VDV-Kernapplikation

Ergänzung zur NM Spezifikation

Stand: 22.11.2010

Thema:	Applikationsdownload und Initialisierung
Dateiname:	SPEC_LUKA_ERW-NM_V1.0.doc
Version:	1.0
Erstellt am:	23.11.2010
Zuletzt geändert am:	19.04.2011 08:58
Ersteller:	VDV KA KG eSol GmbH

Inhaltsverzeichnis

I Abbildungen

Tabellen

Dokumentenorganisation

Änderungshistorie

Version	Datum	Person	Beschreibung
0.1	2010-06-07	G. Galka J. Lutgen	Initiale Version, Strukturierung des Dokuments Entwurf Konfigurationsablauf
0.2	2010-08-23	G. Galka	Beschreibung der Initialisierung und Konfiguration Restrukturierung Spezifikation der technischen Voraussetzungen
0.3	2010-08-31	J. Lutgen	Allg. Review, Überarbeitung der Kommandos und Datenstrukturen. Differenzierung NM für NFC-Handset von sonstigen NM.
0.4	2010-09-01	G. Galka	Einarbeitung der Review Kommentare. Konkretisierungen, Erweiterung der Lebenszyklus-Beschreibung
0.5	2010-09-24	H. Boksem	Beschreibung "Einbringung der Applikation" Abschnitt 4
0.6	2010-09-24	J. Lutgen	Einarbeitung Review Kommentare Überarbeitung Formatierung
0.95	2010-09-24	G.Galka	Überprüfung der Zustandsdefinitionen. Finale Einarbeitung von Kommentaren.
0.96	2010-11-14	J. Lutgen	Überarbeitung des Gesamtdokuments
0.99	2010-11-18	G.Galka	Korrekturen nach Review Definition der Schlüssel-ID für PrK.Config auf Org.-ID|KID Konkretisierung der einzubringenden Daten
1.0	2010-11-22	J. Lutgen	Überarbeitung Konfigurationsdaten, Abschluss
	2011-03-21	WSC	Finalisieren

1 Übersicht

Die in der in der Spezifikation des VDV-KA Nutzermedium [VDVNM] definierten Prozesse wie Applikationsausgabe oder Ausgabe von AFB oder EFS setzten voraus, dass sich das Nutzermedium im "Zustand nach Initialisierung" befindet. Im initialisierten Zustand enthält das Nutzermedium neben den vorbelegten Strukturen für Applikation, Logbuch, Verzeichnis und Kundendaten sowohl die eindeutige Applikationsnummer (appInstanz-ID), als auch das individuelle RSA Schlüsselpaar PkP.NM, sowie das Zertifikat über den öffentlichen Schlüssel PuK.NM des Schlüsselpaares.

Das vorliegende Dokument definiert die Einbringung der für den Wirkbetrieb erforderlichen Daten und die Herstellung des "Zustands nach Initialisierung" gemäß der Spezifikation [VDVNM]

1.1. Terminologie

Im Rahmen des Dokuments werden folgende Begriffe benutzt

Nutzermedium	Smart Card, die den VDV-KA Sicherheitsanforderungen entspricht. Es gibt Nutzermedien, die Software nach dem GlobalPlatform Standard nachladen können und Nutzermedien, die keine Applikationen nachladen können. Letztere beinhalten bereits eine NM Applikation.
NM Applikation	Software die in einer Smart Card (Nutzermedium) läuft und die Funktionen eines VDV-KA Nutzermediums bereitstellt.
Initialisierung	Als Initialisierung wird der Prozess der Installation (Laden) der NM Applikation auf ein Nutzermedium und die Einbringung der für den Wirkbetrieb erforderlichen Daten bezeichnet.
Konfiguration	Als Konfiguration wird der Prozess bezeichnet, der nach der Installation (Laden) der NM Applikation ansetzt und die NM Applikation in den Zustand „Initialisiert“ überführt. Im Rahmen der Konfiguration wird das NM-Authentisierungsschlüsselpaar von der NM Applikation erzeugt, den öffentlichen Teil des Schlüssels ausgelesen (PuK.NM), das entsprechende Zertifikat von der VDV-PKI beantragt und anschließend das Zertifikat sowie das dazugehörige Sub-RA-Zertifikat der VDV-PKI in die NM-Applikation eingebracht. Nach Durchführung der Konfiguration befindet sich die NM Applikation im Zustand "Initialisiert" nach [VDVNM].

Tabelle 1 Terminologie

1.2. Rollen

Hersteller	Der Hersteller stellt Smart Card Plattformen (Nutzermedien) her, die den Anforderungen der VDV-KA Standards an Nutzermedien genügen und die entweder • eine Applikation nach der VDV-KA NM Spezifikation [VDVNM] bereits nach der Herstellung enthält, die gemäß dieser Spezifikation vom Konfigurator konfiguriert werden kann oder • das Nachladen von ausführbarem Code mit Hilfe der GlobalPlatform Mechanismen zulassen und so eingestellt sind, dass NM Applikationen vom App-Loader geladen und vom Konfigurator konfiguriert werden können.
App-Loader	Der App-Loader • Ist für die Authentizität und Vertraulichkeit der NM Applikation im Rahmen des Ladeprozesses verantwortlich und erzeugt das Schlüsselpaar PkP.Config sowie die Datensätze PrK.Config Daten und PuK.Root Daten; • Lädt die NM Applikation in das NM und überführt die Applikation in den Zustand CONFIGURATION. Im Rahmen des Zustandsübergangs wird vom App-Loader der Datensatz PrK.Config Daten eingebracht.
App-Hersteller	Der App-Hersteller • Entwickelt und pflegt die NM Applikation; • Stellt sicher, dass die NM Applikation spezifikationskonform ist; • Übergibt die NM Applikation an den App-Loader.
Konfigurator	Der Konfigurator • Handelt im Auftrag eines KVP; • Authentifiziert NM Applikationen mit Hilfe des PuK.Config und bringt die für den Wirkbetrieb erforderlichen Daten in das Nutzermedium ein. • Überführt die NM Applikation nach abgeschlossener Konfiguration in den Zustand OPERATIONAL (OP). • Betreibt ein Initialisierungssystem, mit dem NM Applikationen initialisiert werden können.

