The Recommended Security Policy for GSM/UMTS Compliant Devices

Addendum to the Mobile Information Device Profile version 2.0.1

Scope of This Document

This addendum is informative. However, all implementations of MIDP 2.0.1 on GSM/UMTS compliant devices are expected to comply with this addendum.

This addendum supersedes The Recommended Security Policy in MIDP 2.0 and JTWI (JSR 185)

MIDP 2.0.1 defines the framework for authenticating the source of a MIDlet suite and authorizing the MIDlet suite to perform protected functions by granting permissions it may have requested based on the security policy on the device. It also identifies functions that are deemed security vulnerable and defines permissions for those protected functions. Additionally, MIDP 2.0.1 specifies the common rules for APIs that can be used together with the MIDP but are specified outside the MIDP. MIDP 2.0.1 specification does not mandate a single trust model but rather allows the model to accord with the device trust policy.

The purpose of this addendum is to extend the base MIDlet suite security framework defined in MIDP 2.0.1 and to define the following areas:

- The required trust model for GSM/UMTS compliant devices
- The domain number and structure, as reflected in the device security policy
- The mechanism of reading root keys from sources external to the device
- Capabilities of MIDlets based on permissions defined by MIDP 2.0.1
- MIDlet behaviour in the roaming network
- MIDlet behaviour when SIM/USIM is changed
- The use of user permission types
- Guidelines on user prompts and notifications

How This Specification Is Organized

This specification is organized as follows:

Sections 2 and 3 establish the relationship between the device security policy, different protection domains, and requirements concerning certificate storage on smart cards. Section 4 specifies the function groups and identifies the permissions and the APIs that need to be protected using the MIDP 2.0.1 security framework. Sections 5 and 6 specify rules that MUST be followed when permissions are granted, and also requirements of user notifications. Section 7 specifies the MIDlet behavior during roaming and after changing the smart card. Finally Section 8 specifies Revocation Checking

References

1. Connected Limited Device Configuration (CLDC)

   http://jcp.org/jsr/detail/30.jsp

2. Mobile Information Device Profile (MIDP) 2.0.1

   http://jcp.org/jsr/detail/118.jsp

3. HTTP 1.1 Specification

   http://www.ietf.org/rfc/rfc2616.txt

4. WAP Wireless Identity Module Specification (WIM) WAP-260-WIM-20010712-a

   http://www.wapforum.org/what/technical.htm

5. WAP Smart Card Provisioning (SCPROV) WAP-186-ProvSC-20010710-a

   http://www.wapforum.org/what/technical.htm

6. PKCS#15 v.1.1

   http://www.rsasecurity.com/rsalabs/pkcs/pkcs-15/

7. USIM, 3GPP TS 31.102: "Characteristics of the USIM applications"

   http://www.3gpp.org

8. RFC 3280 Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile

   http://www.ietf.org/rfc/rfc3280.txt

9. RFC 2119 “Key words for use in RFCs to Indicate Requirement Levels”

   http://www.ietf.org/rfc/rfc2119.txt

10. Online Certificate Status Protocol Mobile Profile

    http://www.openmobilealliance.org

11. Online Certificate Status Protocol

http://www.ietf.org/rfc/rfc2560.txt?number=2560

Definitions

This document uses definitions based upon those specified in RFC 2119.

MUST	The associated definition is an absolute requirement of this specification.
MUST NOT	The definition is an absolute prohibition of this specification.
SHOULD	Indicates a recommended practice. There may exist valid reasons in particular circumstances to ignore this recommendation, but the full implications must be understood and carefully weighed before choosing a different course.
SHOULD NOT	Indicates a non-recommended practice. There may exist valid reasons in particular circumstances when the behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour described with this label.
MAY	Indicates that an item is truly optional.

1 General

GSM/UMTS compliant devices implementing this Recommended Security Policy MUST follow the security framework specified in the MIDP 2.0.1. Additionally, devices that support trusted MIDlets MUST follow the PKI-based authentication scheme as defined in MIDP 2.0.1 specification.

2 Protection Domains

A protection domain is a set of permissions that can be granted to a MIDlet. A MIDlet suite MUST belong to one and only one protection domain. The representation of a domain and its security policy is implementation-specific.

This document defines four protection domains: the Unidentified Third Party Protection Domain, the Identified Third Party Protection Domain, the Operator Protection Domain and the Manufacturer Protection Domain.

Applications bound to the Unidentified Third Party Protection Domain and the Identified Third Party Protection Domain run in an unprivileged environment wherein the user decides on the permissions given to the application (either by the choice of preferences or by answering to user prompts). In contrast to this, applications in the Operator Protection Domain and the Manufacturer Protection Domain run in a privileged environment. However, the privileges given to an application might still depend on the specific Protection Domain to which the application is bound.

3 Authorities on Protection Domains

This section specifies how a MIDlet suite's protection domain is determined, and introduces Authorities on Protection Domains. It also defines mechanisms for provisioning the device with certificates that are Authority Certificates on the different Protection Domains, as well as device behaviour when certificates change in either of these domains. Finally, recommendations on the operation of Authorities on these domains are given.

A MIDlet suite is assigned to a single Protection Domain based on its signature. As a result, one or more Authorities are responsible for applications in the Protection Domain, as specified in the following section.

