confirmed with two kinds of conditions; approval of the cash withdrawal transaction by the host computer including the withdrawal amount and the proved identity for the transaction. The two conditions are assured only by (S5-2-2) based on the mechanism explained below. Regarding the first condition, the proposed card reader can extract the withdrawal amount from (S3-1) while it can extract whether the amount is approved or not from (S4-2) in order to generate (S5-2-2). Although (S4-2) is plain data, the proposed card reader can receive authentic (S4-2) since the card reader can receive it as soon as the smart card outputs (S4-2) before malware, if any, in the PC received it. Concerning the second condition, the card reader's receiving (S4-2) suggests that the identity is successfully proved in (S4-1). It is assured because the host computer sends (S4-1) indicating the approved response only when the PIN is successfully verified.

The withdrawal amount and the PAN in (S3-1) can be altered by malware in the proposed idea; nevertheless the protection priority is low due to the reasons described below. As an altered withdrawal amount in (S3-1) goes directly to the altered amount of cash dispensed to the ATM user, either the user or the FI does not suffer any monetary loss. If the PAN in (S3-1) is altered by malware, it becomes inconsistent with the session key and the master key in the smart card since the PAN is linked with those keys as explained in Sect. 2.1. In this way, the proposed measure can effectively and efficiently prevent unauthorized cash dispensing by protecting only the card reader, the dispenser and (S5-2-2). It is a contrast to the existing measures that try to tightly protect the whole ATM, which is not so practical as described in Sect. 2.2.

As alternative implementation of the proposed measure, the proposed card reader can send an encrypted cash dispensing command to the dispenser instead of (S5-2-2). It is not workable according to the existing ATM services. Some users select denominations of dispensed bills on the screen before cash dispensing. Therefore, the proposed card reader must support such application and must control the GUI on the screen as substitute for the dispensing AP. Furthermore, a card reader constantly needs to know the state of the PC to control the GUI. It is not practical from a viewpoint of the card reader's hardware resource and cost.

### 3.2 Implementation of the Proposed Measure

An implementation example of the proposed measure is out-

lined in Fig. 6. An encrypted communication to transfer the authorized withdrawal amount is implemented through the PC and the existing communication lines between the PC and the each peripheral device. The reason is that there is not any physical communication line between the existing card reader and the existing dispenser. Data Transfer Library (DTL) is newly introduced to simply provide a communication path between those devices. DTL is supposed to be installed in a layer below the CEN/XFS APIs. The tamper-proof secure element in the proposed card reader is equipped with the two functions; one is to extract an authorized withdrawal amount from (S3-1) and (S4-2), and the other is to provide an encrypted communication with the proposed dispenser to securely transfer the authorized withdrawal amount. The tamper-proof secure element in the dispenser supports the two functions; one is to provide an encrypted communication with the card reader, the other is to verify the authenticity of a received cash dispensing command with the received authorized withdrawal amount. The cryptographic key management and a session creation for the encrypted communication are supposed to conform to the international standards [18]–[20]. A session is supposed to have been preliminarily created. The detailed process flows are described as follows. Only modified processes are explained here.

S3-1: The transaction AP creates a transaction request message from the user's PAN and the withdrawal amount, and then sends it to the smart card through the card reader using the CEN/XFS API. The secure element in the card reader captures the message and extracts a withdrawal amount and stores it in the element.

S4-2: The smart card returns the response verification result to the transaction AP through the card reader. The secure element in the card reader captures the verification result and generates the authorized withdrawal amount from the withdrawal amount in (S3-1) and the verification result. Then the secure element encrypts the amount and stores the encrypted amount in the element.

S5-2: The dispensing AP sends a cash dispensing command to the dispenser through the CEN/XFS API and DTL. Once the DTL receives the dispensing command, DTL requests the proposed card reader to send the encrypted amount. Then DTL forwards the dispensing command (S5-2) and the encrypted amount to the proposed dispenser as shown in (S5-2-2).