Tabelle 2: Rollen

2 Initialisierung des VDV-KA Nutzermediums

Das VDV-KA Nutzermedium durchläuft in seinem Lebenszyklus drei Zustände, in denen unterschiedliche Funktionen zur Verfügung stehen. Unterstützt das Nutzermedium nicht das Laden von Applikationen, wird davon ausgegangen, dass die NM Applikation bereits fester Bestandteil des Nutzermediums im Zustand NM APP SELECTABLE ist und der Zustand NM APP LOADABLE entfällt.

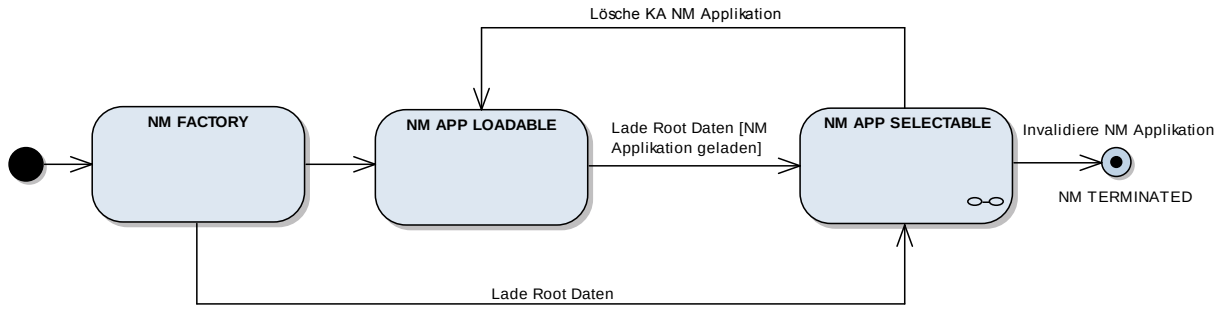

Abbildung 1 Lebenszyklus des Nutzermediums

2.1. Zustand NM FACTORY

Im Zustand NM FACTORY stehen Betriebssystemspezifische Kommandos zur Verfügung, die die Überführung des Nutzermediums in einen Folgezustand ermöglichen.

Ist die NM Applikation bereits geladen, so wird durch die Einbringung der Root-Daten die Applikation in den Folgezustand NM SELECTABLE überführt.

Ist die NM Applikation nicht im Nutzermedium enthalten, so wird durch die Einbringung herstellerspezifischer Daten das Nutzermedium in den Zustand NM APP LOADABLE überführt.

2.2. Zustand NM APP LOADABLE

Im Zustand NM APP LOADABLE ist keine NM Applikation selektierbar. Für die Einbringung der Applikation muss das Nutzermedium dem GlobalPlatform Standard entsprechen. Anforderungen, Prozesse zur Applikationseinbringung sowie Zustände sind in Kapitel 4 beschrieben.

Nach dem erfolgreichen Laden der NM Applikation wird das Nutzermedium durch die Einbringung der Root Daten (s. Kap. 3) in den Zustand NM APP SELECTABLE überführt.

2.3. Zustand NM APP SELECTABLE

Der Zustand NM APP SELECTABLE wird durch die Einbringung der Root Daten (s. Kap. 3) durch den App-Loader oder Hersteller erreicht.

Im Zustand NM APP SELECTABLE sind je nach Zustand der Applikation entweder das Kommando zur Konfiguration CONFIGURE (s. Kap. 5.8) oder – nach erfolgter Konfiguration – die in [VDVNM] festgelegten Kommandos verfügbar.

Die im Zustand NM APP SELECTABLE verfügbaren Kommandos und Prozesse sind in Abschnitt 5 beschrieben.

3 Root Daten

Die Root Daten werden je nach Ausprägung des Nutzermediums vom Hersteller oder vom App-Loader in das Nutzermedium eingebracht. Dies geschieht bei der Überführung des Nutzermediums in den Zustand NM APP SELECTABLE (s. 2.3). Auf die eingebrachten Root-Daten wird im Rahmen der Konfiguration zurückgegriffen, um die NM Applikation gegenüber dem Konfigurator zu authentisieren und Sessionkeys auszuhandeln, der für die Sicherung der Kommunikation mit der NM Applikation während der Konfiguration verwendet werden.

3.1. PrK.Config Daten

Zur Authentifizierung der NM Applikation durch den Konfigurator wird vom App-Loader der PrK.Config eingebracht. Der PuK.Config kann anhand der vom App-Loader festgelegten Schlüssel-ID ermittelt werden. Die Schlüssel-ID ist 6 Byte lang und setzt sich aus der Org.-ID des App-Loader und einer 4 Byte langen App-Loader spezifischen ID zusammen. Der App-Loader stellt sicher, dass die Schlüssel-ID eindeutig ist.

