TECHNICAL SPECIFICATION

ISO/IEC TS 23465-3

First edition
2023-05

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

Part 3: Proxy

Cartes et dispositifs de sécurité pour l'identification personnelle — L'interface du logiciel pour dispositifs de sécurité —

Partie 3: Proxy

Reference number
ISO/IEC TS 23465-3:2023(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 outlined in the ISO/IEC 7816 series is becoming, in some cases, a hindrance to the integration of integrated circuits (ICs) in environments such as mobile phones, handheld devices, connected devices (e.g. M2M, IoT) or other application using security devices.

Stakeholders often request an abstraction layer hiding IC specifics to avoid the complexity of APDU protocols. 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 aims to overcome or mitigate those issues by proposing a new approach that would preserve ICC functionality and allow for a seamless ICC portability onto new systems.

The ISO/IEC 23465 series focuses on a solution by designing an application programming interface (API) and a system with these characteristics:

— It offers a subset, from the ISO/IEC 7816 series, of mostly used multi-sectorial ICC functions.

— It results in no further middleware or very little middleware memory footprint (i.e. simplified drivers).

— It requires a simplified ICC capability discoverability (i.e. with significantly less complexity than ISO/IEC 24727-1). [3]

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

— ISO/IEC 23465-1: 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 (this document): describes the software (SW) called "proxy" which provides different services, e.g. to convert the API calls into serialized messages to be sent to the security device.

The ISO/IEC 23465 series is intended to be used by any sector relying on the interchange defined, but not limited to, the ISO/IEC 7816 series.

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

Part 3: Proxy

1 Scope

This document describes the software (SW) layer called "proxy". It supports the programming interface to security devices and the application using this API to access the application related security devices defined in ISO/IEC TS 23465-2.

This document is applicable to:

— proxy requirements, functionality and layers;

— resolving mechanisms for API functions;

— data structures related to security device handling;

— translation for security device communication;

— serialization/de-serialization syntax and methods.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 23465-1, *Card and security devices for personal identification — Programming interface for security devices — Part 1: Introduction and architecture description*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 23465-1 and the following 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
access rule
data element containing an access mode referring to an action and security conditions to be fulfilled before acting

3.2
client application
implemented application using the ISO/IEC 23465-API requesting services from a security device

3.3
security attribute
condition of use of objects in the card including stored data and data processing functions, expressed as a data element containing one or more *access rules* (3.1)

4 Symbols and abbreviated terms

APDU	application protocol data unit
API	application programming interface
ATR	answer to reset
CBOR	consise binary object representation
DO	data object
ICC	integrated circuit card
JSON	JavaScript open notation
MSE	manage security environment
OMAPI	open mobile API
OSI	open systems interconnection
PACE	password authenticated connection establishment
SM	secure messaging
SW	software
SD	security device, see terms and definitions

5 Proxy related requirements

5.1 Link between client application and security device

The proxy is the piece of software (SW) between the programming interface used by a client application dealing with security devices and the related security device(s). The functionality of the proxy and its APIs deal with communication, connectivity, provisioning of confidentiality and authenticity of the connection to the security device and its security device applications.

The general concept of client applications dealing with a set of security device is outlined in ISO/IEC 23465-1 and is depicted in Figure 1 (see also ISO/IEC 23465-1:2023, Figure. 2). The applications or clients use abstractions of access methods to the security devices, which are outlined as the ISO/IEC 23465-API in ISO/IEC TS 23465-2. Details of low-level security device addressing and accessing methods are kept hidden to these clients. The translation of the abstract API-functions into low-level commands and protocols shall be performed by the proxy, possibly supported by additional libraries and internal APIs.

The clients only have access to the physical security devices by the proxy which acts as a dispatcher for the direct access to the security devices and the related security device applications. In some use cases the proxy may handle as a multiplexer, especially in multitasking environments with parallel running clients.

