INTERNATIONAL STANDARD

ISO/IEC 23465-1

First edition
2023-02

Card and security devices for personal identification — Programming interface for security devices —

Part 1: Introduction and architecture description

Cartes et dispositifs de sécurité pour l'identification personnelle — Interface de programmation pour dispositifs de sécurité —

Partie 1: Introduction et description de l'architecture

Reference number
ISO/IEC 23465-1:2023(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see https://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 17, *Cards and security devices for personal identification*.

A list of all parts in the ISO/IEC 23465 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

Integrated circuit card (ICC) technologies and solutions are widely deployed around the world, but systems for identity tokens and credentials are quickly changing. In this context, the application protocol data unit (APDU) protocol defined in the ISO/IEC 7816 series is becoming, in some cases, a hindrance to the integration of integrated circuits (ICs) (as security devices) in environments such as mobile phones, handheld devices, connected devices (e.g. M2M, IoT) or other applications using security devices.

Several stakeholders are not familiar with, or not very fond of the APDU protocol because of its complexity. They will often circumvent its constraints by requesting an abstraction layer hiding IC specifics. Although the security mechanisms of security devices are well defined in ISO/IEC 7816-4 their implementation and application differ from vendor to vendor and the complexity overstrains most of the application developers.

In software development, a common way to simplify the usage of complex systems is the definition and application of application programming interface (API) functions to access the IC within the devices. Specific knowledge of APDU protocols and details of the IC implementation is not necessary anymore. Also, the complexity and details of the implementation of the security model and the security policy can be shifted from pure application development into system design of the electronic device and its related software.

Therefore, this document is geared towards software (SW)-architects, application programmers or specification developers developing software applications using and addressing ICs as security devices within operating systems or their components.

The projected applications can run on different software and hardware environments. Generalisation of the API definition is key and the dependencies on specific runtime environments and equipment are kept out in principle.

Existing runtime environments already support the access to IC as security devices using different specific APIs, e.g. OpenMobileAPI,[10] PKCS#11,[12] but they always implement a proprietary interface and middleware, which is not commonly applicable. However, even solutions based on those kinds of middleware are perceived as cumbersome in some systems. The market looks for a middleware memory footprint to be as low as possible. This document also aims to overcome or mitigate those issues by proposing a new approach that would preserve ICC functionality and allows for a seamless ICC portability onto new systems.

Since the system is designed for easy support by mobile operation systems, mobile operating system (OS) designers/ implementers are encouraged to support these standardized APIs to access any embedded secure element (eSE) within the mobile device.

In the context of mobile devices, there is a necessity for trusted computing, e.g. by dedicated security hardware. The proposed API helps the application implementer with a standardized common interface to such trusted IC.

The ISO/IEC 23465 series focuses on a solution by designing an API and a system with the following characteristics.

- It offers a set of API calls related to multi-sectorial ICC functionality, derived from the ISO/IEC 7816 series and other ICC related standards.
- It defines the sub-system to perform the conversion from the API function to the interface of the security device (e.g. APDU-interface), called Proxy.
- It results in a description of solutions with no middleware or very little middleware memory footprint (i.e. simplified drivers).
- It defines the simplified ICC capabilities, the discoverability (i.e. with significantly less complexity than ISO/IEC 24727) and examples of usages.

The ISO/IEC 23465 series is comprised of three parts each focusing on a specific topic:

— ISO/IEC 23465-1 (this document): provides an introduction to the series and a short overview of the architecture;

— ISO/IEC TS 23465-2: defines the API for client applications allowing incorporation and usage of security devices;

— ISO/IEC TS 23465-3[1]: describes the software called Proxy which provides different services e.g. to convert the API calls into serialized messages to be sent to the security device.

[1] Under preparation. Stage at the time of publication: ISO/IEC DTS 23465-3.

Card and security devices for personal identification — Programming interface for security devices —

Part 1: Introduction and architecture description

1 Scope

This document introduces and describes the concept of the application programming interface (API) to security devices with the intention to simplify the usage of commands and mechanisms defined by the ISO/IEC 7816 series.