Position	Länge	Wert	Beschreibung
1	1	'E2'	Datenobjekt PrK.Config
2	*3*	'82 XX XX'	Länge des Datenobjekts
3	1	'81'	Tag Schlüssel-ID des PrK.Config
4	1	'06'	Länge der Schlüssel-ID
5	2	'XX XX XX XX XX XX'	Org.-ID App-Loader (2 Byte) | App-Loader spezifische Schlüssel-ID (4 Byte)
6	1	'82'	Tag Schlüsseltyp
7	1	'02'	Länge des Schlüsseltyps
8	2	'00 01'	RSA CRT 2048 Bit
9		'E3'	Tag Schlüsseldaten PrK.Config in CRT Format
10		'82 XX XX'	Länge der Schlüsseldaten PrK.Config
11		'84'	Tag für den ersten Primfaktor des Modulus
12		'81 XX'	Länge
13		'XX … XX'	Wert von P, wobei PQ= Modulus
14		'85'	Tag für zweiten Primfaktor des Modulus
15		'81 XX'	Länge
16		'XX … XX'	Wert von Q, wobei PQ= Modulus
17		'86'	Tag für Exponent 1
18		'81 XX'	Länge
19		'XX … XX'	Wert von DP1 = d mod P-1 (wobei d der private Exponent ist)
20		'87'	Tag für Exponent 2
21		'81 XX'	Länge
22		'XX … XX'	Wert von DQ1 = d mod Q-1 (wobei d der private Exponent ist)

Position	Länge	Wert	Beschreibung
23		'88'	Tag für Koeffizient
24		'81 XX'	Länge
25		'XX … XX'	Wert von R = 1/Q mod P

Tabelle 3 PrK.Config Datenobjekt

3.2. PuK.Root Daten

Der für die Zertifikatsprüfungen erforderliche öffentliche Schlüssel der VDV-KA Root-CA (PuK.Root) sowie die dazugehörige CAR müssen vor der Konfiguration in das Nutzermedium eingebracht werden. Die CAR dient während der Konfiguration zur Identifizierung der Root-CA.

Tag	**Länge**	**Wert**				
’7F 21’	’82 01 3F‘					
		Tag	**Länge**	**Wert**		
		’5F 29’	’01’	’07’ = CPI		
		’42’	’08’	’XX .. XX’ = CAR (8 Byte)		
		’5F 20’	’0C’	’XX .. XX’ = CHR (12 Byte)		
		’5F 4C’	‘07’	‘XX … XX’ = CHA (7 Byte)		
		’5F 24’	’04’	’JJ JJ MM TT’ = EOV		
		’06’	‘09‘	’2A 86 48 86 F7 0D 01 01 05’ = OID		
		’7F 49’	’82 01 01‘	Öffentlicher Schlüssel		
				Tag	**Länge**	**Wert**
				’81’	’81 F8‘	’XX … XX’ = Modulus (1984 Bit)
				’82’	’04’	’XX .. XX’ = öffentlicher Exponent [4 Byte]

Tabelle 4 PuK.Root Datenobjekt

4 Einbringung der NM Applikation

Wird im Zusammenhang mit einem VDV KA NM Service ein Security Domains und / oder Cardlet(s) auf dem SE benötigt, so sollen diese mit den Mechanismen des GlobalPlatform Standards auf das SE geladen werden. Laden dieser.

In Abhängigkeit der gewählten Umgebung, Handy und SE Type gibt es verschiedene Modelle zum Aufbringen von Security Domains und Cardlets auf das SE. In diesem Anhang wird das Aufbringen der Security Domain nicht weiter definiert.

Dies bezogene, wenn ein Security Domain aufgebracht werden muss, müssen die notwendige Schlüssel (wie im GlobalPlatform Spezifikation vorgeschrieben) vom SAM bezogen werden.

Dieser Kapitel beschreibt die verschiedenen Zustände bei der Aufbringung von VDV KA NM Services, der Erzeugung und Verarbeitung des Schlüssels PkP Config sowie der Einbringung der Root Daten. Der hier beschriebene Prozess zur Aufbringung der NM Applikation folgt dem im GlobalPlatform Standard dargestellten Lifecycle Management .

Bevor eine Applikation installiert werden kann, muss es geladen werden und den Status LOADED in der GlobalPlatform Registry bekommen. Dabei wurde das so genannte „Executable Module“ entweder bereits vorab geladen oder es wird mit dem Kommando Load File geladen.

Die folgenden Lifecycle-Zustände sind für das Laden einer NM Applikation relevant.