S5-3: The secure element in the dispenser decrypts (S5-2-2) and confirms whether the dispensing amount in (S5-2) and the authorized withdrawal amount are identical or not. When multiple bill denominations are specified in the dispensing command, the aggregate amount in the command is compared with the authorized withdrawal amount. If those amounts are identical, the dispenser dispenses cash following to the dispensing command.

The structure examples of the secure peripheral devices are depicted in Fig. 7. In general, an existing card reader is equipped with a slot to install a secure element for mutual

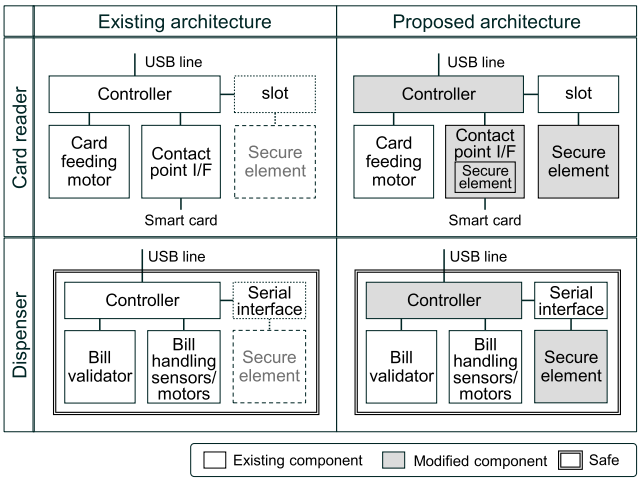


**Fig. 7** Comparison of existing devices and proposed devices.

authentication between a smart card and a terminal. The secure element in Fig. 6 can be installed in the slot. Additionally, the firmware in the controller is also supposed to be protected from unauthorized manipulation with digital signatures installed in the secure element. The firmware running on the RAM in the controller is supposed to be still secure by self-tests with the digital signatures. For example, the firmware hash is calculated periodically such as once every day in the controller. The hash is transferred to the secure element and is verified with the digital signatures. The contact point I/F (interface) to communicate with a smart card is equipped with another secure element. The secure element is cryptographically connected with the secure element installed to the slot in order to protect contents from/to a smart card from unauthorized access inside the card reader. Such structure is practical since the PCI requirements [13], [21] define similar requirements.

Concerning a dispenser, an existing dispenser is equipped with a serial interface to expand the functions in many cases. A circuit board implementing a secure element can be installed to the serial interface. The firmware in the controller is also supposed to be protected from unauthorized manipulation with digital signatures installed in the secure element as well. The firmware running on the RAM in the controller is supposed to be still secure by self-tests as well as the proposed card reader. Furthermore, the whole dispenser is protected from unauthorized physical accesses by a tightly controlled safe. Thus the firmware is logically and physically protected.

Regarding a development cost of the proposed measure, the development items are (1) DTL in the PC, (2) modification of existing firmware of the card reader and the contact point I/F implementing a secure element, and a secure element installed in the slot in the card reader, (3) modification of existing firmware of the dispenser and a circuit board implementing a secure element in the dispenser. Some country's regulations require similar implementation to (1) and (3) in ATMs for other security objectives. Concerning (2), existing device vendors provide card readers

equipped with similar components and structures to protect card holder data in conformity to PCI PTS POI [13]. Thus items (1) (2) (3) can be developed based on existing implementation and components at a reasonable cost.

### 3.3 Validation of the Proposed Measure

It is described here that the security measure proposed in Sect. 3.1, 3.2 can meet the conditions addressed in Sect. 2.3.

**(A) A security measure should not significantly impact management workloads of existing ATM operations.**
The proposed security measure can prevent unauthorized cash dispensing based on the secure peripheral devices equipped with a tamper-proof secure element. Therefore, quite heavy management workloads to tightly protect the PCs are not required.