This document gives guidelines on:

- the system overview and description of the system of the programming interface;
- the architecture description;
- the data model in general, used by the API;
- the use cases and the usage model of the API.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at https://www.iso.org/obp
- IEC Electropedia: available at https://www.electropedia.org/

3.1
client
any type of entity requesting services from a *security device* (3.7)

3.2
ISO/IEC 23465 API
software interface defined in ISO/IEC TS 23465-2

3.3
middleware
software (SW) component allowing two systems from different or similar operating systems interconnection (OSI) layers to communicate with each other

3.4
operating systems interconnection model
OSI model
conceptual model that characterizes and standardizes the communication functions of a network or computing system without regard to its underlying internal structure and technology

3.5
secure digital memory card
SD-card
secure storage media using non-volatile memory

3.6
proxy
sub-system to perform conversion from the application programming interface (API) function to the interface of the *security device* (3.7)

3.7
security device
tamper-resistant secure hardware component which is used in a device to provide the security, confidentiality, and multiple application environment required to support various business models

Note 1 to entry: It may exist in any form factor, e.g. embedded or integrated SE, SIM/UICC, smart card, SD-card.

3.8
serialization
translation of data structures or object state into an octet string for transmitting or storing

3.9
trusted execution environment
TEE
aspect of the mobile device comprising hardware and/or software which provides security services to the mobile device computing environment, protects data against general software attacks and isolates hardware and software security resources from the operating system

[SOURCE: ISO 12812-1:2017, 3.60]

3.10
use case
list of actions or event steps typically defining the interactions between a role and a system to achieve a goal

Note 1 to entry: A role is known as an actor in the Unified Modelling Language.

4 Symbols and abbreviated terms

APDU	application protocol data unit
API	application programming interface
BLE	bluetooth low energy
CBOR	concise binary object representation
DF	dedicated files
eSE	embedded secure element
eSIM	embedded subscriber identity module
IC	integrated circuit
ICC	integrated circuit card
IDL	interface description language
iSIM	integrated subscriber identity module
I^{2}C	inter-integrated circuit
JSON	java script object notation
NFC	near field communication
PCB	printed circuit board

PC/SC	personal computer/smart card
OSI	open systems interconnection
OTA	over the air
SD	secure digital (memory card)
SE	secure element
SIM	subscriber identity module
SoC	system on chip
SPI	serial peripheral interface
SW	software
TEE	trusted execution environment
UICC	universal integrated circuit card
USB	universal serial bus
WIFI	wireless communication technology, defined by the Wifi consortium

5 System overview

5.1 Conditions of use

The utilisation of an API for security devices defined in the ISO/IEC 23465 series of standards is useful in any client application software which needs services of a security device. The application software may run on any electronic devices, e.g. personal computers, terminals or mobile devices. The electronic device contains or is connected to a security device and allows its access by client applications. It is assumed that any software running on the electronic device is separated into several logical or functional layers. Such layers may be designed as a middleware between the client application software and the related security device or may be provided by the device's operating system. API is herein defined as generalised function calls from the client application software to the additional layers in the system.

The API allows the logical access to any available security device, independent from the physical form factor, the technology, the used connectivity and the applied transmission protocol. It hides the physical layer, the data link layer and the network layers of OSI model to the application. The API offers standardized methods and functions to security device services and builds either an abstraction or a subset of the underlying security device interface, e.g. the APDU interface defined by ISO/IEC 7816-4, or both. In this way, the transport, session and presentation layers are also hidden from the client application.

The API is a representation of the application layer to a security device. The application implementer does not need further details to contact, address, select and use a connected security device. But the set of the API functions still allows the application to work with the security device by retrieving all relevant information, functionality and services.

Systems implementing this API facilitate the access to the security device. Dedicated function calls and specific knowledge about the structure and architecture of the security device's application is not needed by the client application programmer. The API generalises function calls, offers less complexity and reduces the need of knowledge about details of the security device.

The API functions are resolved within an additional software, which handles the access to security devices. These additional software components are provided by the manufacturer of the electronic device. Since the number of security devices in a system is not limited, the additional software components have to be enabled to handle the different involved security devices.