Applications that are unsigned as per the MIDP 2.0.1 security framework MUST be installed under the Unidentified Third Party Protection Domain. For any X.509 Certificate under which an application is signed, the device MUST decide whether the certificate is:

- an Authority Certificate on the Identified Third Party Protection Domain. Applications signed "under the authority[1]" of such a certificate, as per the MIDP 2.0.1 security framework, MUST be run under the Identified Third Party Protection Domain.
- an Authority Certificate on the Operator Protection Domain. Applications signed "under the authority[1]" of such a certificate, as per the MIDP 2.0.1 security framework, MUST be run under the Operator Protection Domain.
- an Authority Certificate on the Manufacturer Protection Domain. Applications signed "under the authority[1]" of such a certificate, as per the MIDP 2.0.1 security framework, MUST be run under the Manufacturer Protection Domain.
- another Certificate
  Applications signed with a certificate not bound to any protection domains, MUST NOT be installed on the device.

3.1 Manufacturer and Operator Protection Domains

Recommendation: Permissions in the Manufacturer and Operator Protection Domains are marked as *Allowed* so that downloaded MIDlets suites perform consistently with MIDlet suites pre-installed by the manufacturer, in terms of security and prompts to the user, whenever actions that require user acknowledgement are requested by a MIDlet suite. The organization holding the private keys corresponding to an Authority Certificate on a Manufacturer or Operator Protection Domain SHOULD ensure that MIDlets that are signed under their authority seek permission from the user when accessing security vulnerable APIs and functions. Permissions defined by MIDP 2.0.1 and other APIs, as well as section 5 below, give the guidelines of which functions need protection.

3.1.1 Manufacturer Protection Domain

The Manufacturer Protection Domain Root Certificate is used to verify manufacturer MIDlet suites. The Manufacturer Protection Domain Root Certificate MUST be mapped onto the security policy for the Manufacturer Protection Domain on the

[1] Meaning: signed by a certificate issued under the certification hierarchy of this certificate

device. A device MUST support the security policy for the Manufacturer Protection Domain.

If the Manufacturer Protection Domain Root Certificate is NOT available on the device, the Manufacturer Protection Domain MUST be disabled.

The Manufacturer Protection Domain Root Certificate can be added, deleted or modified only by the manufacturer, who may use an update mechanism whose details are outside the scope of this specification. Any new or updated Manufacturer Protection Domain Root Certificate MUST be associated with the security policy for the Manufacturer Protection Domain on the device. MIDlet suites verified by a previous Manufacturer Protection Domain Root Certificate MUST be disabled.

If the trustedUsage field is present and contains the OID for key usage "*iso(1)org(3)dod(6)internet(1)private(4)enterprises(1)sun(42)*

products(2)javaXMLsoftware(110)midp(2)spec(2)gsm-policy(2)manufacturer(2)", then the certificate is to be considered an Authority Certificate on the Manufacturer Protection Domain.

.At installation of a MIDlet suite signed under the Authority of the Manufacturer Protection Domain, a compliant implementation MUST present the user with the *Organization* and *Country* fields within the Subject field of the signing certificate of a MIDlet suite if these fields are present. If these fields are absent, the implementation SHOULD present the user with other appropriate information from the Subject field. An implementation MAY also present the user with additional information in the Subject field other than *Organization* and *Country* in all cases. This user notification MUST take place before application installation.

The Manufacturer Protection Domain imposes no restriction on the capabilities specified in the MIDP 2.0.1 and other JSRs.

3.1.2 Operator Protection Domain

An Operator Protection Domain Root Certificate is used to verify operator MIDlet suites. Operator Protection Domain Root Certificates MUST be mapped onto the security policy for the Operator Protection Domain on the device.

A device MUST support the security policy for the Operator Protection Domain.

A device MUST support the mechanism [SCPROV] to read Authority Certificates for the Operator Protection Domain stored in the smart card (for example, SIM, USIM or WIM).

Additionally, the device MUST support Authority Certificates for the Operator Protection Domain stored on the device; for example, a device resident Operator Protection Domain Root Certificate.

There is at a maximum one Operator Protection Domain Root Certificates available at either of two specified locations: the smart card or on the device. For example there may be up to one enabled Operator Protection Domain Root Certificate per smart card and up to one enabled device resident Operator Protection Domain Root Certificate.

However, if Operator Protection Domain Root Certificates are found on the device and on the smart card (e.g. SIM, USIM or WIM) at the same time, the root certificate found on the smart card will have a higher priority than a device resident root certificate. Therefore, it is recommended for an operator to put the same root certificate on its smart cards as root certificates which were previously put on its devices (if that's the case for a progressive introduction of root certificates on the SIM card for example). In this way the operator ensures that existing MIDlets will continue to be able to execute when introducing the smart card which contains the root certificate.

If an Operator Protection Domain Root Certificate is NOT available at the specified location in either the smart card or on the device the Operator Protection Domain MUST be disabled.

Potential consequences of a change of the smart card are described in section 7.

The Operator Protection Domain Root Certificate MUST only be deleted or modified by the operator, who may use an update mechanism whose details are outside the scope of this specification.

The user MUST NOT be able to delete Authority Certificates for the Operator Protection Domain.

Any new or updated Operator Protection Domain Root Certificate MUST be associated with the security policy for the Operator Protection Domain on the device.

MIDlet suites verified by a non valid (for example, disabled) Operator Protection Domain Root Certificate MUST be disabled.

Before invoking a MIDlet the device MUST check whether or not the Protection Domain Root Certificate is still valid. If the Protection Domain Root Certificate that verified the MIDlet is no longer valid then the MIDlet MUST NOT be invoked. The device SHOULD then present a suitable message to the user.

Only if the Operator Protection Domain Root Certificate is valid during the installation process the device MUST continue the installation process.

If this condition is not fulfilled the device MUST NOT install the MIDlet and MUST act as described in JSR 118, “Over the Air User Initiated Provisioning Specification” by sending the appropriate error code.

