



## Zentrum für sichere Informationstechnologie – Austria Secure Information Technology Center - Austria

A-1040 Wien, Weyringergasse 35  
Tel.: ++43 1 – 503 19 63 – 0  
Fax: ++43 1 – 503 19 63 – 66

A-8010 Graz, Inffeldgasse 16a  
Tel.: ++43 316 – 873 5514  
Fax: ++43 316 – 873 5520

# SECURITY LAYER *NEU* EXTENDED (SELANEXT)

## KURZBESCHREIBUNG

### VERSION 1.0.3

Thomas Rössler, A-SIT • eMail: [Thomas.Roessler@a-sit.at](mailto:Thomas.Roessler@a-sit.at)

## Inhalt

Inhalt	1
Einführung	2
Das Ziel und unsere Aufgabe	2
Bezug zur Security Layer Spezifikation	2
Installation und Konfiguration	3
Installation	3
Notwendige Konfigurationsdateien	4
Verwendung der PKCS#11-Schnittstelle	6
Verwendung der Security Kapsel im Remote-Modus (Serverbetrieb)	9
Weitere wichtige Anmerkungen	11
Inhalte von Infoboxen – Update von Infoboxen	11
Zusätzliche Funktionen der erweiterten Security Kapsel (Extended)	12
Verfügbare Funktionen nach Security Layer 1.2.0	12
Wechseln zwischen Security Layer Schnittstelle 1.1.0 und 1.2.0	12
Beschreibung des Dialoges beim Anlegen von Infoboxen	13
Infobox-Konfiguration – Verwaltung der Infoboxen in der Security Kapsel	14
Anhang A: Tabellarische Übersicht der implementierten Funktionen	15
Referenzen	17

# Einführung

## Das Ziel und unsere Aufgabe

Im Auftrag der BKA-IKT-Stabsstelle wurde die bereits vorliegende prototypische Implementierung des Security Layers auf den Stand der weiterentwickelten Spezifikation der IKT-Stabsstelle gebracht, sodass er als öffentlich verfügbares Protokollnormal für die Bürgerkarte für Anwendungsentwicklungen im Bereich des E-Government verwendet werden kann. Da bereits die erste Version durch A-SIT entwickelt wurde, können mit der Weiterentwicklung durch A-SIT weitgehende Synergien genutzt werden. Über die Anpassung der Security Kapsel selbst werden sich Tools zur Feststellung der Konformität von entstehenden Produkten und Schnittstellen mit den spezifizierten Eigenschaften und Funktionen der Security Kapsel als sinnvoll und notwendig erweisen.

Wesentliche Neuerungen im Rahmen der aktualisierten Implementierung:

- Aktualisierung des Prototyps auf Spezifikation Version 1.1
- Erweiterung der Transportbindung um https
- Bereinigung des aktuellen Prototyps
  - Manifest, bPK
  - Aktualisierungen (ixsil 1.2.0, xalan, xades...)
  - PKCS#11 Anbindung für aktuelle Karten (z.B. A-Trust)
  - Persistente Speicherung von Infoboxen
- Einbindung des Viewer und der PIN-Eingabe in WebBrowser
- Demontstratoren in Erweiterung der bestehenden Spezifikation
  - Webservice Personenbindung (europäische ID-cards bzw. Wiederholungsidentitäten, z.B. italienischer Personalausweis); Vorschlag einer Ausgestaltungsrichtlinie bei konkreten Karten
  - Server-basierende Security Kapsel - inkl. Vorschlag zur Erweiterung der Spezifikation (Transportbindung für Viewer / PIN-Eingabe)

## Bezug zur Security Layer Spezifikation

Die Security Kapsel in der vorliegenden Version basiert auf der Security Layer Schnittstellenspezifikation 1.1.0 [1]. Eine detaillierte Auflistung und Gegenüberstellung sämtlicher Anforderungen und Vorgaben der Security Layer Spezifikation und der durch die aktuelle Security Kapsel implementierten Funktionalitäten wird auf der Web-Seite der Security Kapsel veröffentlicht, wodurch ein umfassender Funktionsüberblick der aktuellen Kapsel gegeben wird.

Für weiterführende Informationen zu den umgesetzten Schnittstellen und Spezifikationen ([2],[3],[4],[5],[6]) rund um den Security Layer sei auf die Referenzliste im Anhang verwiesen.

## Installation und Konfiguration

Diese Applikation wurde, wie auch schon die vorhergegangene Version der Security Kapsel, vollständig als Java-Anwendung konzipiert und implementiert. Die vorliegende Version wurde für die *Java Runtime (JRE) Version 1.4.1\_06* entwickelt, und sollte daher auch zumindest mit

Java Runtime Version 1.4.1

verwendet werden.

### ACHTUNG

*Aufgrund Probleme mit der JRE 1.4.1\_06 bezgl. der Auflösung von externen Referenzen wird die Verwendung der*

#### ***Java Runtime Version 1.4.0\_(04)***

*empfohlen. In der aktuellen Distribution ist auch diese beigefügt. Die Verwendung höherer Versionen ist bei Bedarf mit Vorsicht zu prüfen.*

Darüber hinaus werden für die Verwendung der Security Kapsel folgende Java-Bibliotheken benötigt:

- IAIK:
  - IAIK JCE 3.0.3
  - IAIK IXSIL 1.2.0 SL
  - IAIK PKCS#11Wrapper 1.2.11
  - IAIK PKCS#11 Provider 1.1.7
  - IAIK iSaSilk 3.06
  - IAIK PKI Beta1
  - IAIK PKI CONFIG Beta1
  - IAIK LDAP Beta1
  - IAIK ECC Beta1
  - IAIK CMS Beta2internal1
- Xerces 2.4.0
- Xalan 2.5.1
- LOG4J 1.2.7

Sämtliche notwendigen und hier aufgeführten Java Bibliotheken sind in der Security Kapsel bereits enthalten.