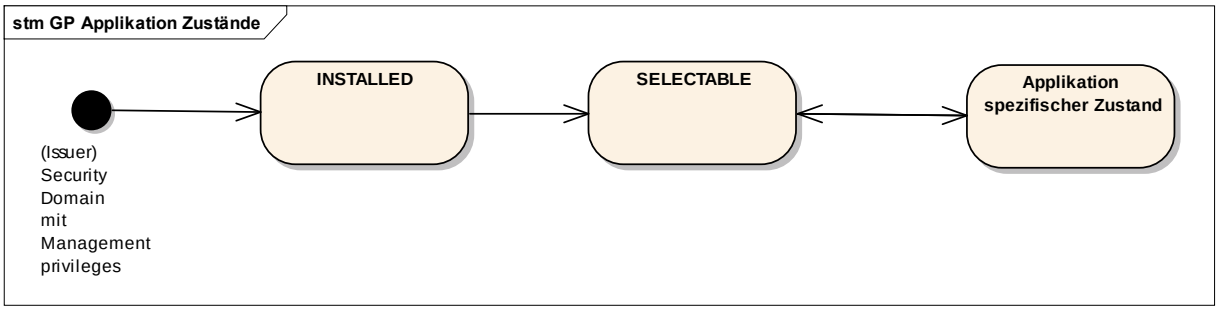

Abbildung 2 GlobalPlatform Applikation-Zustände

Die Applikation wird aus dem vorher geladenen „Executable Module“ in der Issuer Security Domain oder einer KA NM spezifischen Security Domain mit „Management Privileges“ instanziiert. . Danach werden die Lifecycle-Zustände der Applikation durch die „Global Platform Environment“ (genannt OPEN) verwaltet. Nach diesem Schritt ist die Applikation in der GlobalPlatform Registry mit dem Zustand INSTALLED eingetragen.

Nachdem die Applikation installiert wurde (d.h. in den Zustand INSTALLED gebracht), kann sie im Sinne von GlobalPlatform personalisiert werden. Im Falle der VDV KA NM Applikation handelt es sich hierbei um die Einbringung der Root Daten. Hierfür ist die „Personalization Support“ in GlobalPlatform anzuwenden, d.h. das Kommando INSTALL (für Personalisierung) gefolgt vom Kommando STORE DATA.

Nachdem die Applikation im Sinne von GlobalPlatform personalisiert ist, wird sie in den Zustand SELECTABLE versetzt. Damit wird die Applikation von außen selektierbar und kann Kommandos direkt empfangen.

Weitere Zustände werden dann durch die Applikation selber verwaltet (applikationsspezifische Zustände).

Im Zustand LOCKED, der allen anderen Lifecycle-Zuständen übergeordnet ist, ist die Applikation nicht mehr von außen selektierbar. Dieser Zustand kann aus verschiedenen Gründen durch die Karte selber oder durch eine externe Entität gesetzt werden und kann nur durch die Security Domain (mit „LOCK Privileges“) wieder abgesetzt werden.

Die folgenden Abschnitte beschreiben die entsprechenden Prozesse in mehr Detail.

4.1. Schritte um ein Cardlet zu laden, installieren, personalisieren und selectierbar zu machen.

1. Selektieren des Cardlet Managers.
2. Ausführen einer gegenseitigen Authentifizierung (Kommandos Initialize Update und External Authentication).

Falls die Personalisierung des Cardlets nicht unmittelbar auf die Installierung folgen soll, werden folgende Schritte durchgeführt:

1. Laden des Cardlets in die Karte mit dem Kommando INSTALL [for load] (der Life Cycle Status des Executable Load File des Cardlets ist LOADED).
   Hinweis: Das Kommando INSTALL [for load] dient als Anfrage zum Laden.
2. Registerierung des Cardlets mit dem Kommando INSTALL [for install] (der Life Cycle Status des Cardlets ist dann INSTALLED).
   Hinweis: Das Kommando INSTALL [for install] wird benutzt um die Installierung der Applikation zu beantragen.
3. Änderung des Status zu Selektierbar mit dem Kommando INSTALL [for make selectable] (der Life Cycle Status ist dann SELECTABLE).
   Hinweis: Das Kommando INSTALL [for make selectable] wird benutzt um eine vorher installierte Applikation selektierbar zu machen.

Falls die Personalisierung des Cardlets direkt nach der Installierung erfolgen soll, so wird das Kommando INSTALL [for personalization] wie folgt verwendet.

1. Laden des Cardlets in die Karte mit dem Kommando INSTALL [for load] (der Life Cycle Status des Executable Load File des Cardlets ist LOADED).
   Hinweis: Das Kommando INSTALL [for load] dient als Anfrage zum Laden.
2. Änderung des Status zu Selektierbar mit dem Kommando INSTALL [for personalization] (der Life Cycle Status des Cardlets ist jetzt SELECTABLE).Das Cardlet erwartet die Personalisierung jetzt mit STORE DATA Kommandos.
3. Senden der vorbereiteten STORE DATA Kommandos zur Personalisierung des Cardlets (der Life Cycle Status des Cardlets ist dann PERSONALIZED).

4.1.1. GlobalPlatform- und Cardlet-States

SI	STEPS	GP state	CARDLET STATE
1	INSTALL [for load]	OP_READY/INITIALIZED/SECURED	LOADED
2	INSTALL [for install]	OP_READY/INITIALIZED/SECURED	INSTALLED
3	INSTALL [for make selectable]	OP_READY/INITIALIZED/SECURED	SELECTABLE
4	INSTALL [for personalization]	OP_READY/INITIALIZED/SECURED	SELECTABLE
5	STORE DATA	OP_READY/INITIALIZED/SECURED	PERSONALIZED

Tabelle 5 GlobalPlatform- und Cardlet-States

4.2. Schritte zur Installation einer Supplementary Security Domain - Zustände, die das Laden, Installieren und Personalisieren eines Cardlets unterstützen.

Die Security Domain muss personalisiert sein, um das Laden, Installieren und Personalisieren eines Cardlets zu unterstützen.