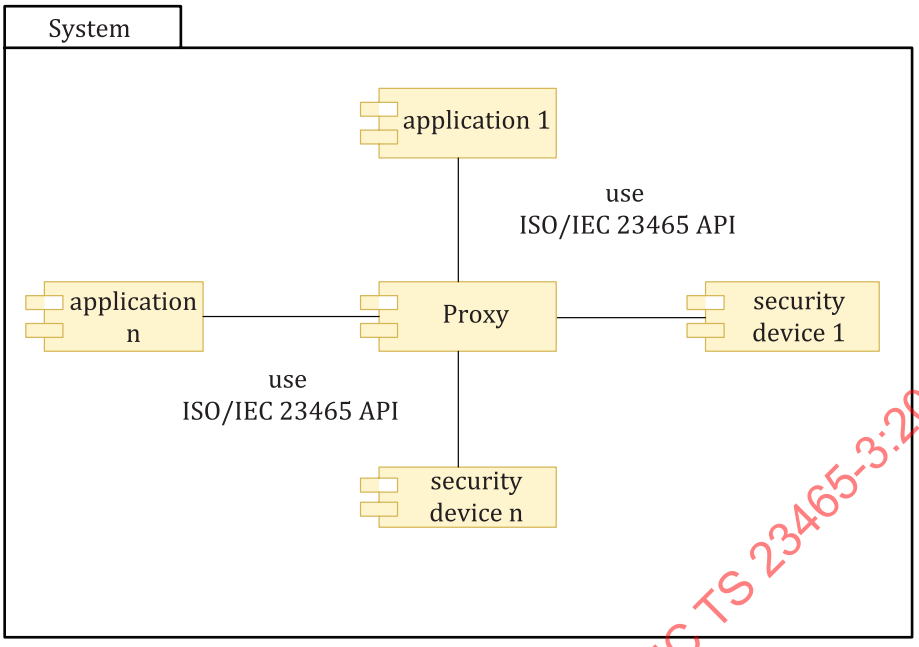

Figure 1 — Components of the system with distributing proxy

5.2 Proxy layers

5.2.1 Proxy layer model

Figure 2 depicts the layers and components of the proxy to perform the client application requests by the defined API. The proxy as a SW component requires some additional SW parts in a real implementation. These are not outlined here, but are mentioned separately.

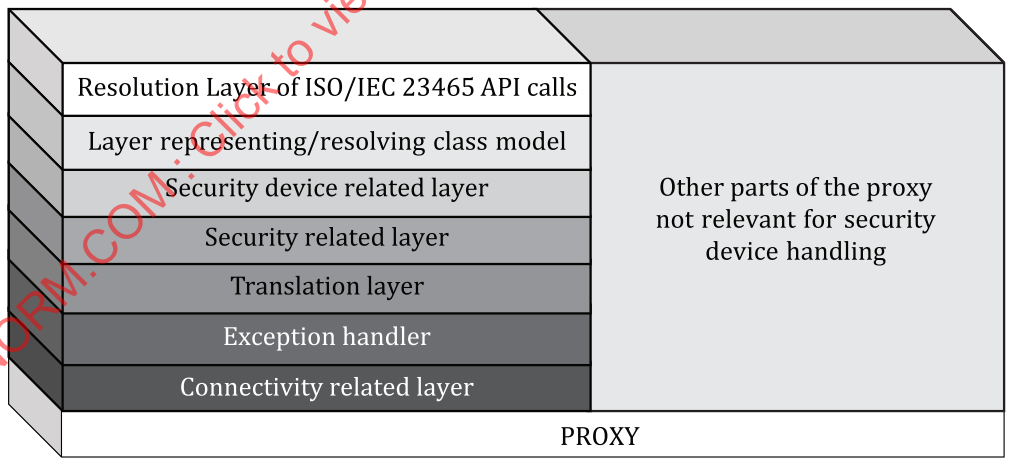

Figure 2 — Components of a proxy

5.2.2 Resolution layer of ISO/IEC 23465-API calls

Any API call needs a piece of programming code, called glue code, which has no functional relevance but is essentially to adhere to the different code parts. Any real implementation resolves, with this additional glue SW, the requirements of the applied programming language.

When several client applications are allowed to act in parallel, this layer shall be able to handle several tasks independently.

5.2.3 Layer representing/resolving class model

The class model, outlined in ISO/IEC TS 23465-2, represents the view of the client and part of the proxy to the object-oriented definition of a security device and its components. The client uses the references and methods of the classes provided by the instantiated objects. The instantiation of the objects is done in this layer. The proxy provides the object references to the client, administrates the objects in accordance to the resources of the systems, e.g. the available or addressed security devices and its related data.

5.2.4 Administration layer for security devices

All relevant data of the administrated and addressable security devices are handled in this layer. The layer tallies the security devices, identifies the access methods, administrates the security device individually and stores all relevant information of the available security devices.