**(B) A security measure should not significantly impact ATM system availability.**
The proposed measure does not rely on the tightly protected PC but on the secure peripheral devices. Frequent OS updating/hardening for security patch, which would significantly impact ATM system availability, is not a necessary condition in the proposed measure. FIs can take enough time to comprehensively test so many software components in the PC before releasing them to prevent occasional system troubles.

**(C) "Jackpotting" cannot be successful even though integrity of all software related to dispensing commands is not assured.**
The proposed measure can prevent "Jackpotting" without relying on integrity of all software of the PC. Even if the integrity of DTL is not assured, "Jackpotting" cannot still be successful as DTL is just a communication pass to transfer encrypted data. The proposed security measure can work as defense in depth in cases that the PC is compromised.

In this way, the proposed measure can harmonize with existing ATM systems and operations by meeting the three conditions. The requirements of the EUROPOL's guidance and recommendations do not meet the three conditions as described in Sect. 2.3. The comparison between the EUROPOL's requirements and the proposed measure is summarized in Table 1. As far as the authors investigated the existing security guidelines of other countries and ATM vendors, security company's solutions and patents, there are neither methods nor solutions meeting the conditions (A) to (C). It is noted that the proposed measure can also prevent "Black Boxing" since the proposed dispenser does not accept unauthorized dispensing commands received from an external computer as it does not receive any authorized withdrawal amounts.

**Table 1** Comparison of the EUROPOL's requirements and the proposed measure.

| Conditions | EUROPOL's requirements | Proposed measure |
|---|---|---|
| Condition (A) | Not satisfied | Satisfied |
| Condition (B) | Not satisfied | Satisfied |
| Condition (C) | Not satisfied | Satisfied |

We developed a prototype system of the proposed measure with an existing ATM system to confirm the operational feasibility. Circuit boards equipped with a Java Card™*3 based secure element were implemented into an existing card reader and an existing dispenser. We confirmed that the dispenser dispensed cash when the dispensing amount in a command received from the PC and the authorized withdrawal amount received from the card reader are identical. We also confirmed that the dispenser did not accept dispensing commands without any authorized withdrawal amount or with an altered withdrawal amount.

## 4. Application to Ticketing Devices

The concept of the proposed measure can be applied not only to ATMs but also to ticketing devices, such as railway tickets that physical media are handled in accordance with payment (Fig. 8). The fundamental framework of the proposed measure is illustrated in Fig. 8 (a). Peripheral device (i) extracts information to protect the property (vii) from unauthorized accesses in the input /output data of the device (i). Then device (i) generates information (ii) to verify the authenticity of a command (iv) accessing protected