Authority Certificates are read from the Certificate Directory File (CDF) for trusted certificates [WIM]. Authority Certificates found in the trustedCertificates file on the WIM are considered to be Authority Certificates on the Operator Protection Domain or onto the Identified Third Party Protection Domain (see section 3.2.0.1), depending on the trustedUsage field in the CommonCertificateAttributes associated with the certificate [PKCS#15]:

If the trustedUsage field is present and contains the OID for key usage

> “*iso(1)org(3)dod(6)internet(1)private(4)enterprises(1)sun(42) products(2)javaXMLsoftware(110)midp(2)spec(2)gsm-policy(2)operator(1)*”,

then the certificate is to be considered an Authority Certificate on the Operator Protection Domain.

If the trustedUsage field is absent, or does not contain this OID but

> *“iso(1)org(3)dod(6)internet(1)private(4)enterprises(1)sun(42) products(2)javaXMLsoftware(110)midp(2)spec(2)gsm-policy(2)identifiedParty(32)”*

the certificate is to be considered an Authority Certificate on the Identified Third Party Protection Domain (see section 3.2.1).

Figure 1 summarizes assigning Authority Certificates obtained through [SCPROV] to Protection Domains.

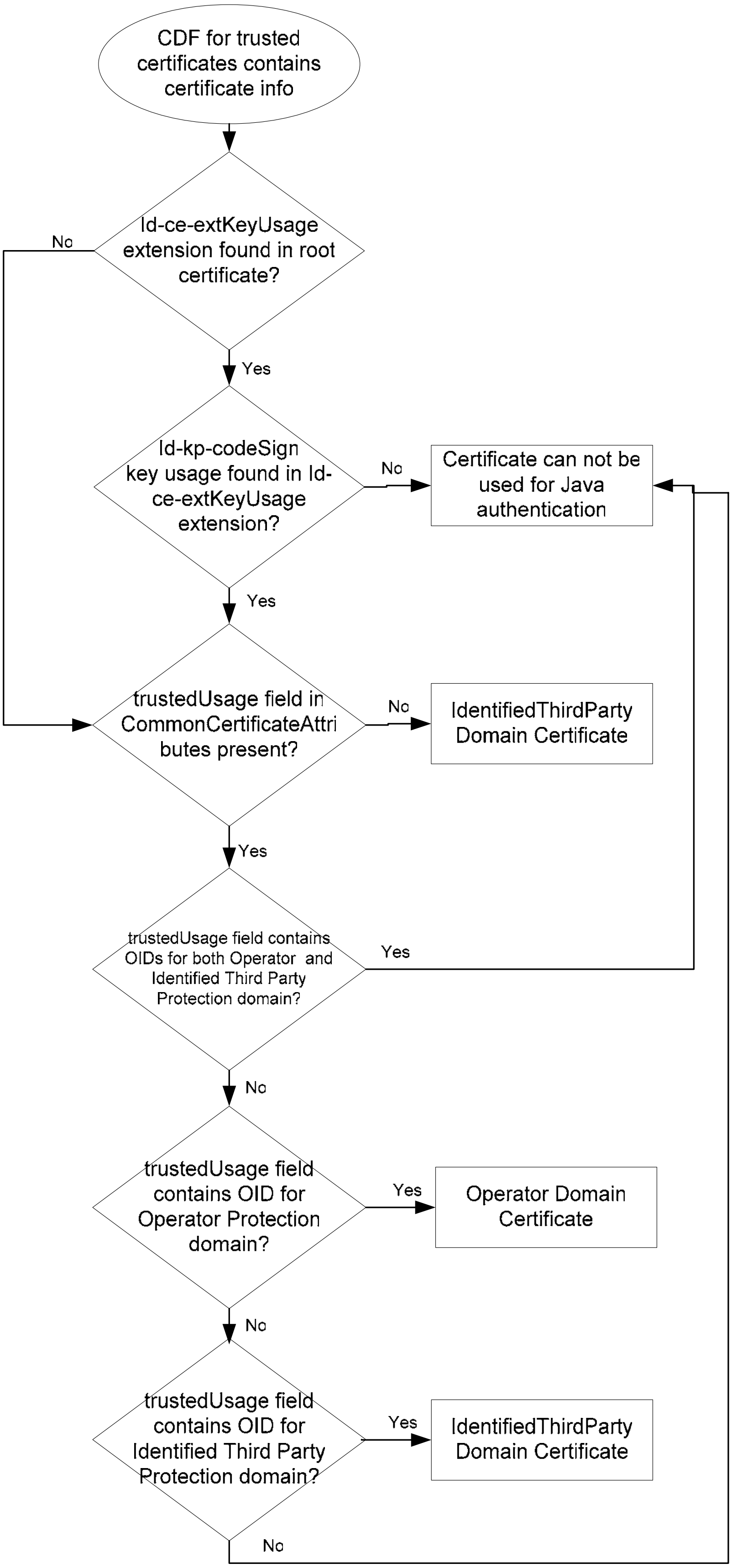

CDF for trusted certificates contains certificate info
Id-ce-extKeyUsage extension found in root certificate?
No
Yes
Id-kp-codeSign key usage found in Id-ce-extKeyUsage extension?
No
Certificate can not be used for Java authentication
Yes
trustedUsage field in CommonCertificateAttributes present?
No
IdentifiedThirdParty Domain Certificate
Yes
trustedUsage field contains OIDs for both Operator and Identified Third Party Protection domain?
Yes
No
trustedUsage field contains OID for Operator Protection domain?
Yes
Operator Domain Certificate
No
trustedUsage field contains OID for Identified Third Party Protection domain?
Yes
IdentifiedThirdParty Domain Certificate
No

Figure 1: Assigning Operator and Identified Third Party Authority Certificates to Protection Domains

At installation of a MIDlet suite signed under the Authority of the Operator Protection Domain, a compliant implementation MUST present the user with the Organization and *Country* fields within the *Subject* field of the signing certificate of a MIDlet suite if these fields are present. If these fields are absent, the implementation SHOULD present the user with other appropriate information from the *Subject* field. An implementation MAY also present the user with additional information in the *Subject* field other than Organization and *Country* in all cases. This user notification MUST take place before application installation.