Die Installierung und Personalisierung einer vorab geladenen Supplementary Security Domain (SSD) erfolgt in folgenden Schritten:

1. Selektieren der ISD;
2. Öffnen eines sicheren Kanals;
3. Absenden der Kommandos INSTALL [for install & make selectable] für das vorab geladene SD Module. Danach ist die SSD im Zustand SELECTABLE (optional müssen hier Installier-Parameter mitgegeben werden, falls die SSD „Extradition“ akzeptieren soll);
4. Selektieren der SSD;
5. Öffnen eines sicheren Kanals (mit default keys);
6. Personalisierung der SSD (Schreiben von Schlüsseln für Secure Messaging).

Folgendes muss gemacht werden, um ein Cardlet zu Installatieren (associating) in die SSD:

1. Selektieren der ISD;
2. Öffnen eines sicheren Kanals ;
3. Install [for load] für das Cardlet mit Angabe der zu assoziierenden SSD;
4. Install [for install & make selectable].

4.2.1. GlobalPlatform-, Security Domain- und Cardlet-Zustände

SI	Schritt	GP Zustand	SD Zustand	CARDLET Zustand
1	INSTALL [for load]	OP_READY/INITIALIZED/SECURED	PERSONALIZED	LOADED
2	INSTALL [for install]	OP_READY/INITIALIZED/SECURED	PERSONALIZED	INSTALLED
3	INSTALL [for make selectable]	OP_READY/INITIALIZED/SECURED	PERSONALIZED	SELECTABLE
4	INSTALL [for personalization]	OP_READY/INITIALIZED/SECURED	PERSONALIZED	SELECTABLE
5	STORE DATA	OP_READY/INITIALIZED/SECURED	PERSONALIZED	PERSONALIZED

Tabelle 6 GlobalPlatform-, SecurityDomain- und Cardlet-States

4.3. Referenzen

4.3.1. Cardlet Life Cycle Zustände

Die Card Life Cycle Zustände werden von der OPEN (Global Platform Umgebung) administriert. Die ISD erbt die Card Life Cycle Zustände:

OP_READY

INITIALIZED

SECURED

CARD_LOCKED

TERMINATED

4.3.2. Executable Load File/ Executable Module Life Cycle

Das Executable Load File kann nur den Status LOADED haben.

4.4. Applikation Life Cycle Zustände

INSTALLED

SELECTABLE

LOCKED

Hinweis: Wenn eine Applikation oder Security Domain sich bereits im Zustand SELECTABLE befindet, wird sie die eigenen Life Cycle Zustände selbst kontrollieren und administrieren. Übergänge zwischen diesen Zuständen werden durch die Applikation bzw. Security Domain definiert und nicht durch die OPEN kontrolliert.

4.4.1. Security Domain Life Cycle Zustände

INSTALLED

SELECTABLE

PERSONALIZED

LOCKED

5 Konfiguration des Nutzermediums

Nach der Einbringung der Root Daten befindet sich die KA-NM Applikation im Zustand CONFIGURATION. Die Applikation kann selektiert werden und steht für die gesicherte Einbringung der für den VDV-KA Wirkbetrieb erforderlichen Daten bereit.

Neben dem Kommando SELECT FILE unterstützt die NM Applikation im Zustand CONFIGURATION ausschließlich das Kommando CONFIGURE (s. 5.1). Andere Kommandos werden von der selektierten NM Applikation mit einem entsprechenden Fehlercode abgelehnt.

SELECT FILE entspricht dem in [VDVNM] spezifizierten SELECT FILE Kommando, mit der Ausnahme, dass ausschließlich die Versionsinformation der NM Applikation in den Antwortdaten ausgegeben wird.

Mit dem Kommando CONFIGURE werden Daten in die Applikation eingebracht, die zur Überführung der Applikation in den Zustand OPERATIONAL erforderlich sind. Die Datenstrukturen

- Applikationsdaten,
- Applikationslogbuch,
- WES (Datenstruktur angelegt, nicht benutzt),
- Kundendaten,
- PIN und PUC mit Default-Werten,
- Prioritäten und
- Letzte Transaktion

werden von der Applikation automatisch angelegt und sind mit den spezifizierten Werten (s. [VDVNM] vorbelegt.

Der Zustand OPERATIONAL entspricht dem "Zustand nach Initialisierung" nach VDV-KA Standard [VDVNM]. Erst nach der Überführung in den Zustand OPERATIONAL können die in [VDVNM] spezifizierten Kommandos und Prozesse mit der Applikation ausgeführt werden.

5.1. Zustände der NM Applikation

Die NM Applikation durchläuft im Lebenszyklus die Sammelzustände CONFIGURATION (CFG) und OPERATIONAL (OP).