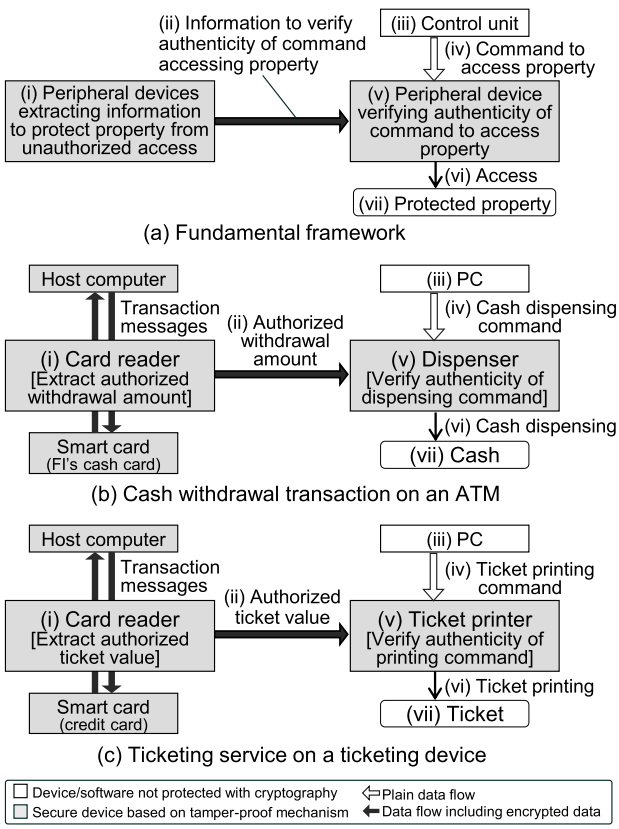


**Fig. 8** Fundamental framework and implementation examples.

property (vii). After that, device (i) sends information (ii) to peripheral device (v) accessing the protected property (vii). If the device (v) successfully verifies the authenticity of the command (iv) received from the control unit (iii) with information (ii), it accesses (v) the property (vii).

As shown in Fig. 8 (b), in a case of cash withdrawal transaction on an ATM with a smart card (FI's cash card), (i) is the proposed card reader, (ii) is an authorized withdrawal amount, (iii) is the PC, (iv) is a cash dispensing command, (v) is the proposed dispenser, (vi) is cash dispensing, and (vii) is cash respectively. As depicted in Fig. 8 (c), in a case of ticketing service on a ticketing device with a smart card (credit card), (i) is the proposed card reader, (ii) is an authorized ticket value charged to the user's credit account, (iii) is the PC, (iv) is a ticket printing command, (v) is a ticket printer verifying the authenticity of a printing command, (vi) is ticket printing, and (vii) is a ticket respectively. In this way, the concept of the proposed measure can be applied not only to ATM cash withdrawal transaction systems but also to other physical media handling systems in accordance with payment.

## 5. Conclusion

In this paper, we proposed a new ATM security measure based on the secure peripheral devices to prevent logical attacks for unauthorized cash dispensing so-called "Jackpotting" and "Black Boxing." In the security measure, the proposed dispenser verifies the authenticity of a cash dispensing command received from the PC with the withdrawal transaction evidence securely transferred from the proposed card reader of an ATM. The card reader can capture the transaction evidence since all transaction data flows through the card reader in a smart card transaction. As the proposed measure relies on security of the peripheral devices, it can work as defense in depth when the PC is compromised. The measure can also meets the three conditions so as not to impose on FIs heavy burden to tightly control the PCs though the existing measures do not meet them.

We also explained the fundamental framework of the proposed measure can be applied not only to ATM cash withdrawal transactions but also ticketing service on a ticketing device. Furthermore, we expect that the primary idea of the proposed measure can also be applied to withdrawal transactions using a magnetic stripe card, deposit transactions, money transfer transactions on ATMs. Such new measures are going to be proposed as future works.

This paper did not propose a measure to protect a withdrawal amount in plain data of a transaction request message sent from the PC to a smart card since any monetary loss does not occur even if the message is altered. That is, altered withdrawal amount in the message goes directly to the altered amount of cash dispensed to the ATM user. However, some users may be embarrassed by the unexpected cash amount and the situation would bring other frauds. Hence protecting the transaction request message is also a remaining issue that will be tackled as a future work.

## Acknowledgments

The authors would like to express our gratitude to the members of Hitachi-Omron Terminal Solutions, Corp. for supporting security measure discussions: Y. Taniyama, E. Mizuno, M. Yoshii, and Y. Horii. We are also grateful to Prof. T. Ogino, K. Ito, J. Takubo of Connected Consumer Device Security Council for discussion of security measures and security guidelines.

*1 Windows is either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
*2 EMV is a registered trademark in the U.S. and other countries and an unregistered trademark elsewhere. The EMV trademark is owned by EMVCo.
*3 Java and Java Card are registered trademarks of Oracle and/or its affiliates.