MIDlet suites installed in the Operator Protection Domain MUST store, along with the application itself, a hash of the Authority Certificate under which the certificate used to sign the application was issued. The hash algorithm to be used is the following: starting with the Authority Certificate, compute the 20-byte SHA-1 hash of the value of the BIT STRING subjectPublicKey (excluding the tag, length, and number of unused bits) of that certificate. This method is commonly used to compute key identifiers, especially to accelerate trust chain building [RFC3280, §4.2.1.2]. The implementation MUST NOT assume for optimization purposes that X.509 key identifiers or PKCS#15 labels have the correct value and MUST compute the hash themselves. In the event of a smart card change this hash MUST be used by the device to decide when a given MIDlet suite should be disabled, as specified in Section 7.

3.2 Third Party Protection Domains

Recommendation: All permissions in the Identified Third Party and the Unidentified Third Party Protection Domains are subject to user permission according to the security policies. Differences between the two Third Party Protection Domains are:

- The identification of the Third Party which created the application
- Different default and other user settings

3.2.1 Identified Third Party Protection Domain

MIDlet suites in this domain MUST be granted permissions as per the policy set in section 5 below. This policy mixes user prompting at runtime as well as user settings in order to ensure the safety of all applications in this protection domain.

Applications that are authenticated using Authority Certificates on the Identified Third Party Protection Domain are mapped to the Identified Third Party Protection Domain. There is no explicit limitation on the number of Identified Third Party Protection Domain Root Certificates available either on the device or at the specified location in the smart card.

All Identified Third Party Protection Domain Root Certificates MUST be mapped onto the security policy for the Identified Third Party Protection Domain on the device. A device MUST implement the Identified Third Party Protection Domain. However, if there are no Identified Third Party Authority Certificates available either on the device or at the specified location in the smart card the Identified Third Party Protection Domain MUST be disabled.

A device SHOULD support the mechanism [SCPROV] to read Authority Certificates on the Identified Third Party Domain stored in the (U)ICC. Additionally, Authority Certificates for the Identified Third Party Domain MAY be stored on the device. A device MUST support at least one Authority Certificate storage mechanism. If both mechanisms are supported, the device MUST use Authority Certificates from both mechanisms.

Authority Certificates are read from the Certificate Directory File (CDF) for trusted certificates [WIM]. Authority Certificates found in the trustedCertificates file on the WIM are considered to be Authority Certificates on the Operator Protection Domain (see section 3.1.2) or onto the Identified Third Party Protection Domain, depending on the trustedUsage field in the CommonCertificateAttributes associated with the certificate [PKCS#15]:

If the trustedUsage field is present and contains the OID for key usage "*iso(1)org(3)dod(6)internet(1)private(4)enterprises(1)sun(42)*

products(2)javaXMLsoftware(110)midp(2)spec(2)gsm-policy(2)identifiedParty (3)", then the certificate is to be considered an Authority Certificate on the Identified Third Party Protection Domain. Please see Figure 1 for details.

Authority Certificates may be placed in the trustedCertificates Certificate Directory File (CDF) of a WIM, SIM, or USIM or on the device. If Authority Certificates are stored directly on a SIM or USIM, that is, not under the WIM application, then they shall be stored in the EF trustedCertificates CDF located under DF(PKCS#15), as defined by [SCPROV]. Authority Certificates can be obtained only from the trusted CDF (the card holder can not update this directory) and not from any other directory of the smart card.

Any Authority Certificates obtained after device manufacture MUST NOT be used for authentication of MIDlet suites. This does NOT prevent obtaining Identified Third Party Authority Certificates from the specified location in the SIM, USIM or WIM.

At MIDlet suite installation, an implementation MUST present the user with the *Organization* and *Country* fields within the *Subject* field of the signing certificate of a MIDlet suite if these fields are present. If these fields are absent, the implementation SHOULD present the user with other appropriate information from the *Subject* field. An implementation MAY also present the user with additional information in the *Subject* field other than *Organization* and *Country* in all cases. This user notification MUST take place before MIDlet suite installation. When the user is prompted to grant permissions for a MIDlet suite to function groups, the prompt MUST identify the identified source with the appropriate fields within the *Subject* field of the signing certificate as stated above.

The user MUST be able to delete or disable Identified Third Party Authority Certificates. The deletion of these certificates is only possible for Authority Certificates stored on the device and not for those obtained using [SCPROV]. If an Identified Third Party Authority Certificate is to be deleted, the implementation MUST adequately warn the user of the consequence of the deletion. A disabled Identified Third Party Authority Certificate MUST NOT be used to verify downloaded MIDlet suites. The user MUST be able to re-enable a disabled Identified Third Party Authority Certificate that was previously disabled by the user. Furthermore, if an Identified Third Party Authority Certificate is deleted or disabled (for example, revoked, deleted, or disabled by the user) the Identified Third Party Protection Domain MUST no longer be associated with this Authority Certificate. If the user chooses to delete or disable an Identified Third Party Authority Certificate, the implementation MAY provide an option to delete the MIDlet suites authenticated to it.