Means for such administration is an internal registry holding all the information of available security devices. In case of removable security devices, this layer shall register when mounting and de-register when detaching the security devices. The registry entries for such devices has to be updated accordingly.

NOTE The entries of such a registry are used to inform the client about the details of the requested security devices.

5.2.5 Security related layer

This layer provides the means to perform all the security mechanisms related to the security devices.

Any security device supports or requires security features, directly or indirectly. Depending on the security attributes of the security device application, these security-related activities have to be provided or performed, sometimes in cooperation with the client application, sometimes hidden to the client application. In the latter case the security activities shall be performed by this proxy layer.

Relevant information of fulfilled security features shall be stored, in case this information is required for further security device access.

A security device requires sometimes an authentication procedure according to external specifications, to set-up a secure channel, which are be completely hidden to the client application. In this case a session management has to be considered in the security related layer.

5.2.6 Translation layer

The most important abstraction of the access methods to security devices is the concealment of the direct communication, offered by the API. As outlined in ISO/IEC 23465-1, the evolution of security device technologies can require different protocols and data formats than those currently being used. The conversion into protocols understood by the used security device shall be done in this layer.

The translation layer provides the means to perform the translation of the client application's initiated activity into security device related protocols and formats. An abstract API call from the client application finally results in at least one command understood by the addressed security device.

In contrast, responses of the security device after processing of the command(s) are returned in the format of the used security device protocol. This layer re-translates the replied data into the formats defined by the API which are finally expected and understood by the client application.

The translation process is understood as a serialisation/marshalling and de-serialisation/unmarshalling technique, which is well-known in SW technology and used by many computer languages.

NOTE The actual protocol dealing with security devices based on ICC technology uses APDU, defined by ISO/IEC 7816-4. Upcoming application specifications use, e.g. CBOR, JSON or other coded protocols which can be relevant in the future.

5.2.7 Exception handler

Most of the API functions defined in ISO/IEC TS 23465-2 expect exception messages. Dedicated return values of the security devices, e.g. defined in ISO/IEC 7816-4, shall be translated by the translation layer. In case of errors or unforeseeable situations, the program flow is interrupted by throwing an exception This is managed by this exception layer.

The service can be used in conjunction with the translation layer since a response of a security device can require exception handling.

5.2.8 Connectivity related layer

The physical connection of the security device in the system has to be used by the proxy. Depending on the runtime environment of the used system, security device/reader-related device drivers or middlewares are available. This layer uses existing APIs which may be manufacturer dependent and requires a system related implementation. Some systems offer standardized APIs which can facilitate the interaction between proxy and security devices, e.g. the Global Platform Open Mobile API[6].

5.3 Conditional proxy functionality

5.3.1 Multitasking/Multiplexing

ISO/IEC 23465-1 outlined different runtime environments in which client applications can use the ISO/IEC 23465-API. The behaviour of a proxy varies from either supporting only one security device by one client, or one client with access to several security devices (in a single task), or several clients with access to several security devices in parallel in a multitasking runtime environment.

According to the needs of the latter situation, the proxy shall be designed as a SW running in a multitasking environment. When several clients interact with one security device simultaneously, the proxy shall obey the ability of the security device. An existing solution for such simultaneous access on a single security device is possible today when logical channels are supported. The approach of multiple logical interfaces in addition to logical channels also allows the parallel processing of security device applications in a single physical security device.

The security device's behaviour shall be managed by the proxy. Logical channel or multiple logical interface support can be part of a discovery mechanism of the proxy or a specific information in the proxy's registry of security devices.

5.3.2 Crypto functionality

Many API functions defined in ISO/IEC TS 23465-2 deal with crypto-functionality related to the security device. Addressing means used by the client application and the internal management within the security device usually differs. The knowledge about the behaviour, the needs of crypto protocols and the translation into security device related formats is provided by this service.

The proposed API methods for crypto operations cover all algorithms and mechanisms which are currently in common usage. Since the calculation is performed on the security device, the proxy shall adopt and structure the data in a way that the addressed security device can use the data.

Some crypto API calls are separated into calls for initialization of the method followed by one or more subsequent calls of the crypto API for processing. The sequence of APIs shall be controlled by the proxy.

Existing crypto protocols performed on security devices need additional commands to set-up a crypto-functionality. ISO/IEC 7816-4 defines concepts and commands which are used to set-up the security environment (MSE command) with essential information, expected by the security devices in advance. These activities should be completely hidden to the client application. The application of such security device related commands shall be performed by the proxy.