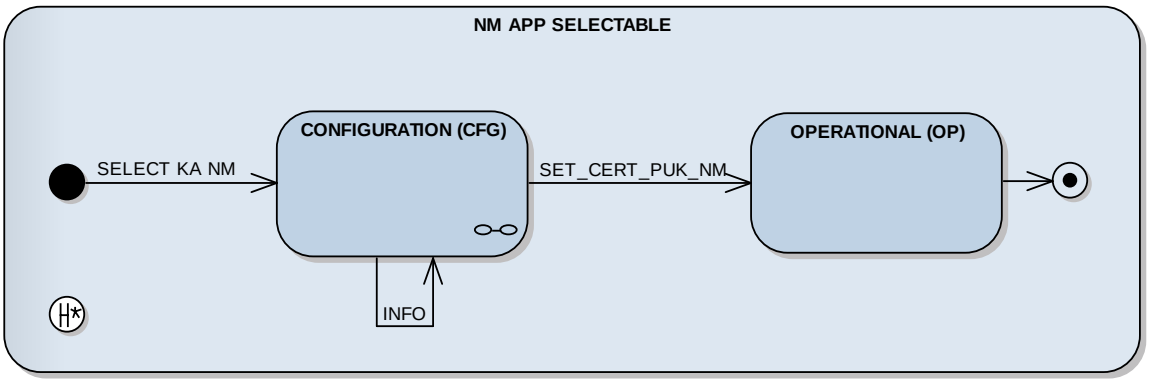

Abbildung 3 Lebenszyklus der NM Applikation

Der Zustand CFG umfasst alle für die Initialisierung des Nutzermediums relevanten Zustände. Nach abgeschlossener Initialisierung wird die NM Applikation irreversibel in den Zustand OP überführt. Anschließend sind bei selektierter NM Applikation ausschließlich die in [VDVNM] spezifizierten Kommandos und Prozesse verfügbar.

5.2. CONFIGURATION (CFG)

Eine nicht initialisierte NM Applikation befindet sich im Zustand CONFIGURATION (CFG). Im Zustand CFG werden die für den operativen Betrieb des Nutzermediums erforderlichen Daten eingebracht. Der Konfigurator prüft zuvor die Echtheit der NM Applikation mit dem PuK.Config.

Ein Nutzermedium befindet sich genau dann im Zustand CFG, wenn die Applikation mit der AID D2760001354B414E4D303100 selektierbar ist, die Root Daten eingebracht wurden und die Konfiguration noch nicht abgeschlossen wurde.