## Installation

Die Security Kapsel kann sowohl mit als auch ohne Java Runtime bezogen werden. Im ersteren Fall ist die Umgebung und die beigefügte Runtime vorkonfiguriert. Somit kann die

Security Kapsel nach dem Dekomprimieren in ein auszuwählendes Zielverzeichnis mit der Datei `start.bat` gestartet werden.

Anderenfalls, das heißt wenn man nur die Klassendateien und Bibliotheken der Security Kapsel bezieht, sind nach dem Dekomprimieren der Dateien noch ein paar manuelle Eingriffe notwendig. Vor allem ist sicher zu stellen, dass eine Java Runtime in der Version 1.4.1 (oder höher) bereits installiert wurde. Darüber hinaus ist, um die von der vorliegenden Implementierung vorausgesetzte Xalan Bibliothek in der Version 2.5.1 verwenden zu können, diese in das so genannte ENDORSED-Verzeichnis der Java Runtime Installation zu kopieren. Dazu wird im LIB-Verzeichnis der JRE ein ENDORSED-Verzeichnis erstellt (sofern nicht bereits vorhanden), in das die Dateien der Xalan-Bibliothek kopiert werden müssen.

Zum Beispiel:

Java Runtime-Verzeichnis:	c:\J2sdk1.4.1_06\jre
ENDORSED-Verzeichnis:	c:\J2sdk1.4.1_06\jre\lib\endorsed
dorthin zu kopierende Datei:	xalan.jar (Version 2.5.1) - diese ist aus dem LIB-Verzeichnis der Security Kapsel Installation zu entnehmen

Abschließend ist noch die Startdatei (`start.bat`) der Security Kapsel zu modifizieren. Die Variable `JAVA_HOME` muss auf das Installationsverzeichnis der zu verwendeten Java Runtime gelegt werden.

Zum Beispiel: `JAVA_HOME=c:\J2sdk1.4.1_06\jre`

**Achtung:** Bei Verwendung einer bestehenden Java Runtime können durch das Kopieren der Xalan-Bibliothek in das ENDORSED-Verzeichnis der JRE-Installation auch andere Applikationen beeinflusst werden. Durch diese Maßnahme wird generell anstatt der ursprünglich verwendeten und mit der Java Runtime mitgelieferten Xalan-Bibliothek die Xalan-Bibliothek Version 2.5.1 verwendet. Dadurch kann das ordnungsgemäße Funktionieren andere Java-Programme welche die selbe Java Runtime benutzen beeinträchtigt werden.

Grundsätzlich ist daher die Verwendung einer separaten JRE zu bevorzugen.

## Notwendige Konfigurationsdateien

Die Einstellungen der Security Kapsel werden vorwiegend in Konfigurationsdateien, sog. properties-Dateien, vorgenommen. Die Grundeinstellung der Kapsel wurde so vorgenommen, sodass im Regelfall ohne weiteres Zutun die Kapsel lauffähig ist. Für besondere Funktionalitäten, wie beispielsweise der PKCS#11-Unterstützung oder erweiterte PKI-Funktionen, müssen einige der Parameter den jeweiligen Gegebenheiten angepasst werden. Die dafür notwendigen Maßnahmen werden am Ende dieses Kapitels angeführt.

Die wichtigsten Konfigurationsdateien sind:

Datei	Pfad <sup>1</sup>	Bedeutung
<code>securitykapsel.properties</code>	<code>/properties</code>	Hauptkonfigurationsdatei
<code>pkcs11.ini</code>	<code>/properties</code>	Festlegung der PKCS#11 Unterstützung für Signaturerstellungsgeräte
<code>pkiprofile.xml</code>	<code>/properties/pki/configuration</code>	PKI-Profildatei
<code>pkiconfig.xml</code>	<code>/properties/pki/configuration</code>	PKI-Konfigurationsdatei
<code>card.properties</code>	<code>/cardXY</code>	Konfigurationsdatei der jeweiligen Kartenabstraktion

Grau hinterlegte Felder der Tabelle beziehen sich auf Konfigurationsdateien, die im Regelfall nicht unbedingt durch den Anwender bearbeitet werden müssen.

Grundsätzlich sind alle hier genannten Konfigurationsdateien ausreichend kommentiert, sodass an dieser Stelle nur die zu erwartenden Einstellungen umrissen werden.

**securitykapsel.properties** ist die Hauptkonfigurationsdatei und beinhaltet alle wichtigen Grundeinstellungen der Security Kapsel. Hier können folgende Einstellungen vorgenommen werden:

- Anzahl der Kartenabstraktionen sowie Pfadangaben der Karten-Konfigurationsdateien
- Protokoll Einstellungen (Binding) – zB.: diverse Ports der Security Layer Schnittstelle, sowie die Ports des Remote-Mode.
- Proxy-Einstellungen
- Debug-Mode und Debug-Parameter

**pkcs11.ini** ist die Konfigurationsdatei der PKCS#11-Unterstützung und wird für das Initialisieren der PKCS#11-Kartenabstraktion zur Verwendung von Hardwaresignaturerstellungsgeräten benötigt. Eine kurze Erläuterung der hier notwendigen Einstellungen erfolgt im nächsten Abschnitt.