5.3.3 Discovery functionality

ISO/IEC TS 23465-2 offers the API call **isoIec23465_getSDList** which provides the security device attributes of the accessible security devices. This set of information shall be collected and distributed by the proxy. The discovery service is responsible for the collection and distribution of the relevant attributes.

This can be performed by specific discovery mechanisms related to the security devices known by additional interfaces, middlewares or other means. A possible registration service (see 5.3.4) with a persistent registry can be an additional source of information to be filled in the list of security device attributes.

5.3.4 Registration functionality

This service may be established to administrate the connected security devices. It allows the storage of security device related information in a registry of the proxy. The service periodically polls the existence and the identification of security devices in the system and updates the registry accordingly. This behaviour is also important for removable devices.

The registration and discovery functionality services are normally handled in conjunction with each other.

6 Instantiation of class objects

6.1 Class model

The class model for the API was introduced in ISO/IEC TS 23465-2, together with the object handling and the resolution of the instances. Figure 3 (see also ISO/IEC 23465-1:2023, Figure 7) depicts the proposed way to handle the API calls in the proxy.

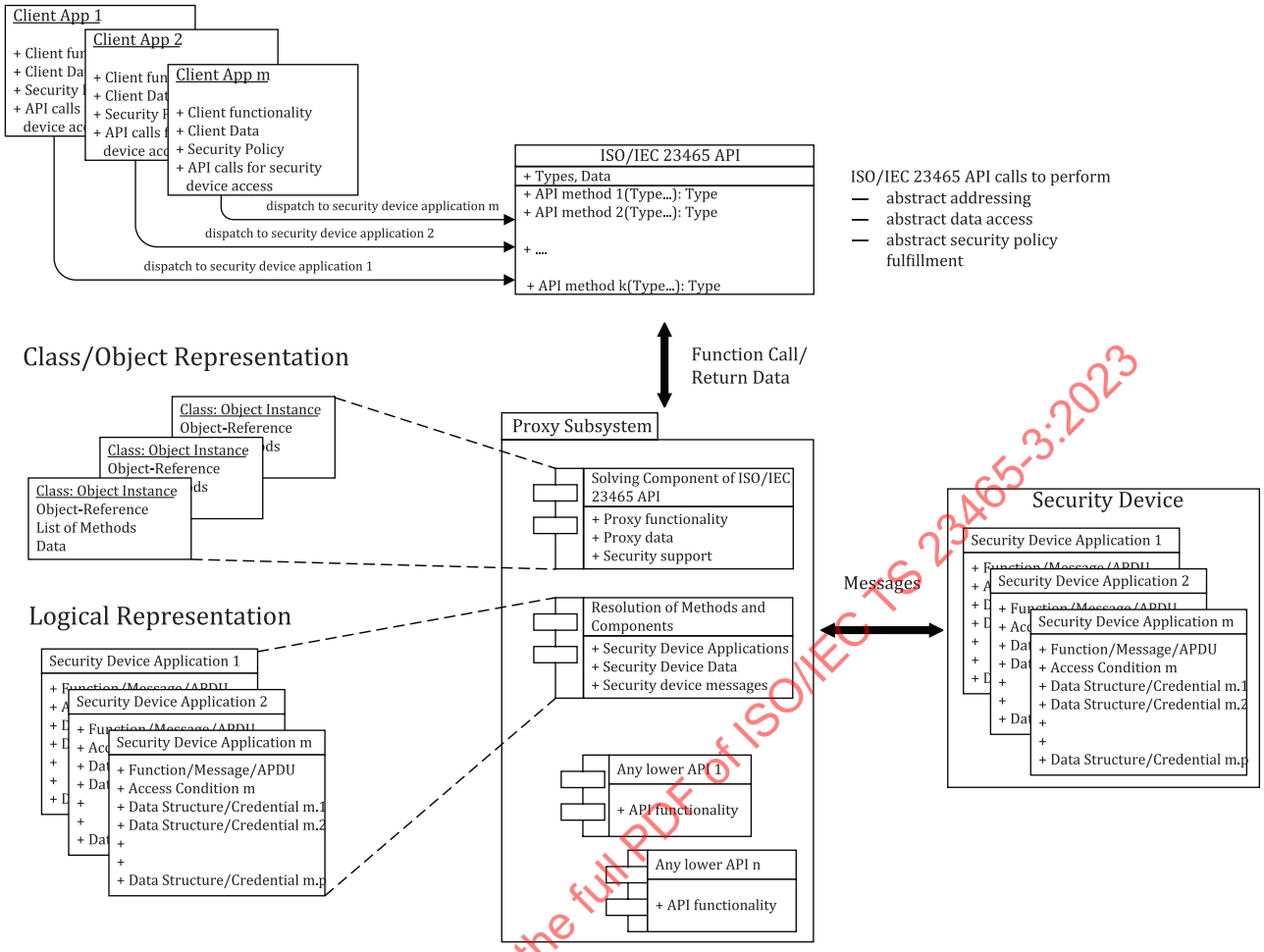

Figure 3 — Class model in context of the API and the security device

The client applications use the ISO/IEC 23465-API to handle the user functionality. The application refers to the class definitions and establishes the access to the objects defined in the class model via the API calls to the proxy. The calls are handled in the proxy's resolving layer (see 5.2.2). Based on the references of the instantiated objects, subsequent API calls on these objects are applicable.

The proposed class model and its possible usage reflects the structure and the content on the security device. Any client application can only use objects and methods which are supported by the underlaying security device. The proxy is the entity which conveys the application request, e.g. addressing of objects/methods and finally transforming into adequate format to the security device.

The real objects on the security device have their counterparts in the proxy as objects defined in the class model. Usable methods reflect the set of commands of the security device. The proxy resolves the usage of methods into sequences of security device commands, at least into one command.

An example is provided in Annex A.

6.2 Process of instantiation

After identifying the corresponding security device (in case more than one are available) the client application starts to interact with the security device by instantiating the SdApplication related to the client application. The link and binding of the instance of this object to its physical counterpart on the security device shall be established by the proxy.

The underlaying resolution layer detects the application on the security device by the selection mechanism of ISO/IEC 7816-4. The proxy generates the reference of this object and links its methods to the command set of the client related application on the security device.

According to the results of this process the proxy collects all relevant information:

— The object reference becomes valid, when the security device application exists.