5.3. Applikationszustand CONFIGURATION (CFG)

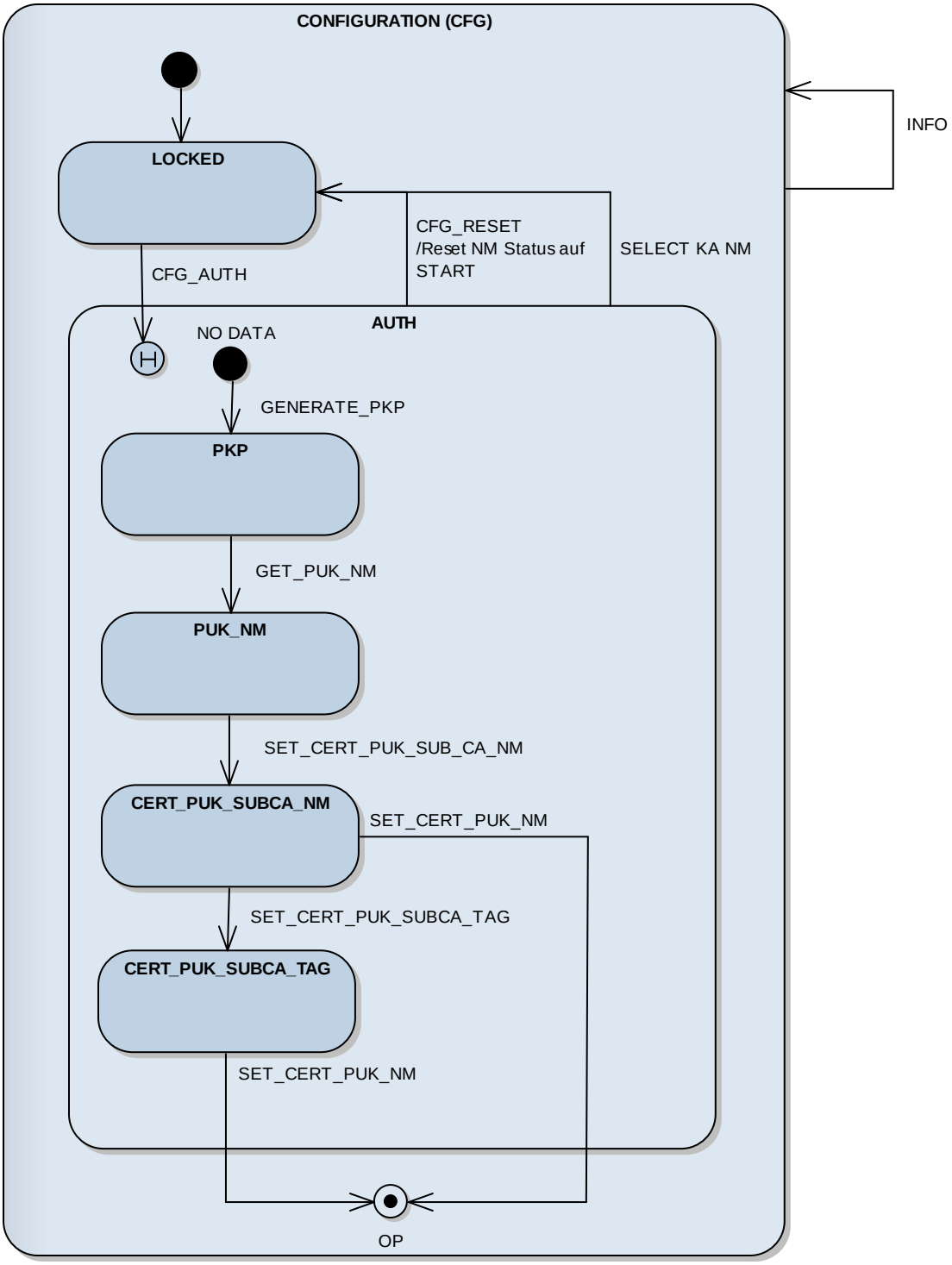

Abbildung 4 Subzustände CONFIGURATION

Zustand		ID	Beschreibung
LOCKED		'FF'	Die NM Applikation ist selektiert, der Konfigurator muss die Applikation authentifizieren.
AUTH		*Var.*	Die NM Applikation ist selektiert, das Initialisierungssystem hat die NM Applikation authentifiziert und der flüchtige Sitzungsschlüssel SKi sowie der Sendefolgenzähler SSC sind vereinbart worden. AUTH ist ein flüchtiger Sicherheitszustand.
	NO DATA	'00'	Ausgangszustand der Applikation nach dem Laden der Applikation oder nach erfolgreicher Ausführung des Kommandos CFG_RESET.
	PKP	'10'	Das Schlüsselpaar für das NM wurde erzeugt.
	PUK_NM	'20'	Der öffentliche Schlüssel des Nutzermediums wurde ausgelesen und die appInstanz_ID gesetzt.
	CERT_PUK_SUBCA_NM	'30'	Das Zertifikat einer VDV-KA Komponenten-CA für Nutzermedien wurde geladen.
	CERT_PUK_SUBCA_TAG	'31'	Das Zertifikat der VDV-KA Komponenten-CA für VDV-KA Tags wurde geladen.

Tabelle 7 Unterzustände CONFIGURATION

5.4. Authentifizierung der NM Applikation

Um zu gewährleisten, dass Konfiguratoren keine Wirkdaten in nicht authentische Nutzermedien einbringen, muss das Konfigurationssystem in der Lage sein, Nutzermedien vor der Konfiguration zu authentifizieren. Im Rahmen der Authentifizierung werden die temporären Sitzungsschlüssel SKi und SKc vereinbart, die zur Absicherung der Kommunikation dienen.

Für die Authentifizierung wird das asymmetrische Schlüsselpaar PkP.Config (PuK.Config, PrK.Config) genutzt, dass vom NM Hersteller oder App-Loader erzeugt wurde. Der öffentliche Teil PuK.Config wird Konfiguratoren zur Verfügung gestellt. Der private Teil PrK.Config wird bei der Überführung der NM Applikation in den Zustand CONFIGURATION (5.2) als Teil des PrK.Config-Datenobjekts (s. 3.1) in das NM eingebracht.

5.4.1. Authentifizierungsdaten

Bezeichung	Länge in Byte	Beschreibung
C.Config	16	Zufallszahl erzeugt durch den Konfigurator. Wird verschlüsselt an das Nutzermedium übergeben.
CD.Auth	256	Kryptogramm über die Daten SKi | C.Config erzeugt mit dem PuK.Config
MAC.Auth	16	MAC über die Zufallszahl C.Config im CBC Modus, erzeugt mit dem Schlüssel SKi.
SKi	32	Sitzungsschlüssel für das Verfahren AES-256 zur Integritätssicherung der Kommunikation. Wird vom Konfigurator erzeugt.
SSC	16	Sequenzzähler zur Absicherung der Kommunikation zwischen Konfigurator und Nutzermedium. Der Startwert des SSC ist C.Config.