The security policy for the Identified Third Party Protection Domain MUST NOT grant any permission on the device as *Allowed.* All permissions granted by the Identified Third Party Protection Domain MUST be *User* permissions, that is, user interaction is required for permission to be granted. Table 1 specifies the function groups and the available user permission types for MIDlet suites in the Identified Third Party Protection Domain. Tables 2 through 12 specify the mapping of permissions and APIs onto different function groups.

3.2.2 Unidentified Third Party Protection Domain

MIDlet suites in this domain MUST be granted permissions as per the policy set in section 5 below. This policy mixes user prompting at runtime as well as user settings in order to ensure the safety of all applications in this protection domain. It should be noted that versions of MIDP prior to 2.0.1 referred to this Domain as “Untrusted.”

MIDlets suites that are unsigned will belong to the Unidentified Third Party Protection Domain. A device MUST support the security policy for the Unidentified Third Party Protection Domain. The implementation MUST inform the user whenever a new MIDlet suite is installed in the Unidentified Third Party Protection Domain. The notification MUST indicate that the source of the application cannot be verified. The user must be able to make an informed decision based on the available information before granting permissions to an application.

When the user is prompted to grant permissions to an application, the prompt MUST visually indicate that the application does not come from a trusted source. Implementations are recommended to indicate this by showing a different icon on the screen shown for the security prompt for MIDlet suites that are authenticated by Authority Certificates on the Manufacturer, Operator or Identified Third Party Protection Domain then for MIDlet suites that are not signed or can not be authenticated by all Authority Certificates on the Manufacturer, Operator or Identified Third Party Protection Domain. The recommended icons are shown in figure 2 below.

Figure 2: Recommended icons for the security prompt for authenticated (left) and unauthenticated MIDlets (right)

4 Permissions for Downloaded MIDlet Suites

4.1 Mapping MIDP 2.0.1 Permissions onto Function Groups in Protected Domains

A device with a small display may not be able to present all permissions on an API level to the user in a single configuration settings menu in a user friendly manner. Therefore the device is not required to present all individual permissions for user confirmation. Rather, a certain higher-level action triggered by the protected function should be brought to the user for acceptance. The high level functions presented to the user essentially capture and reflect the actions and consequences of the underlying individual permissions. These so-called function groups are as follows:

Network/cost-related groups:

Phone Call – the group represents permissions to any function that results in a voice call.

Call Control – the group represents permissions to any function that results call setup or teardown of a restricted network connection.

Net Access – the group represents permissions to any function that results in an active network data connection (for example GSM, GPRS, UMTS, etc.); such functions must be mapped to this group.

Low Level Net Access – the group represents permissions to any function that results in an active low level network data connection (for example Sockets, etc.); such functions must be mapped to this group.

Messaging – the group represents permissions to any function that allows sending or receiving messages (for example, SMS, MMS, etc.)

Restricted Messaging – the group represents permissions to any function that allows sending or receiving messages to a restricted messaging service (for example, Cell Broadcast, etc.)

Application Auto Invocation – the group represents permissions to any function that allows a MIDlet suite to be invoked automatically (for example, push, timed MIDlets, etc.)

Local Connectivity – the group represents permissions to any function that activates a local port for further connection (for example, COMM port, IrDA, Bluetooth, etc.)

Authentication - the group represents permissions to any function that gives a MIDlet suite access to authentication functionality.

User-privacy-related groups:

Multimedia recording – the group represents permissions to any function that gives a MIDlet suite the ability to do any kind of multimedia recording (for example capture still images, or to record video or audio clips).

Read User Data Access – the group represents permissions to any function that gives a MIDlet suite the ability to read a user's phone book, or any other data in a file or directory.

Write User Data Access – the group represents permissions to any function that gives a MIDlet suite the ability to add or modify a user's phone book, or any other data in a file or directory.

Smart Card Communication – the group represents permissions to any function that gives a MIDlet suite the ability to communicate with the smart card.

Location – the group represents permissions to any function that gives a MIDlet suite access to Location information.

Landmark - the group represents permissions to any function that gives a MIDlet suite access to Landmark information.

Whenever new features are added to MIDP they should be assigned to the appropriate function group. In addition, APIs that are specified elsewhere (that is, in other JSRs) but rely on the MIDP security framework should also be assigned to an appropriate function group. If none of the function groups defined in this section is able to capture the new feature and reflect it to the user adequately a new function group MUST be defined in this document by requesting an update to this document from MSA or MIDP as appropriate.

If a new function group is to be added, the following should be taken into consideration: the group to be added MUST not introduce any redundancy to the existing groups, the new group MUST be capable of protecting a wide range of similar features. The latter requirement is to prevent introducing narrowly scoped groups. The new function group SHOULD be sufficiently future-proof to contain new features added by future APIs and should not only concern the features being initially included in it.

It is the function groups and not the individual permissions that should be presented when the user is prompted. Furthermore, it is the function groups that should be presented to the user in the settings of a given MIDlet suite.

Table 1 presents the policy that must be enforced using the security framework as defined in MIDP 2.0.1. The table specifies the available permission settings for each function group defined. Settings that are effective at the time the MIDlet suite is invoked for the first time, and remain effective until the user changes them in the MIDlet suite's configuration menu, are called "default settings." Settings available to the user in the configuration menu, to which the user can change from a default setting, are called "other settings." Together, default and other settings form a pool of available configuration settings for the MIDlet suite. Default and other settings are presented for each function group and both Third Party Protection Domains. The naming of the function groups is implementation specific but MUST follow the guidelines of the function group names defined in this document as well as the definitions of these groups.

Table 2 presents individual permissions defined in the MIDP 2.0.1, and map to the function groups specified in this section. An individual permission MUST occur in only one function group.