Conversely, several client applications can use a single security device. It is possible for each client application to use a different application inside the security device. Furthermore, applications inside a security device are separated from each other from a security point of view.

5.2 Simple system configuration

The simplest possible configuration is outlined Figure 1. This kind of system consists (possibly among other components) of just one software environment running any application and incorporates the security device. Such configuration is, e.g. a SoC device proposed for mobile devices. In this case, the mobile device environment contains the complete client application and the security device for which the implementation is running on the same system and within the same runtime environment as the client application itself.

API calls from the client application directly lead to function calls in the security device. The security device implements the resolution of the API and handles the processing of the requested operation on the security device. Thus, from a client application programmer's point of view, the API calls from the client application just call the corresponding method implementation of the security device.

In this simple situation, there is neither any other layer in between nor is there any kind of message serialization. The API calls and the resolution of the API calls are done in the common system. To achieve this, the security device programmer normally provides a library to the system manufacturer or application.

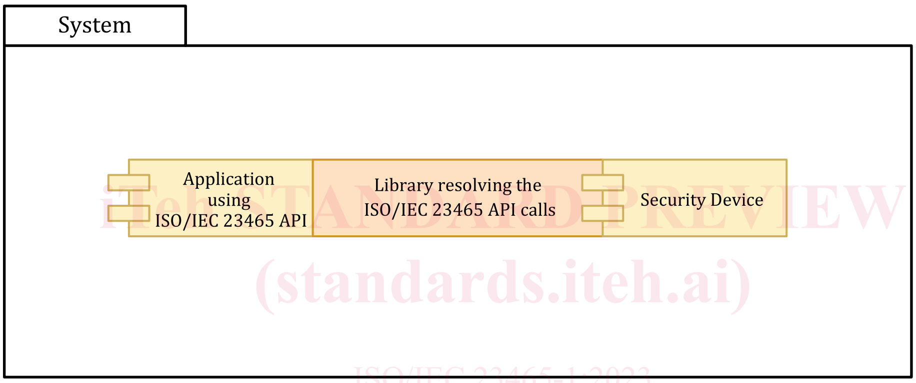
System
Application using ISO/IEC 23465 API
Library resolving the ISO/IEC 23465 API calls
Security Device

Figure 1 — Simple system configuration

5.3 Complex system configuration

Compared to the simple configuration in Figure 1, the API defined in the ISO/IEC 23465 series of standards applies also to configurations where a system contains more than one client application or more than one security device, as outlined in Figure 2. The API supports the possibility that more than one security device is available. It contains methods to select a specific security device among the available ones. This kind of selection requires another component, called Proxy, mediating between client applications and security devices. The client application calls API methods which trigger the Proxy and address the appropriate requested functionality in the selected security device.

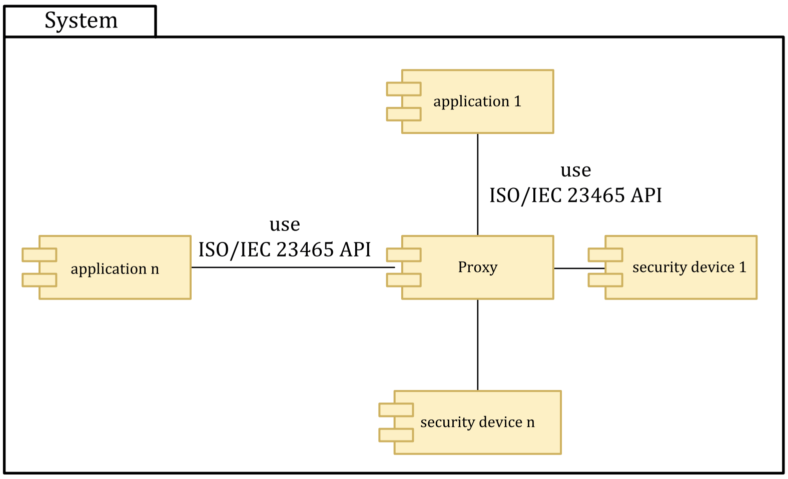

Figure 2 — Complex system configuration

Even in complex system client applications, security devices and the Proxy may run in the same runtime environment. Typically, a security device is deployed as dedicated hardware. If so, client applications and (part of) the Proxy on one hand runs in a runtime environment which differs from the security device.