**pkiprofile.xml** und **pkiconfig.xml** sind die Konfigurationsdateien für die PKI-Unterstützung der Security Kapsel. Im Regelfall sollten diese Dateien nicht verändert werden. Entwickler und Benutzer, die über entsprechende Erfahrungen mit PKI verfügen, können anhand dieser Dateien das PKI-Verhalten der Signaturprüfung beeinflussen. Für die genauen Möglichkeiten und Einstellungen, die diese Konfigurationsdateien bieten, sei deshalb hier nur an die weiterführende Dokumentation der in der Implementierung verwendeten PKI-Module verwiesen [7].

---

<sup>1</sup> Pfadangabe relativ zum Installationsverzeichnis der Security Kapsel

**card.properties:** Neben den genannten Konfigurationsdateien gibt es für jede vorhandene Kartenabstraktion eine weitere Konfigurationsdatei. Initial wird die Security Kapsel mit drei Kartenabstraktionen ausgeliefert:

1. Karte 1 „Signaturkarte“
2. Karte 2 „Buergerkarte“
3. PKCS#11 Karte (für PKCS#11-Unterstützung, falls vorhanden)

Jede dieser vordefinierten sowie auch jede weitere Kartenabstraktionen ist in der Hauptkonfigurationsdatei (`securitykapsel.properties`) einzutragen, wobei auch der Name und der Pfad der für die Karte zu verwendenden Konfigurationsdaten anzugeben sind. Somit ist für jedes Kartenmodell ein eigenes Konfigurationsverzeichnis (z.Bsp. `card1/`) und eine eigene Konfigurationsdatei (z.Bsp. `card1/card.properties`) notwendig. Die Konfigurationsdatei enthält dabei im Wesentlichen folgende Einstellungen:

- Allgemeine Einstellungen wie Name, Grafikdetails, Darstellungsvorgaben
- Keyboxen: für jedes Schlüsselpaar wird eine eigene Keybox definiert. Für jede dieser Keyboxen müssen diverse Attribute vorgegeben werden, so zum Beispiel PIN, Name und Pfad zur betreffenden PKCS#12-Datei, JCE-Provider, etc.
- Infoboxen: für die sog. Infoboxen einer Karte wird eine separate Konfigurationsdatei (`infobox.xml`) verwendet. Darin sind entsprechend weitere Vorgaben zu den Daten und Eigenschaften der einzelnen Infoboxen enthalten. Generell werden die Infoboxen und deren Eigenschaften in dieser zusätzlichen Datei vordefiniert und ggf. mit Werten vorbelegt (z.B.: PIN, etc.). Weitere Zugriffe erfolgen dann ausschließlich über die Security Layer Schnittstelle und deren Infobox-Requests.

Nähere Details zur Bearbeitung der `card.properties`- und `infobox.xml`-Dateien sind den Kommentaren in den Dateien selbst zu entnehmen.

## Verwendung der PKCS#11-Schnittstelle

Die aktuelle Security Kapsel erlaubt es auch Hardware-Signaturerstellungsgeräte, wie zum Beispiel Smart Cards, zur Signaturerstellung zu verwenden. Die Einbindung dieser Geräte erfolgt über die PKCS#11 Schnittstelle wodurch grundsätzlich jede Signaturerstellungshardware, die diese Schnittstelle unterstützt, verwendet werden kann.

Um die PKCS#11 Schnittstelle der Security Kapsel verwenden zu können, muss in einer der Konfigurationsdateien der Security Kapsel (`\properties\pkcs11.ini`) der notwendige und im Zusammenhang mit ihrem Signaturerstellungsgerät zu verwendende PKCS#11 Treiber angegeben werden.

Zum Beispiel: folgender Eintrag ist in der Datei

```
INSTALLATIONVERZEICHNIS\properties\pkcs11.ini
```

vorzunehmen:

```
PKCS11_NATIVE_MODULE=C:\ERACOM\Cprov SDK\bin\sw\cryptoki.dll
```

Die entsprechende Treiberdatei, hier im Beispiel cryptoki.dll, ist vom jeweiligen Kartenhersteller zu beziehen.

Des Weiteren sind ggf. noch andere Einstellungen und Konfigurationen in der pkcs11.ini Datei vorzunehmen. Kommt es beispielsweise trotz korrekter Angabe der Treiberdatei während der PKCS#11-Initialisierungsphase zu Fehlermeldungen (zum Beispiel CKR\_ARGUMENTS\_BAD), so sollte in erster Linie folgender Eintrag umgesetzt werden:

```
MULTI_THREAD_INIT = false           bzw. MULTI_THREAD_INIT = true
```

Ausserdem kann es bei manchen Treibern notwendig sein, die Slot-Variable entsprechend der Hardwarekonfiguration explizit festzulegen. Dazu muss folgender Eintrag der PKCS11.ini-Datei entsprechend gesetzt werden:

```
z.B.: SLOT_ID = 1                   bzw. auf den entsprechenden Wert.
```

Alle weiteren Einstellungsmöglichkeiten der pkcs11.ini-Konfigurationsdatei sind optional und gut kommentiert. Im Rahmen der Installation empfiehlt es sich die weiteren Möglichkeiten und deren voreingestellte Werte zu überprüfen. Für weitere Informationen zu den Konfigurationsmöglichkeiten der PKCS#11-Unterstützung sei auf die Dokumentation des PKCS#11-Moduls verwiesen [8].