— Additionally, available information, e.g. data objects of the FCI, application related access conditions, are collected and assigned to the object/reference.

— The client application retrieves the reference of the instance.

— The client application addresses any further calls/methods via the instance reference.

NOTE The logical and physical interaction with the security device through a communication layer (e.g. APDU) is handled by the proxy and its subsequent layers. The usage of other APIs providing connection to the security device (e.g. Open Mobile API) is possible.

The proxy handles the memory requirements and its allocation related to its services autonomously, completely hidden to the client applications. The client application only takes care of its own memory requirements.

6.3 Life cycle of an instance

Objects of a client application instantiated according to 6.2 are directly linked to the physical content of the security device. A request of instantiation by the client application in the first hand generates the reference to this object.

In general, a reference and the instance to such object shall be usable by the client application until the session of the client application is closed or the object and its ressources are released. A destruction of the object as a method is not part of this document.

In complex system architectures the proxy shall administrate the list of objects and its references for each security device instantiated by each client application separately. Closing a client application, in case of multi-tasking environments, shall release all the references related to that client application.

NOTE In case of a simple system architecture with only one client application and one security device, the life cycle of all instantiated objects is terminated at the latest by ending the session.

7 Services of the proxy

7.1 Security device management

7.1.1 Identification of security device

One of the main functions of the proxy is to manage and administrate the security devices within the system.

In a system with fixed installed security devices a proxy may get the information by install parameters offered by the system directly or by additional APIs (e.g. OMAPI) from an underlaying operation system (e.g. Realtime OS, mobile phone operating systems) conveying the relevant information.

The ISO/IEC 7816 series defines the ATR as a first identification means for an ICC or the EF.ATR to provide additional information. Since a security device in a running system may be not reset (e.g. in a mobile device) not all information is available in all cases. ISO/IEC 7812-1 defines additional identification means, e.g. issuer identification number, which can be used for identification of a security device. All this information may be used in closed systems.

ISO/IEC TS 23465-2 defines a set of information collected in a table of security device attributes which describes the ability and content of the security device in a flexible and detailed way. This information shall be used internally by the proxy and delivered to the external world/client applications via the API call. Any security device should store a describing data object which offers the required information. This identification means shall be initialized and personalized by the provider of the system. The following requirements should be considered:

— The proxy stores the security device attribute persistently for each controlled security device.