### References

[1] Symantec, "Backdoor.Padpin," Press Release, Symantec Security Response, Oct. 2014, https://www.symantec.com/security_response/writeup.jsp?docid=2014-051213-0525-99&tabid=2
[2] Kaspersky Lab., "Tyupkin Virus (Malware) | ATM Security," Press Release, https://www.kaspersky.com/resource-center/threats/tyupkin-malware-atm-security-malware
[3] Symantec Official Blog: Backdoor.Ploutus Reloaded – Ploutus Leaves Mexico, http://www.symantec.com/connect/blogs/backdoorploutus-reloaded-ploutus-leaves-mexico
[4] The Times of India, "ATM JACKPOT WITH MALWARE," TIMES NATION | Politics & Policy, May 2015, http://www.pressreader.com/india/the-times-of-india-mumbai-edition/20150509/282003260992233
[5] EUROPOL, "27 arrested in successful hit against ATM Black Box attacks," Press Release, May 2017, https://www.europol.europa.eu/newsroom/news/27-arrested-in-successful-hit-against-atm-black-box-attacks
[6] The European Association for Secure Transactions (EAST), "EAST reports 2016 crime stats for Europe's ATMs; black box attacks up 287 percent," April 2017, https://www.atmmarketplace.com/news/east-reports-2016-crime-stats-for-europes-atms-black-box-attacks-up-287-percent/
[7] European law enforcement agency, "Guidance and recommendations regarding logical attacks on ATMs," https://www.ncr.com/content/dam/ncrcom/content-type/brochures/EuroPol_Guidance-Recommendations-ATM-logical-attacks.pdf
[8] China Zhijian Publishing House, "GA 1280-2015, Security requirements for automatic teller machines" (in Simplified Chinese), http://fsms.bsmi.gov.tw/cat//opac_book/book_detail.asp?systemno=0000296088
[9] ATM marketplace, "ATMs left behind as Windows XP support ends," April 2014, http://www.atmmarketplace.com/articles/atms-left-behind-as-windows-xp-support-ends/
[10] J. Bräuer, B. Gmeiner, and J. Sametinger, "A Risk Assessment of Logical Attacks on a CEN/XFS-based ATM Platform," International Journal on Advances in Security, vol.9, no.3&4, pp.122–132, ISSN 1942-2636, Dec. 2016.
[11] CEN, "Extensions for Financial Services (XFS) interface specification Release 3.30 - Part 1: Application Programming Interface (API) - Service Provider Interface (SPI) - Programmer's Reference," European Committee for Standardization, Aug. 2015, ftp://ftp.cen.eu/

CWA/CEN/WS-XFS/CWA16926/CWA%2016926-1.pdf
[12] EMVCo, "EMV Integrated Circuit Card Specifications for Payment Systems Book 3 Application Specification, Version 4.3," Nov. 2011, https://www.emvco.com/wp-content/uploads/documents/EMV_v4.3_Book_3_Application_Specification_20120607062110791.pdf
[13] PCI SSC, "Payment Card Industry (PCI) PIN Transaction Security (PTS) Point of Interaction (POI) Modular Security Requirements Version 5.1," March 2018, https://www.pcisecuritystandards.org/documents/PCI_PTS_POI_SRs_v5-1.pdf
[14] PCI SSC, "Payment Card Industry (PCI) PIN Security Requirements Version 2.0," Dec. 2014, https://www.pcisecuritystandards.org/documents/PCI_PIN_Security_Requirements_v2__Dec2014_b.pdf
[15] ISO 9564-1:2017, ISO 9564-2:2014, ISO 9564-4:2016, "Financial services – Personal Identification Number (PIN) management and security –," https://www.iso.org/standard/68669.html, https://www.iso.org/standard/61448.html, https://www.iso.org/standard/61246.html
[16] EMVCo, LLC, "EMV Integrated Circuit Card Specifications for Payment Systems Book 2 Security and Key Management Version 4.3," Nov. 2011, https://www.emvco.com/terms-of-use/?u=wp-content/uploads/documents/EMV_v4.3_Book_2_Security_and_Key_Management_20120607061923900.pdf
[17] PCI SSC, "Payment Card Industry (PCI) Data Security Standard Requirements and Security Assessment Procedures Version 3.2.1," May 2018, https://www.pcisecuritystandards.org/documents/PCI_DSS_v3-2-1.pdf
[18] ISO 11568-1:2005, ISO 11568-2:2012, ISO 11568-4:2007, "Banking – Key management (retail)," https://www.iso.org/standard/34937.html, https://www.iso.org/standard/53568.html, https://www.iso.org/standard/39666.html
[19] ANSI X9.24-1-2017, ANSI X9.24-2-2016, ANSI X9.24-3-2017, "Retail Financial Services Symmetric Key Management," https://webstore.ansi.org/RecordDetail.aspx?sku=ANSI+X9.24-1-2017, https://webstore.ansi.org/RecordDetail.aspx?sku=ANSI+X9.24-2-2016, https://webstore.ansi.org/RecordDetail.aspx?sku=ANSI+X9.24-3-2017
[20] ANSI X9.63-2011 (R2017), "Public Key Cryptography for the Financial Services Industry - Key Agreement and Key Transport Using Elliptic Curve Cryptography," https://webstore.ansi.org/RecordDetail.aspx?sku=ANSI+X9.63-2011+(R2017)
[21] PCI SSC, "Payment Card Industry (PCI) Point-to-Point Encryption: Solution Requirements and Testing Procedures Version 2.0 (Revision 1.1)," July 2015, https://www.pcisecuritystandards.org/documents/P2PE_v2_r1-1.pdf