It is recommended that MIDlet suites in the Manufacturer and Operator Protection Domains adhere to the permission guidelines provided in the tables, and present appropriate prompts to the user for the functions identified as security protected.

Table 1: Function groups and user settings for Third Party Protection Domains

Function group	**Identified Third Party Protection Domain**		**Third Party Protection Domain**	
Phone Call	default setting	Oneshot	default setting	Oneshot
	other settings	Blanket, Session, No	other settings	No
Net Access	default setting	Session	default setting	Oneshot
	other settings	Blanket, Oneshot, No	other settings	Session, No
Low Level Net Access	default setting	Session	default setting	Oneshot
	other settings	Blanket, Oneshot, No	other settings	Session, No
Messaging	default setting	Oneshot	default setting	Oneshot
	other settings	Blanket, Session, No	other settings	No
Restricted Messaging	default setting	Oneshot	default setting	Oneshot
	other settings	Blanket, Oneshot, No	other settings	No
Application Auto Invocation	default setting	Oneshot	default setting	Oneshot
	other settings	Blanket, Session, No	other settings	Session, No
Local Connectivity	default setting	Session	default setting	Oneshot
	other settings	Blanket, Oneshot, No	other settings	Blanket, Session, No
Multimedia recording	default setting	Session	default setting	Oneshot
	other settings	Blanket, Oneshot, No	other settings	Session, No
Read User Data Access	default setting	Oneshot	default setting	Oneshot
	other settings	Blanket, Session, No	other settings	No
Write User Data Access	default setting	Oneshot	default setting	Oneshot
	other settings	Blanket, Session, No	other settings	No
Location	default setting	Session	default setting	Oneshot
	other settings	Blanket, Oneshot, No	other settings	Session, No

Landmark Store	default setting	Session	default setting	Oneshot
	other settings	Blanket, Oneshot, No	other settings	Session, No
Smart Card Communication	default setting	No	default setting	No
	other settings	No	other settings	No
Authentication	default setting	Oneshot	default setting	No
	other settings	No	other settings	No
Call Control	default setting	Oneshot	default setting	Oneshot
	other settings	Blanket, Session, No	other settings	No

The device MAY enhance and simplify the user experience by applying a single set of configuration settings (default or other), not just to a single MIDlet suite, but to all MIDlet suites for a given signer. This option MUST NOT compromise the function groups and available settings defined in Table 1. If such an option exists, the user will be prompted to save the settings and reuse them in the future for MIDlet suites from the same signer. Such a feature MAY also inform the user that a given source has already been accepted and has an alias to the saved configuration settings. For each application, the implementation MAY read requested permissions from the MIDlet-Permissions and MIDlet-PermissionsOpt attributes, notify the user which capability the application requires, and prompt the user to accept or reject installation of the application.

If the security policy for the default and other settings of the domain for the Net Access and Low Level Net Access function groups are identical then the low level network access group may be merged into the Net Access function group. When these two function groups are merged, the implementation MUST behave as if all of the permissions are included in the Net Access function group.

If the security policy for the default and other settings of the domain for the Messaging and Restricted Messaging function groups are identical then the Restricted Messaging group may be merged into the Messaging function group. When these two function groups are merged, the implementation MUST behave as if all of the permissions are included in the Messaging function group.

If the security policy for the default and other settings of the domain for the Phone Call and Call Control function groups are identical then the Call Control function group may be merged into the Phone Call function group. When these two function groups are merged, the implementation MUST behave as if all of the permissions are included in the Phone Call function group.

Blanket permission given for some combinations of Function groups can lead to higher risks for the user. For MIDlet suites in the Identified Third Party Protection Domain the user MUST be notified of the higher risk involved and also acknowledge that this risk is accepted to allow such combinations to be set. The combination of Blanket permission in Function groups where this applies is:

- Any of Net Access, Messaging or Local Connectivity set to Blanket in combination with any of Multimedia recording or Read User Data Access set to Blanket

This restriction does not apply to the Unidentified Third Party Protection Domain, since these combinations are forbidden in this domain according to table 1.

Additionally, the Blanket setting for Application Auto Invocation and the Blanket setting for Net Access are mutually exclusive. This constraint is to prevent a MIDlet suite from auto-invoking itself, then accessing a chargeable network without the user being aware. If the user attempts to set either the Application Auto Invocation or the Network Function group to "Blanket" when the other Function group is already in "Blanket" mode, the user MUST be prompted as to which of the two Function groups shall be granted "Blanket" and which Function group shall be granted "Session".

For each Phone Call and Messaging action, the implementation MUST present the user with the destination phone number or the destination name before the user approves the action. For the Messaging group, if the implementation maps a single API call to more than one message (that is, the implementation supports disassembly/reassembly), the implementation MUST present the user with the number of messages that will actually be sent out. This requirement is to ensure that the user always understands the network costs associated with running the program, whatever API calls are involved.