— Each security device stores the DO "security device attribute" and the proxy initiates the retrieval of the information autonomously.

The information is managed in the administration layer of the proxy (see 5.2.4) and is stored by the registration service (see 5.3.4).

7.1.2 Retrieval of security device information

The registry of the proxy stores administration data of the onboard security devices which allows the identification and accessibility of the security device.

Table 1 shows what the registry should contain.

Table 1 — Registry information

Content	Meaning	Size	Example
Security device enumeration	Unique number for addressing	integer	1, 2, 3...
Security device attributes	defined in ISO/IEC TS 23465-2:2023, Table 2	variable	See ISO/IEC TS 23465-2
Connectivity/routing information	Addressing information used of the proxy	variable	Reader information, network address, port information, API calls to the underlaying operating system

The storage area for the registry depends on the system architecture and may be managed dynamically, especially in case of removable security devices.

The registry content shall be stored after retrieval:

— The proxy activates the internal known security devices and reads the DO of security device attributes.

— The proxy addresses the available security devices via the underlaying OS and get the security device attributes.

— The proxy gets the registry information in the initialization/personalisation/administration phase by a production or TSM service and stores it internally.

— The proxy retrieves/removes security device attributes of removable security devices and administrates the registry.

7.1.3 Security device related functionality

The ISO/IEC 23465-API hides the characteristics and the behaviour of the underlaying security devices. The access to the physical security device finally shall be performed by the proxy which has to know how to deal with it.

The security device characteristics are divided into the following:

— Logical command set: Each security device may use a different command set. The related layer in the proxy utilizes the appropriate command definitions when the API calls have to be resolved.

— Connectivity characteristics: The proxy uses the connectivity characteristics to connect the security device. This can be API calls (e.g. OMAPI) to connect the security device drivers/middleware defined by the security device issuer, interface functions and others.

— Transmission characteristics: If the transmission is performed by the proxy, the transmission details are used to convey the security device command to the lower OSI-levels.

7.2 Security related layer

7.2.1 General

One of the main reasons for using a security device in general is to provide the overall security for any application and to allow storage of confidential information in a safe and secured way. The client applications are aware of the security requirements of the application and apply for security related services of the security device.

The role in which the proxy may act, depends on the architecture of the system. Several designs can be considered.

— The proxy has no evidence of the security needs of the client application and can only support the security related API calls in accordance of the class model and the related methods.

— The proxy acts partially as security agent, who performes/provides some low-level security protocols, e.g. a PACE protocol.

— The proxy is completely trustworthy and acts as the security counterpart to the security device, e.g. authentication, secure messaging (SM) possible actions performed by the proxy.

Depending on the design of the system, the characteristics and the security requirements of the proxy can be different.

The access to security device resources based on the fulfilment of the security attributes is controlled by the security device only. ISO/IEC 7816-4 describes mechanisms and the methodology of access control within a security device. The security device application controls the access rules and conditions to its ressources. Access is only allowed when the conditions are fulfilled.

The coding of the access rules is sometimes complex and can lead to obstacles for applications, since the interpretation is difficult. A main goal of the ISO/IEC 23465 series is to facilitate and relieve a client application from access rule handling.

7.2.2 Security conditions and fulfilment

7.2.2.1 Security conditions handled by client application

Any **object23465** provides methods to get the access rules and access condition (**isoIec23465_getAccessRule/isoIec23465_getAccessCondition)**. The client application requests data to interpret the information and starts the required activities for fulfilment. Depending on the implementation, several ways for implementation are possible:

— The returned content is in the ISO/IEC 7816-4 format and must be interpreted by the client application/programmers.

— The proxy supports the interpretation of ISO/IEC 7816-4 access rule coding. The content is changed into a more readable format, which can be easily interpreted by the client application/programmers.

— Access rules of complex security device authentication protocols (e.g. PACE) may be facilitated. Respectively, steps of the protocols may be handled by the proxy without involvement of the client application.

— The client application knows the required access rules implicitly and applies the required API calls for fulfilment.

7.2.2.2 Security conditions partly handled by proxy

Client application requests access rules/conditions via the API calls **isoIec23465_getAccessRule/ isoIec23465_getAccessCondition**. The proxy may interpret the security condition and filters the security conditions which have to be processed by the proxy. Other security conditions are conveyed to the client application, possibly interpreted and facilitated into a more readable format.