Das derart in die Security Kapsel eingebundene Signaturerstellungsgesetz wird in Form der PKCS#11-Kartenabstraktion repräsentiert. Analog zur Konfiguration der anderen Kartenmodelle ist auch für die PKCS#11-Karte eine card.properties-Konfigurationsdatei in ihrem Kartenverzeichnis vorgesehen. Hier müssen in den Keybox-Einstellungen die Bezeichner der den jeweiligen Schlüsselpaaren (Keypairs) zuzuordnenden Schlüssel des Signaturerstellungsgesetzes zugewiesen werden. Die Zuweisung erfolgt unter Verwendung der Schlüsselbezeichner (Label). Beispielhaft wäre folgender Eintrag für die erste Keybox der PKCS#11-Kartenabstraktion:

```
Datei:          data-pkcs11card/card.properties

Keybox.1.name = SecureSignatureKeypair
# PIN-OPTION NICHT VERAENDERN - MUSS hasNoPinDoConfirm BLEIBEN
Keybox.1.pin   = hasNoPinDoConfirm
Keybox.1.type  = PKCS11KeyBox
Keybox.1.JCEProvider = IAIK PKCS#11:1
Keybox.1.KeyName   =Max Buerger's IAIK Test CA ID
Keybox.1.CertificateName =Max Buerger's IAIK Test CA ID
```

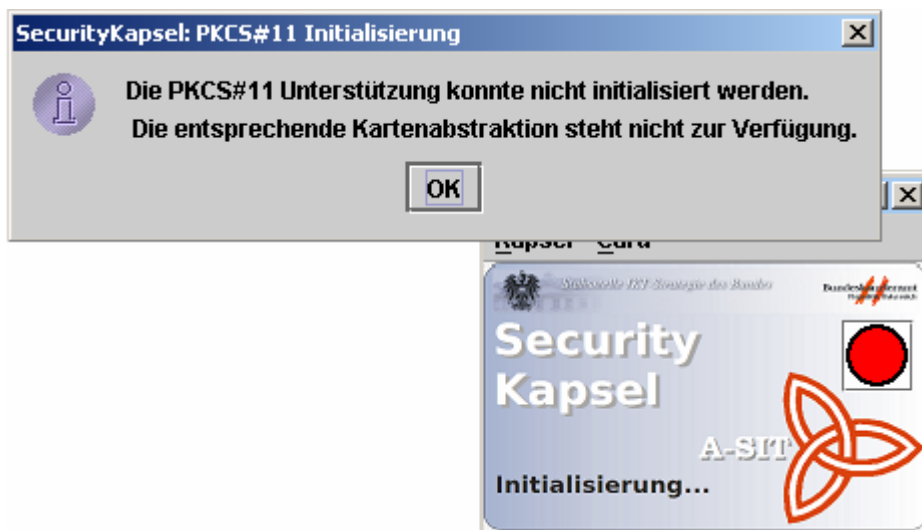
Die card.properties-Datei der PKCS#11-Karte ist weitestgehend mit zwei Keyboxen vordefiniert. Im allgemeinen sind daher nur die Bezeichner (Label) der Schlüsselpaare und der Zertifikate einzutragen (wie im obigen Beispiel für die erste Keybox gezeigt.) Ansonst stehen auch bei dieser Kartenabstraktion alle Einstellungsmöglichkeiten, wie von den anderen Kartenmodellen bekannt, zur Verfügung. Auch die Infoboxen der PKCS#11-Kartenabstraktion werden durch eine eigene properties-Datei festgelegt und repräsentiert. Näheres ist bitte aus den Kommentaren in den jeweiligen properties-Dateien zu entnehmen.

**Achtung:** Der Inhalt der Infoboxen werden auch bei der PKCS#11-Kartenabstraktion nur in der Security Kapsel gespeichert und nicht in den Speicherbereich des Signaturerstellungsgesetzes (Smartcard) geschrieben.

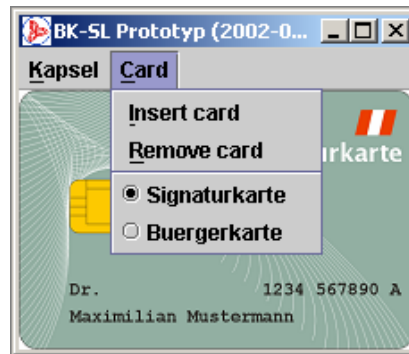
Nach der erfolgreichen Konfiguration der PKCS#11-Unterstützung wird beim Starten der Security Kapsel der angegebene Treiber geladen und das entsprechende Signaturerstellungsgesetz in die Kapsel eingebunden. Im Menü „Card“ steht ein weiteres Kartenmodell zur Auswahl:



Sollte kein PKCS#11 Treiber vorhanden, angegeben bzw. gefunden werden, so erscheint beim Starten der Security Kapsel ein entsprechender Hinweis:



Wird dieser bestätigt, so kann die Security Kapsel wie in der vorhergegangenen Version im vollen Funktionsumfang genutzt werden. Es fehlt lediglich im Menü „Card“ die Möglichkeit zur Auswahl der PKCS#11 Unterstuetzung (pkcs11-card). Somit bleiben die beiden Software emulierten Signaturerstellungsgesetze (Buergerkarte, Signaturkarte) zu Auswahl.