5.4 Generic examples using the different configurations

5.4.1 SoC — Example of simple system configuration

The simplest configuration of a system using security devices is a system running client applications and security devices in the same runtime environment (see 5.3). A library within the runtime environment is supposed to act as the Proxy between the client application and the security device. This library performs the API call resolution and the security device access.

5.4.2 ID-systems

A slightly more complex system runs client applications and security devices in different runtime environments with separated software layers. Examples are ID-systems in general (e.g. border control, banking, health) typically holding a client application as a part of a host connected to a card reader. Security devices in this configuration are ID-cards temporarily connected to the ID-system. To simplify the development of such systems, it would be appropriate not only to standardize the interface between the client applications and the Proxy, but also to standardize the communication between the Proxy and the security devices.

Figure 3 depicts a generic simple system showing the conditions of usage with an client application using the API defined in ISO/IEC TS 23465-2 within a single runtime environment. The Proxy can be outlined as a library or as a module in a more complex reader operating system.

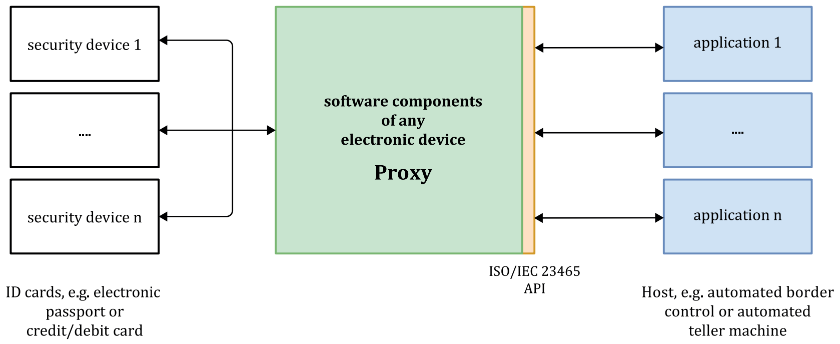
security device 1
....
security device n
software components of any electronic device
Proxy
application 1
....
application n
ISO/IEC 23465 API
ID cards, e.g. electronic passport or credit/debit card
Host, e.g. automated border control or automated teller machine

Figure 3 — ID-system with host and card reader

5.4.3 Mobile devices with multiple security devices

Figure 4 shows an example of a complex system configuration using different security devices within a mobile device (e.g. mobile phones or tablet computers). The mobile app, downloaded from an app store, requests, in the course of the client application, functionality from security devices by API function calls. The software system beneath the mobile app, introduced as Proxy in the ISO/IEC 23465 series, handles the function calls and transforms or serializes them into a data stream understandable by the security devices. Depending on the implementation of the security devices, the function calls are operated by the addressed security device and the results/responses are transferred and transformed adequately to the calling mobile app. Within mobile devices, usually a fixed number of security devices are permanently connected, e.g. USIM, ISIM, eSIM, eSE, which can be used by different mobile apps loaded on the mobile phone.

The Proxy can use additional APIs to achieve the physical access to the security device, e.g. provided by the mobile operating system. Applications running on personal computers or similar devices can use additional existing SW drivers, e.g. PC/SC, which can facilitate the access. Security device form factors of mobile devices beside SIMs are, e.g. soldered ICs on the PCB, connected USB-devices, SD-cards or connected ID cards via card reader interfaces (NFC, BLE).

ISO/IEC 7816-4 defined APDUs or other representation of messages are transmitted by the lower level transmission protocols determined by the technology used in the addressed security device. Examples for this protocol types are, e.g. USB, I^{2}C, SPI or serial communication according to ISO/IEC 7816-3, representing the physical layer in the OSI model.

The API hides all these security device details completely from the client applications. ISO/IEC TS 23465-3 gives examples on how a Proxy implements the different layers.