All proxy-related security conditions are performed without involvement of the client application. A client application relies on the trustworthiness of the proxy functionality.

7.2.2.3 Security conditions fully handled by proxy

System architectures using a fully trustworthy proxy do not need an interpretation of the security conditions of the related security device. A request for security conditions is normally not performed and the fulfilment is done without the attention of the client application.

In this situation the proxy requests the security conditions from the selected security device and performs the activities to fulfil the required access rules automatically. All information and data to fulfil the security operations shall be conveyed to the proxy. In this scenario a proxy may use its own secure key store for the relevant key material.

7.2.3 Interpretation of ISO/IEC 7816-4 security conditions

This document proposes the interpretation of security attributes defined in ISO/IEC 7816-4:2020, 9.3. The security attributes may be expressed in compact or expanded format. They are always separated in the access mode, describing the commands to be protected, and the security condition for this access mode, possibly a concatenation of conditions. The complexity of security attributes derives from the logical combination of security conditions by AND, OR and NOT operations.

The concept of the ISO/IEC 23465 series aims to facilitate the usage of security devices. The proxy shall offer methods to translate complex security attributes into a format which is more understandable.

The following rules for usage of security conditions apply:

— Complex and nested security conditions shall be separated in sets of simplified security conditions.

— OR operations shall be separated into a set of security application to the same access mode.

— Concatenations of security conditions are allowed with AND operations.

— A NOT operation shall be applied only to a single security condition.

— Applications on a security device shall be designed to ease the translation process of the security attributes.

NOTE The possible complexity of security attributes in the security device often leads to the opinion that the usage of security devices is difficult. The translation of security requirements of a security device application into security attributes, expressed with ISO/IEC 7816-4 means, is a task of the security device provider. The client application only needs to know how to achieve the rights to apply actions on the security device.

7.2.4 Security attributes for complex multistep authentication

Authentication protocols with several steps are used to establish a secure channel between the security device and the external world. Several protocols are used, the most popular being the “password authenticated Diffie-Hellman key agreement protocol” PACE, defined in. [5]

NOTE The PACE protocol uses five commands between terminal and security device to finally establish the secure channel. As an input and starting point an MSE command initializes the security environment and defines at minimum the cryptographic mechanism reference and the password reference to be used. Four subsequent GENERAL AUTHENTICATION commands exchange the encrypted nonce, the security device’s domain parameter, ephemeral public keys and finally the authentication tokens. The latter ones are used to verify the correct derived session keys.

— With the impetus from the client application, a proxy may perform the PACE protocol’s five steps and will be authenticated by the security device. Finally, a secure channel is established. Any further access to the security device by the client application needs to be converted into the SM session. At the least, the required security attribute for further communication will be fulfilled.

— The client application can perform a chip and terminal authentication also described in.[5] In this case the SM session established with PACE between the proxy and security device will be changed into an end-to-end SM channel between external word and security device. The API call from the client application to the proxy (e.g. ENVELOPE) only transports the payload which has to be transferred to the security device.

— Other models may also be applicable. An established secure channel between proxy and the security device is introduced while another SM channel between the proxy and client application applies. The proxy shall decipher/encipher the incoming command/responses as a relay.

The applied solution depends on the proxy implementation and the demand from the external world.

NOTE Some implementations (e.g. Java card) do not use security attributes according to the ISO/IEC 7816 series but use an own state machine for checking the fulfilment of security conditions.

7.2.5 Security attribute “user verification”

User verification is an important security feature for many client applications and is usually supported by all security devices and its applications.

The ISO/IEC 23465-API supports the the fulfilment of this security attribute with class password and its related methods. The API offers all possible implementations of user verification. Finally, the supported API depends on the structure of passwords and counters of the implemented security device application.

7.2.6 Security attribute “authentication”

Authentication is an important security feature that allows the proof of authenticity of both communication partners. The internal, external or mutual authenthication are activities that fulfil the security attribute.

The ISO/IEC 23465-API provides the classes key and authenticator and its relevant methods to handle this security attribute on the security device.

7.2.7 Security attribute “secure messaging” (SM)

The security attributes for SM defines conditions and data settings for confidentiality and data authentication. As outlined in 7.2.4, the SM must be at least fulfilled at the edge of the security device. The origin of the secured message depends on the implementation and requirement of the proxy, the client application or the external world.