**Achtung:** Je nach verwendeter Signaturerstellungshardware/Smartcard kann vom Hersteller der Zugriff auf gewisse Funktionalitäten über die PKCS#11-Schnittstelle deaktiviert werden. So zum Beispiel können bei bestimmten österreichischen Smartcards keine qualifizierten Zertifikate oder gar sichere Signaturen mit PKCS#11-Befehlen abgerufen bzw. ausgelöst werden.

## Verwendung der Security Kapsel im Remote-Modus (Serverbetrieb)

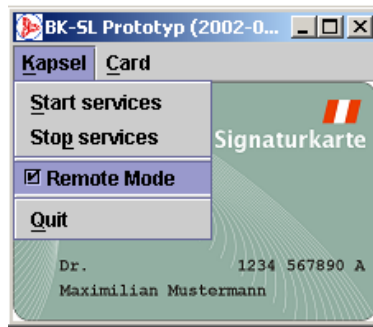
Die Security Kapsel bietet die Möglichkeit, die Security Layer Schnittstelle auch über den Clientrechner hinaus in Form eines minimalen Servers anderen Rechnern im Netzwerk zur Verfügung zu stellen. In diesem Modus, dem so genannten Remote-Modus, erfolgt die PIN-Abfrage sowie die optionale Anzeige der zu signierenden Dokumente durch ein Browser-Fenster des Clientrechners. Folgende Voraussetzungen müssen für die Verwendung dieses Remot-Betriebes am Clientrechner gewährleistet werden:

- die *Applikation*, die den Security Kapsel benützt, verwendet ein *Browser-Fenster* (zum Beispiel Web-Applikationen, etc.)
- der verwendete Browser lässt die Ausführung von *Java-Script* zu

Serverseitig muss die Security Kapsel für den Remote-Mode entsprechend konfiguriert werden. Im Wesentlichen müssen dazu in erster Linie folgende Einträge in der Hauptkonfigurationsdatei der Security Kapsel – `securitykapsel.properties` – beachtet werden:

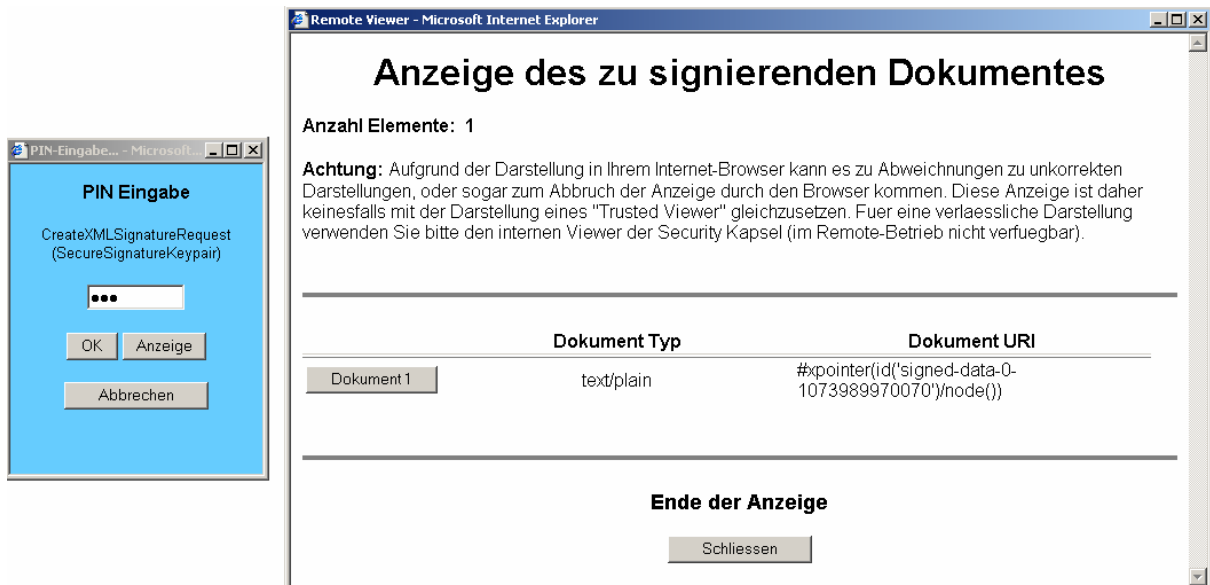
```
# Remote Control Binding
#
binding-remotecontrol-port = 4000
binding-remoteviewer-port = 4001
binding-remotecontrol-protocol = http
```

Hier sind hauptsächlich die beiden port-Einträge von Bedeutung. Diese legen fest, auf welchen Ports des Server-Rechner die Informationen zur PIN-Eingabe bzw. zur Voransicht der zu signierenden Dokumente dem Clientrechner zur Verfügung gestellt werden. In der Regel können die Voreingestellten Werte beibehalten werden. Sollten diese Ports bereits in Verwendung stehen, so sind die Einstellungen abzuändern. (Anmkg.: Der Protocol-Eintrag soll nicht verändert werden.)



Um zwischen dem normalen lokalen Betrieb der Security Kapsel zum Remote-Betrieb zu wechseln, ist im Menü Kapsel die Option „Remote Mode“ auszuwählen. Der Haken beim Menüpunkt zeigt den aktivierten Remote-Betrieb an. Ab sofort werden zur Anzeige der PIN-Abfrage sowie zur Voransicht der zu signierenden Dokumente Browserfenster des Clientrechners verwendet.