**Hisao Ogata** received the B.S. and M.S. degrees from Kyusyu University in 1987 and 1989, respectively. During 1989-2004, he worked for Hitachi, Ltd. researching neural networks and pattern recognition. He has been working for Hitachi-Omron Terminal Solutions, Corp. developing biometrics and ATM security since 2004. He is a Ph.D. candidate of Yokohama National University, a member of IEICE and ISO/IEC JTC1 SC37.

**Tomoyoshi Ishikawa** received the B.S. degree in Informatics and Mathematical Science from Kyoto University in 2001. During 2001-2004, he worked for OMRON Corporation, developing ATM software. He has been working for Hitachi-Omron Terminal Solutions, Corp. designing ATM software and its security since 2004. He is a member of CEN Workshop on Extensions for Financial Services (CEN/XFS).

**Norichika Miyamoto** received the B.S. and M.S. degrees from Kyoto University in 1984 and 1986, respectively. During 1986-1987, he worked for Nippon Kokan Corporation, developing process control system for steel manufacturing. During 1987-2004, he worked for OMRON Corporation, developing Unix workstation software and operating system for multiprocessor system. He has been working for Hitachi-Omron Terminal Solutions, Corp. developing ATM devices and constructing oversea business strategy since 2004.

**Tsutomu Matsumoto** is a professor of the Graduate School of Environment and Information Sciences, Yokohama National University and directing the Research Unit for Information and Physical Security at the Institute of Advanced Sciences. He received Doctor of Engineering from the University of Tokyo in 1986. Starting from Cryptography in the early 80's, he has opened up the field of security measuring for logical and physical security mechanisms. Currently he is interested in research and education of Embedded Security Systems such as IoT Devices, Network Appliances, Mobile Terminals, In-vehicle Networks, Biometrics, and Artifact-metrics. He is serving as the IEICE Technical Committee on Hardware Security, the chair of Japanese National Body for ISO/TC68 (Financial Services), and a core member of the Cryptography Research and Evaluation Committees (CRYPTREC). He was a director of the International Association for Cryptologic Research (IACR) and the chair of the IEICE Technical Committee on Information Security and served as an associate member of the Science Council of Japan (SCJ). He received the IEICE Achievement Award, the DoCoMo Mobile Science Award, the Culture of Information Security Award, the MEXT Prize for Science and Technology, and the Fuji Sankei Business Eye Award. the Fuji Sankei Business Eye Award.