In each case the applicant of SM has access to keys/entities allowing the transformation into a SM format which can be understood by the security device and vice versa.

NOTE Several possibilities can be considered to achive this. Generally, an entity, acting as an HSM, can transform the payload in both directions.

7.2.8 Implicit fulfilment of security conditions

Many specifications define security conditions without the use of ISO/IEC 7816-4 definitions. The required preceded activites are implicitly known by the client application/external world. The security device blocks the access if the pre-conditions are not set properly.

The client application or the external world shall handle the fulfilment by applying the relevant API calls to set the corresponding security conditions.

NOTE The GlobalPlatform card specifications defines for the issuer domain access (e.g. on Java card operating systems) the access via secure channels. An authentication process establishes this implicitly.

7.3 Connectivity layer

In case of direct access, e.g. when the proxy and the security device are located in the same runtime environment, the connectivity layer handles the physical communication and conveys the command-response pairs directly. The communication and transmission specific properties can be internally known and are applied accordingly.

In other scenarios a security device specific driver may support the proxy and the connectivity layer. This additional piece of SW provided by the security device manufacturer can also be used in case of separated runtime environments. A standardised API for interopability reasons can be an advantage.

Connectivity via modules in separated runtime environments requires APIs. These APIs are actually provider or operating system specific. An exemption is the Open Mobile API which is already used in implementitations, e.g. supported by mobile phone operation systems.

In case of several security devices in a system, the connectivity layer may offer different mechanisms and channels, described above to address the security devices. The connectivity layer is enabled to address each security device individually.

7.4 Multi-client/multi-security device support

7.4.1 Task management

A proxy handles at least one client application and one security device. ISO/IEC 23465-1 describes architectures with several client applications and several security devices. This requires the ability of the proxy to administer multi instances of security device objects in numerous different security devices within a multitasking environment, depending on the needs and implementation of the system architecture.

The proxy may be called by different client applications in parallel asynchronously. The implementation of the proxy shall be designed to handle the API calls nearly simultaneously, with all the well-known restrictions and requirements. A possible task manager layer should be taken into account to set-up tasks per client application, managing the resources (e.g. time, memory, stack), and finally, releases the tasks and their occupied ressources when the client application stops.

7.4.2 Management of security devices

The different architectures outlined in ISO/IEC 23465-1 allow the use of

— several security devices in a session with at least one client application,

— different security device applications in one security device, and

— parallel processing of client application by a proxy.

The command handling of the ISO/IEC 7816 series generally deals with a single security device application within one session. This behaviour always reflects the small resources of ICC operating systems, which allows only single task or single thread handling.

A general design of the system allows any client application the access to several security devices. A client application is entitled to use more than one security device and manage this by using the object references. The client application API call with this dedicated reference builds the close binding between security device application and client application.

The same situation is allowed when several client applications may access several security devices. The proxy shall be able to handle objects for different client applications, linked to the same physical security device.

ISO/IEC 7816-4 with its channel mechanism offers a way to allow different applications to also have access to different security device applications on the same physical device. The access of the proxy to the security device implicitly initiates a set-up of a new channel. Further calls to that security client application by the client application are handled with the dedicated channel number.

NOTE The usage of this mechanism is limited since the number of channels is restricted.

An object reference in a security device application can be resolved only once in a session, as long as it is not released. Subsequent calls for referencing the same object shall be prohibited by the proxy using, e.g. a list of referenced objects assigned to each client application.

8 Life cycle model of the proxy

8.1 Start-up phase

Each SW module of the system outlined in ISO/IEC 23465-1 is initialized when the SW system runs through its start-up routines.

The proxy begins to set-up the connectivity to the registered security devices. The following interaction can be considered:

— Direct connection: the security devices will be initialised, e.g. with a Power-On_Reset and the initialisation of the transmission handling, e.g. ATR and PPS transmission.

— Access via APIs: The proxy calls the start-up function for the services and checks the availability of the security device, collects or use the information of its internal registry (see 5.3.3 and 5.3.4).

The basic activation of the interfaces/security devices enables further access to the security devices via the ISO/IEC 23465-API.

8.2 Initialization and administration

An administration interface to the proxy allows the first initialization or update of data and functionality. Depending on the architecture of the system, this can be part of the runtime environment of the system SW or library, which is separated from the system or is a specific part of the client application.

NOTE Initialization or administration is part of the maintenance of the system and is normally done by the manufacturers or OEMs.