**Anmerkung:** Die Ports für den Remote-Betrieb werden nur aktiviert, wenn der Remote-Mode im Menü „Kapsel“ aktiviert wurde. Ansonst, d.h. nach Deaktivierung des Remote-Modes, sind diese Ports inaktiv und es können keine Informationen, wie zum Beispiel die Voransicht der zu signierenden Dokumente, von außen über diese Ports (remote) abgerufen werden.



**Achtung:** Im Remote-Modus muss darauf geachtet werden, dass während sich ein Security Layer Request in Abarbeitung befindet, kein weiterer angenommen werden kann. Dieser Remote-Mode ist daher nicht mit einer vollwertigen Serverlösung gleichzusetzen, bei der eine beliebige Anzahl von Client-Anfragen gleichzeitig behandelt bzw. gehalten werden können. Vielmehr wurde diese Betriebsart für den Fall konzipiert, dass sich ein Anwender seiner auf einem Rechner installierten Security Kapsel auch von anderen Rechnern aus bedienen kann.

## Weitere wichtige Anmerkungen

### Inhalte von Infoboxen – Update von Infoboxen

Werden in einer Infobox XML-Daten gehalten so ist darauf zu achten, dass in den Daten der Infobox selbst kein XML-Header (`<?xml version="1.0" ... ?>`) enthalten ist. Andernfalls kann, wenn der Infobox-Inhalt als XML ausgelesen werden soll, die Kapsel den Inhalt nicht in ein gültiges XML-Dokument integrieren. Speziell wenn Infoboxen durch Manipulation der entsprechenden Infobox-Dateien verändert werden ist darauf zu achten.

## Zusätzliche Funktionen der erweiterten Security Kapsel (Extended)

Die vorliegende Security Kapsel ist auch in einer erweiterten Version verfügbar (Security Kapsel extended). Diese ist als Interimsversion zwischen der aktuell vorliegenden nach Security Layer Spezifikation 1.1.0 implementierten Kapsel und einer künftigen Version gemäß Security Layer Spezifikation 1.2.0 zu sehen. Somit stellt bereits jetzt diese erweiterte Version der Kapsel grundsätzliche Funktionen gemäß Security Layer Spezifikation 1.2.0 [9] zur Verfügung.

## Verfügbare Funktionen nach Security Layer 1.2.0

Als wesentlichste Erweiterung können in dieser erweiterten Kapsel die Namespace-Einstellungen zwischen Security Layer Spezifikation 1.1.0 und 1.2.0 im Versions-Menü im laufenden Betrieb umgeschaltet werden. Nach dieser Umschaltung werden vorläufig alle bisherigen schon von Spezifikation 1.1.0 bekannten Funktionen unterstützt. Die XML-Requests und Responses sind dann allerdings im Namespace der neuen Security Layer Spezifikation gefasst. Zusätzlich wurden folgende neue Funktionen implementiert:

- Null-Operation (`NullOperationRequest`)
- Anlegen einer Infoboxen (`InfoboxCreateRequest`)
- Löschen einer Infobox (`InfoboxDeleteRequest`)
- Hashwert-Berechnung (`CreateHashRequest`)
- Hashwert-Verifikation (`VerifyHashRequest`)
- Berechnen von wbPK in Abhängigkeit des Infobox-spezifischen Parameter bei Personenbindungsabfragen (`IdentityLinkDomainIdentifier`)

## Wechseln zwischen Security Layer Schnittstelle 1.1.0 und 1.2.0

Die erweiterte Version der vorliegenden Kapsel bietet die Möglichkeit, die von der Kapsel angebotene Schnittstellenversion während des Betriebes umzuschalten. Dazu ist im Menü „Version“ eine entsprechende Checkbox vorgesehen.



Wird diese Checkbox aktiviert, so können zusätzlich zu den aus Spezifikation 1.1.0 bekannten Funktionen auch die oben erwähnten Requests der Security Layer Spezifikation 1.2.0 verarbeitet werden. Bei Deaktivierung verhält sich die Security Kapsel gemäß Spezifikation 1.1.0.

**Achtung:** Wird Versionsunterstützung 1.2.0 aktiviert, so steht der Remote-Mode nicht mehr zur Verfügung. Der dafür vorgesehene Menüpunkt wird in Folge der Umstellung deaktiviert. Sollte sich die Kapsel vor der Umstellung im Remote-Mode befinden, so wird dieser beendet und die Kapsel im Standardmodus (Interner-Modus) betrieben. Somit werden alle Benutzerdialoge, wie bspw. PIN-Eingabe, Dialog zur Erstellung von Infoboxen im Zuge des InfoboxCreateRequest etc., lokal am Arbeitsplatz über konventionelle Dialogfenster dargestellt.

Nach dem Rückstellen der Kapsel auf Version 1.1.0 steht der Remote-Mode wieder uneingeschränkt zur Verfügung.