Table 2: Assigning permissions specified in MIDP 2.0.1 to function groups

MIDP 2.0.1 - JSR 118		
Permission	**Protocol**	**Function group**
javax.microedition.io.Connector.http	http	Net Access
javax.microedition.io.Connector.https	https	Net Access
javax.microedition.io.Connector.datagram	datagram	Low Level Net Access
javax.microedition.io.Connector.datagramreceiver	datagram server (without host)	Low Level Net Access
javax.microedition.io.Connector.socket	socket	Low Level Net Access
javax.microedition.io.Connector.serversocket	server socket (without host)	Low Level Net Access
javax.microedition.io.Connector.ssl	ssl	Low Level Net Access
javax.microedition.io.Connector.comm	comm	Local Connectivity
javax.microedition.io.PushRegistry	All	Application Auto Invocation

5 Permissions Granted to a MIDlet Suite by the Authorization Mechanism

Permissions granted to a MIDlet suite are effectively the intersection of the domain permissions, permissions requested in MIDlet-Permissions and the MIDlet-Permissions-Opt attributes. See the section “Granting permissions to trusted MIDlet suites”: The permissions granted to the MIDlet suite are the intersection of the requested permissions with the union of the allowed and user granted permissions. For security reasons the permission attributes MUST be included in both the JAR manifest and in the JAD so that the device will have information about the requested permissions before the JAR is downloaded. These attribute values MUST be identical or otherwise the MIDlet suite MUST NOT be installed or invoked. All JAD-only implementations of these attributes (both in accepting policy for devices and in MIDlet suites) SHOULD be deprecated. The way in which a MIDlet suite's granted permissions are presented to the user is implementation-specific, but the following rules MUST apply:

- The user MUST be able to change the default permission setting to any setting available for a given MIDlet suite permission, provided they are in accordance with the implementation notes in section 5.2 and with default and available sets of user permission types provided as guides in the tables in Section 5. This latitude will allow the user to upgrade or downgrade the default permissions as required.
- If MIDlet permissions are grouped according to capabilities they represent, permissions granted to a MIDlet suite will be rendered into the function groups to be presented to the user. If function grouping is used, the default permission applies to the whole group of permissions under the group. So does the available set of types of user permissions. If the default permission is changed, the change is effective for the entire group at once rather than to the individual permissions under this group.
- A function group cannot be a union of permissions with different default settings and other settings. Therefore the tables in Section 5 follow the convention of having the same default and available settings for all permissions in a single function group. This rule MUST be taken into account when designing new permissions and policies.

A device MUST maintain security related data for each installed MIDlet suite, in addition to generic MIDlet suite information such as MIDlet suite name and version number. The security related data MUST NOT be accessible by any Third Party Protection Domain MIDlet. The data MUST include at least the following:

- The signer of the MIDlet suite, for example, the *Subject* field in the signing certificate, if the MIDlet suite was signed. At least MIDlet-Vendor MUST be stored along with the installed MIDlet suite.
- Data related to the Protection Domain Root Certificate a signed MIDlet was authenticated to; at minimum the *Subject* field of the Protection Domain Root Certificate.

- Data related to a certificate that signed the MIDlet suite; at minimum the certificate's *Subject*, *Issuer*, and *Serial Number* fields. (As an alternative, a device may store the entire certificate chain that came with the MIDlet descriptor file.)
- A list of permissions granted to the MIDlet suite.

A device MUST be able to present information related to the application signer in a user-friendly manner.

Requirements on Restricted APIs

When the user grants permission to a function group, this action effectively grants access to all individual permissions under this function group.

An implementation MUST guarantee that a SecurityException is thrown when the caller does not have the appropriate security permissions.

If a MIDlet uses the capabilities defined in MIDP and other APIs, the following rules MUST apply:

- All the external API functions that need to be protected by the MIDP 2.0 security framework MUST have permissions defined in those JSRs, and follow the naming rules identified in the MIDP 2.0 Specification, titled “Security for MIDP Applications.”
- The functions that are not deemed security-protected by specification can be accessed explicitly by untrusted MIDlet suites, as per general MIDP security rules.
- If an external API does not define permissions for security-protected functions because the API specification is release earlier than MIDP 2.0, any functions that relate to network access MUST still have the user prompt implemented by the device.
- A device cannot access the network without appropriate user notification.
- All licensee open classes MUST adhere to the permission framework as defined in this document.

6 User Prompts and Notifications

The following rules MUST be followed in order to ensure informed user consent to MIDlet actions:

- Any chargeable event generated by a MIDlet in one of the two Third Party Protection Domains MUST be preceded by user notification in accordance with user permission settings, for example, showing the phone number or corresponding name the MIDlet is dialing, the URL being connected to, or the recipient of an SMS.
- Any chargeable event in progress (for example, peer-to-peer connection the user is charged for) MUST be indicated to the user.
- A MIDlet MUST get user approval to connect to the network, in accordance with user permission settings of the policy.
- Any MIDlet permissions SHOULD be presented to the user in an intuitive, user-friendly manner.

- A MIDlet MUST not be able to override security prompts and notifications to the user generated by the system or virtual machine.
- A MIDlet MUST not be able to simulate security warnings to mislead the user.
- A MIDlet MUST not be able to simulate key-press events to mislead the user.