## Beschreibung des Dialoges beim Anlegen von Infoboxen

Eine der Neuerungen im Rahmen der Security Layer Spezifikation 1.2.0 ist die Möglichkeit des Anlegens von Infoboxen per Infobox-Request. Laut Spezifikation wird dabei dem Anwender mittels Dialogmenü die Möglichkeit zur Abänderung der Eigenschaften der neu zu erstellenden Infobox gegeben. Es öffnet sich nach dem Verarbeiten des InfoboxCreateRequest folgender Dialog:

The screenshot shows a Windows-style dialog box titled "Angaben zum Anlegen einer Infobox". It contains the following fields and controls:

- Box-Name:** Text input field containing "TestInfobox".
- Box-Type:** Text input field containing "BinaryFile".
- Ersteller:** Text input field containing "A-SIT Testservice".
- Erklärung:** Text area containing "Infobox zur Demonstration."
- Zugriffsrechte-Lesen:** Text input field containing "(certifiedGovAgency:labs.cio.gv.at);".
- Zugriffsrechte-Schreiben:** Text input field containing "(certified:193.170.251.\*);(certifiedGovAgency:labs.cio.gv.at);".
- PIN-Lesezugriff:** A dropdown menu set to "confirm", followed by an empty text input field and a "Wiederholung:" label with an empty text input field.
- PIN-Schreibzugriff:** A dropdown menu set to "PIN", followed by an empty text input field and a "Wiederholung:" label with an empty text input field.

At the bottom, there is a prompt "Bitte machen Sie Ihre Eingabe..." and two buttons: "OK" and "Abbrechen".

Je nach Request kann der Benutzer folgende Parameter abändern:

- Zugriffseinschränkungen für Lesezugriffe (Zugriffsrechte-Lesen)
- Zugriffseinschränkungen für Update-Zugriffe (Zugriffsrechte-Schreiben)
- Benutzerinteraktion für Lesezugriffe (PIN-Lesezugriff)
- Benutzerinteraktion für Update-Zugriffe (PIN-Schreibzugriff)

Für Lese-/Schreibzugriffe können gemäß den in Spezifikation 1.2.0 beschriebenen Möglichkeiten folgenden Benutzerinteraktionen ausgewählt werden:

- none: Benutzer wird beim Zugriff nicht informiert
- info: Benutzer wird über Zugriff informiert
- confirm: Benutzer muss Zugriff bestätigen
- PIN: entspricht `confirmWithSecret`; Zugriff nur nach PIN-Eingabe möglich

Zur Festlegung von Zugriffseinschränkungen (Access-Authorization) können paarweise Einschränkungsklassen (`AuthenticationClass`) und Domainnamen bzw. Patterns (`RequesterID`) gesetzt werden (für nähere Beschreibung dieses Sicherheitskonzeptes siehe Detailspezifikation „Zugriffsschutz auf Funktionen der Bürgerkartenumgebung“ in [9]). In diesem Dialog können in den jeweiligen Einträgen diese Zugriffseinschränkungen paarweise wie folgt angegeben werden:

```
(AuthenticationClass:RequesterID);
```

Dadurch ist es einfach möglich, zu den im Request festgelegten Zugriffseinschränkungen neue hinzuzufügen, bzw. die vorgeschlagenen abzuändern. Es kann eine beliebige Anzahl derartiger Einschränkungen angegeben werden. Jedes dieser Klasse-Pattern-Paare ist in Klammern zu setzen und mit einem Semikolon (;) abzuschließen.

```
Beispiel: (pseudoanonym:*.cio.gv.at);(certified:193.170.*);
```

**Achtung:** Diese erweiterte Version der Security Kapsel bietet die Möglichkeit des Setzens derartiger Zugriffseinschränkungen, sowohl im Zuge des `InfoboxCreateRequests` als auch in der Konfigurationsdatei der Infoboxen selbst. Jedoch werden diese Angaben derzeit bei den Infobox-Zugriffen nicht berücksichtigt. Diesbezüglich wurde das Sicherheitskonzept der neuen Security Layer Spezifikation 1.2.0 noch nicht umgesetzt.

Sieht ein Infobox-Create-Request es nicht vor dem Anwender die Möglichkeit zu bieten die vorgeschlagene Attribute der neuen Infobox abzuändern, so bleiben die hier beschriebenen Eingabefelder inaktiv.

**Achtung:** Im Falle einer vorgesehenen PIN-Benutzerinteraktion ist die PIN jedenfalls durch den Anwender im Dialogfenster wiederholt anzugeben. Lese- und Schreib-PIN sind dabei voneinander unabhängig zu wählen. Bei fehlender Eingabe bleiben diese Passwörter ungesetzt.

## Infobox-Konfiguration – Verwaltung der Infoboxen in der Security Kapsel

Eine weitere Neuerung dieser erweiterten Version der Security Kapsel ist die Repräsentation der Infoboxen in der Kapsel selbst mittels eines XML-Files. Für jede Kartenabstraktion ist ein derartiges XML-File vorzusehen (z.B. `infobox.xml`) und in der Hauptkonfigurationsdatei der jeweilige Karte (`card.properties`) einzutragen (`infobox.file=...`). Die Form dieser Datei ist an die Form des `InfoboxCreateRequests` gemäß Security Layer Spezifikation 1.2.0 angelehnt:

```

<InfoboxDescriptor>
  <Infobox>
    <InfoboxIdentifier>IdentityLink</InfoboxIdentifier>
    <Creator>Testaussteller</Creator>
    <Purpose>Zur Demonstration.</Purpose>
    <ReadUserConfirmation>confirmWithSecret</ReadUserConfirmation>
    <UpdateUserConfirmation>confirmWithSecret</UpdateUserConfirmation>
    <ReadPin>1234</ReadPin>
    <UpdatePin>5678</UpdatePin>
    <ReadAccessAuthorization>
      <RequesterID AuthenticationClass="anonym">String</RequesterID>
    </ReadAccessAuthorization>
    <UpdateAccessAuthorization>
      <RequesterID AuthenticationClass="anonym">String</RequesterID>
    </UpdateAccessAuthorization>
    <InfoboxType>BinaryFile</InfoboxType>
    <DataFile>card1/IdentityLink.xml</DataFile>
  </Infobox>
  <Infobox>
    ...
  </Infobox>
</InfoboxDescriptor>

```

Eine detaillierte Beschreibung der verwendeten Elemente ist daher in [9] zu finden. Zusätzlich werden allerdings für jede Infobox bzw. auch für jeden Key einer Array-Infobox die Daten in einer eigenen Datei abgelegt, die jeweils in einem `DataFile`-Element festgelegt wird:

```
<DataFile>card1/IdentityLink.xml</DataFile>
```

Die Adressierung der so angegebenen Dateien erfolgt relativ zum Hauptverzeichnis der Security Kapsel sofern nicht absolute Pfade angegeben werden. Beim Anlegen neuer Infoboxen mittels `InfoboxCreate-Request` werden neue Dateien im jeweiligen Kartenverzeichnis angelegt und in der `infobox.xml`-Datei eingetragen.

## Anhang A: Tabellarische Übersicht der implementierten Funktionen

Folgende Tabelle zeigt einen Überblick der in der vorliegenden Security Kapsel implementierten Funktionen. Zusätzlich werden nochmals die im SL12-Mode zur Verfügung stehenden ausgewählten Funktionalitäten der Security Layer Spezifikation 1.2 zusammengefasst.

		Security Layer Spezifikation Version 1.1											zusätzlich aus Spezifikation Version 1.2 <sup>2</sup>				
Release Version		CreateCMSSignatureRequest	VerifyCMSSignatureRequest	CreateXMLSignatureRequest	VerifyXMLSignatureRequest	InfoboxAvailableRequest	InfoboxReadRequest	InfoboxUpdateRequest	CreateSessionKeyRequest	CreateSymmetricSecretRequest	GetPropertiesRequest	GetStatusRequest	CreateHashRequest	VerifyHashRequest	InfoboxCreateRequest	InfoboxDeleteRequest	NullOperationRequest
SeLa <b>N</b>	1.0.0	—	—	X	X	X	X	X	—	—	—	X	—	—	—	—	—
SeLa <b>Next</b>	1.0.3	—	—	X	X	X	X	X	—	—	—	X	X	X	X	X	X

Legende: — ... nicht verfügbar  
X ... verfügbar

<sup>2</sup> nur im „Version 1.2“-Mode verfügbar

## Referenzen

- [1] Arno Hollosi, Gregor Karlinger  
**Schnittstellenspezifikation des Security-Layers der Bürgerkarte Version 1.1.0.**  
abrufbar unter: <http://www.buergerkarte.at>  
Chief Information Office AUSTRIA, August 2002.
- [2] Arno Hollosi, Gregor Karlinger  
**Einführung zum Security-Layer für das Konzept Bürgerkarte Version 1.1.0.**  
abrufbar unter: <http://www.buergerkarte.at>  
Chief Information Office AUSTRIA, August 2002.
- [3] Arno Hollosi, Gregor Karlinger  
**Bindung des Schnittstellenprotokolls des Security-Layers für das Konzept Bürgerkarte an Transportprotokolle Version 1.1.0.**  
abrufbar unter: <http://www.buergerkarte.at>  
Chief Information Office AUSTRIA, August 2002.
- [4] Arno Hollosi, Gregor Karlinger  
**Fehlercodes zur Schnittstellenspezifikation des Security-Layer für das Konzept Bürgerkarte Version 1.1.0.**  
abrufbar unter: <http://www.buergerkarte.at>  
Chief Information Office AUSTRIA, August 2002.
- [5] Arno Hollosi, Gregor Karlinger  
**Standardisierte Key- und Infoboxen des Security-Layers für das Konzept Bürgerkarte Version 1.1.0.**  
abrufbar unter: <http://www.buergerkarte.at>  
Chief Information Office AUSTRIA, August 2002.
- [6] Arno Hollosi, Gregor Karlinger  
**Errata der Spezifikationen zum Security-Layer der Bürgerkarte Version 1.1.0.**  
abrufbar unter: <http://www.buergerkarte.at>  
Chief Information Office AUSTRIA, August 2002.
- [7] Wolfgang Bauer  
**IAIK PKI (Dokumentation zur IAIK-PKI-Library).**  
der Security Kapsel Dokumentation beigelegt (iaik\_pki.pdf)  
Inst. f. Angew. Informationsverarbeitung u. Kommunikationstechnologie (IAIK), 2003.
- [8] Karl Scheibelhofer  
**IAIK JCE Provider for PKCS#11, Version 1.1.7: Information.**  
abrufbar unter : [http://jce.iaik.tugraz.at/products/15\\_PKCS11\\_Provider/index.php](http://jce.iaik.tugraz.at/products/15_PKCS11_Provider/index.php)  
Inst. f. Angew. Informationsverarbeitung u. Kommunikationstechnologie (IAIK), 2003.
- [9] Arno Hollosi, Gregor Karlinger  
**Das Modell Bürgerkarte - Version 1.2.0 PREVIEW.**  
<http://www.buergerkarte.at/konzept/securitylayer/spezifikation/preview-20040305>  
Chief Information Office AUSTRIA, März 2004.