7 MIDlet Download and Execution While Roaming and After Changing the Smart Card

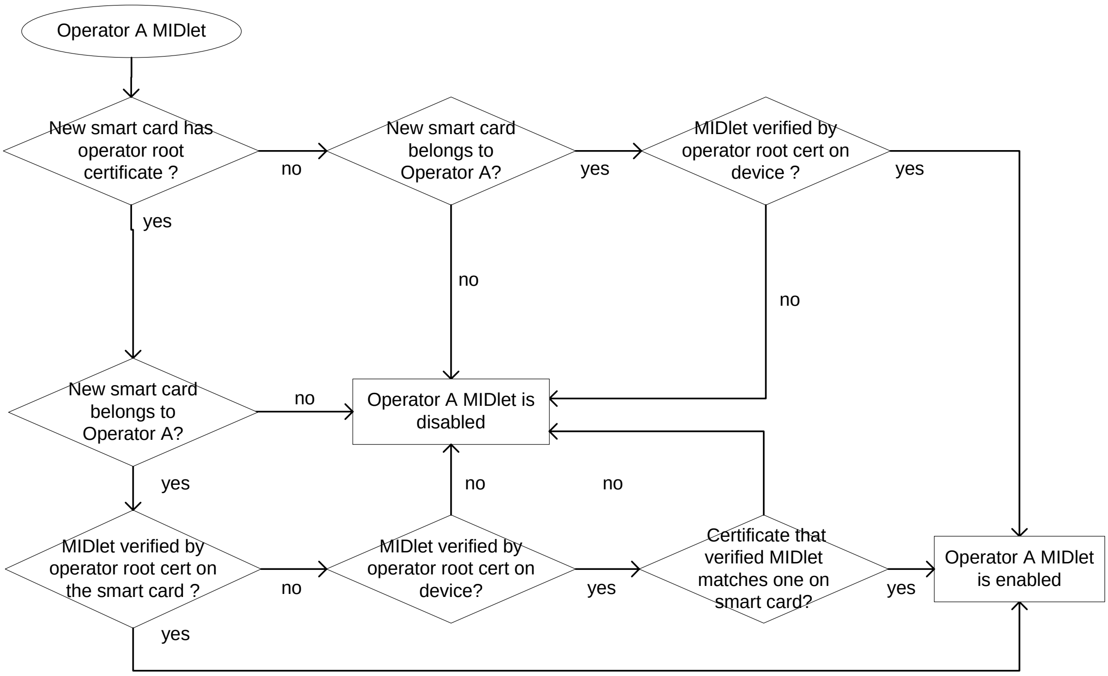
All previously authorized and installed MIDlet suites MUST act in accordance with the domain security policy when the device is roaming, or when the device smart card is changed.

Newly downloaded MIDlet suites are authenticated to a Protection Domain Root Certificate currently available either on the device or at the specified location on the smart card (for example, SIM, USIM or WIM) and are authorized in accordance with the security policy.

If device roaming or a smart card change causes a failure to access network resources that the MIDlet was previously authorized to access, then the implementation MUST NOT throw a SecurityException. This failure is not related to MIDlet suite authorization, so the implementation MUST throw an IOException instead.

The permissions assigned to MIDlet suites installed in the Manufacturer and Unidentified Third Party Protection Domain MUST NOT be affected by changes of the smart card but MIDlet suites installed in the Operator or Identified Third Party Protection Domain MUST NOT execute if, after a smart card change, Authority certificate that was used to authenticate the MIDlet suite to the Identified Third Party or Operator Protection Domain is no longer available and until the corresponding Authority certificate becomes available again.

Furthermore, whether a MIDlet suite in the Operator Protection Domain can be executed depends on the verification steps highlighted in the diagram below.



Operator A MIDlet
New smart card has operator root certificate ?
no
New smart card belongs to Operator A?
yes
MIDlet verified by operator root cert on device ?
yes
yes
no
no
New smart card belongs to Operator A?
no
Operator A MIDlet is disabled
yes
no
no
MIDlet verified by operator root cert on the smart card ?
no
MIDlet verified by operator root cert on device?
yes
Certificate that verified MIDlet matches one on smart card?
yes
Operator A MIDlet is enabled
yes

Figure 3: rules to apply before executing an operator domain related MIDlet

Figure 3 serves the purpose of showing possible scenarios and their outcomes with respect to execution of already installed MIDlets in the Operator Protection Domain. In order to conclude whether a MIDlet in the Operator Protection Domain may execute or will have to be disabled the following steps MUST be performed:

1. Compare "root key hash" values, computed as the 20-byte SHA-1 hash of the value of the BIT STRING subjectPublicKey (excluding the tag, length, and number of unused bits) of the Operator Protection Domain Root Certificate that verified a given MIDlet. Compute the 20-byte SHA-1 hash of the value of the BIT STRING subjectPublicKey (excluding the tag, length, and number of unused bits) of each Authority Certificate stored in the new smart card with the Operator-domain key-usage field (Operator-domain root key hashes). A MIDlet in the Operator Protection Domain is disabled if its authenticating root key hash does not correspond to one of the new Operator Protection Domain root key hashes generated after the smart card was changed.
2. Compare the MCC+MNC portion of the IMSI that is stored alongside Operator Protection Domain Root Certificates on the device with the ones read from the newly inserted smart card. A MIDlet in the Operator Protection Domain is disabled if the values do not match.

Please note that depending on the possible scenario one or both of the above steps MUST be performed immediately after the smart card change or before execution of the MIDlet after a smart card change.

If a MIDlet suite (either related to Operator or Identified Third Party Protection Domain) cannot be executed due to the smart card change, the device MUST NOT delete the MIDlet suite. The device MAY inform the user in advance via an appropriate mechanism whether a MIDlet suite could execute or not, for example using a "disabled" look and feel in the display. However, the user MUST be able to delete these disabled MIDlets suites. If the device can not inform the user in advance of the possibility to execute a MIDlet suite, it MUST inform the user when he tries to execute the MIDlet suite that the application can not be executed without the authorizing Protection Domain Root Certificate. The device SHOULD also give the user the option to get information on the Protection Domain Root Certificate that was used to authenticate the application to the Operator or Identified Third Party Protection Domain. This information SHOULD include the *Subject* field of the Authority Certificate.

An implementation MAY additionally perform steps 1 and/or 2 at any time and accordingly disable MIDlets suites in the Operator Protection

8 Revocation Checking

For all signed applications, the implementation MUST check revocation status using the Online Certificate Status Protocol (OCSP). A compliant implementation SHOULD support OCSP Mobile Profile as specified in the OMA [10]. Alternatively, an implementation MAY implement OCSP according to RFC 2560 [11].