Tabelle 8 Authentifizierungsdaten

5.4.2. Ablauf der Authentifizierung

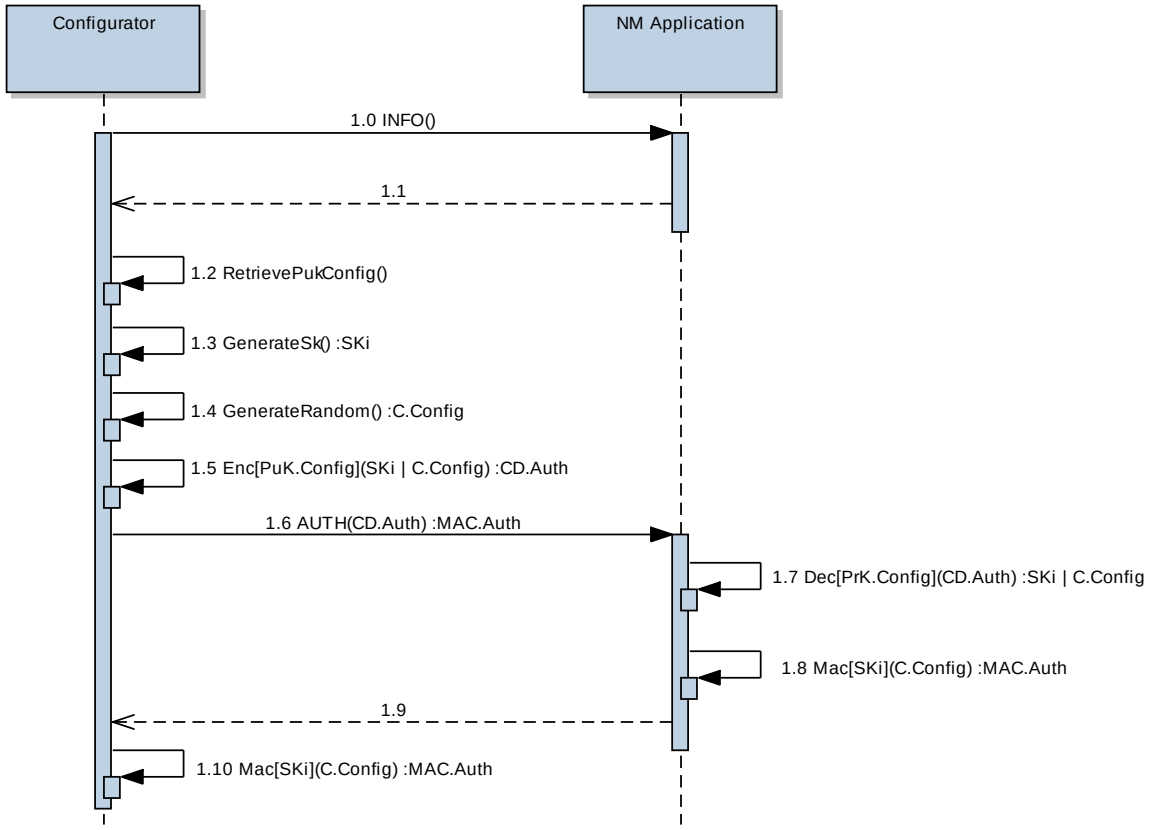

Abbildung 5 Authentisierungsablauf

Der Konfigurator ermittelt die in der NM Applikation hinterlegte Schlüssel ID des PrK.Config und sucht den entsprechenden öffentlichen Schlüssel PuK.Config im eigenen System.

Der Konfigurator erzeugt die Sitzungsschlüssel SKi und die Zufallszahl C.Config, verschlüsselt die verketteten Daten SKi | C.Config mit dem PuK.Config zum Kryptogramm CD.Auth und übermittelt CD.Auth an das Nutzermedium.

Das Nutzermedium entschlüsselt CD.Auth. War die Entschlüsselung erfolgreich, übernimmt das Nutzermedium die Sitzungsschlüssel SKi und setzt den SSC := C.Config.

Anschließend erhöht die NM Applikation den SSC um 1 und berechnet den CBC MAC (MAC.Auth) über die Zufallszahl C.Config mit dem Ski und AES-256. Der MAC.Auth wird in den Antwortdaten als Quittung ausgegeben.

Der Konfigurator setzt seinerseits den SSC := C.Config, inkrementiert den SSC um 1 und berechnet den MAC.Auth. Anschließend vergleicht der Konfigurator den berechneten MAC.Auth mit dem von der NM Applikation empfangenen MAC.Auth. Sind die Daten identisch, gilt die NM Applikation als authentisch und die NM Applikation befindet sich im flüchtigen Sicherheitszustand AUTH. Ansonsten bricht der Konfigurator den Vorgang mit einer Fehlermeldung ab und löscht C.Config und Ski.

5.4.3. Berechnung des Kryptogramms CD.Auth

Über die Daten SKi | C.Config berechnet der Konfigurator die kodierte Nachricht EM der Länge 255 Bytes (Bytelänge des Modulus - 1) gemäß EME-OAEP-Encoding (siehe Abschnitt A.2.3 der Spezifikation des VDV-SAM bzw. den Standard PKCS#1 v2.1) unter Verwendung des leeren Parameterstrings P.

Der Konfigurator verschlüsselt EM mit dem öffentlichen Schlüssel RSA-Schlüssel PuK.Config zum Kryptogramm CD.Auth.

5.5. Konfigurationsdaten

Im Rahmen der Konfiguration werden folgende Konfigurationsdaten in das Secure Element des KA NM eingebracht.

Datum	Beschreibung
Cert-PuK-Sub-CA-NM	Zertifikat über den öffentlichen Schlüssel der Sub-CA der VDV-PKI, die das Zertifikat für das NM ausstellen wird.
Cert-PuK-Sub-CA-TAG	Zertifikat über den öffentlichen Schlüssel der VDV-KA Komponenten-CA für VDV-KA Tags.
Cert-PuK-NM	Zertifikat über den öffentlichen Schlüssel des Mediums (Nutzermedium oder Tag).
appInstanz_ID	appInstanz_ID der Applikation

Tabelle 9 Konfigurationsdaten

5.6. Fehlercodes

Treten bei der Ausführung des Kommandos CONFIGURE Fehler auf, wird in den Antwortdaten einer der folgenden Fehlercodes ausgegeben.

Returncode (SW1SW2)	Beschreibung
'67 00'	Falsche Länge
'63 CF'	Überprüfung der Signatur fehlgeschlagen
'69 85'	Nutzungsbedingungen nicht erfüllt
'6A 81'	Funktion/Sub-Kommando nicht unterstützt
'6A 80'	Fehlerhafte Kommandodaten
'69 88'	SM Datenobjekte fehlen
'69 88'	SM Datenobjekte inkorrekt
'6A 86'	Unerlaubter Parameter P1 oder P2
'6A 88'	Daten nicht gefunden
'6D 00'	Unbekanntes Kommando (INS)
'6F XX'	Applikationsspezifischer Fehlercode

Tabelle 10 Returncodes

5.7. SELECT FILE Kommando

Das Kommando SELECT FILE verhält sich konform zur Spezifikation [VDVNM]. Bei der Parametrierung P2='0C' wird nur das Datenobjekt '80', das die Versionsangaben zur Applikation enthält, ausgegeben.
Da im Zustand CONFIGURATION keine Verzeichnisdaten vorhanden sind, fehlen die Verzeichnisdaten 'E1'

5.8. CONFIGURE Kommando

Mit dem Kommando CONFIGURE werden die für den Wirkbetrieb erforderlichen Daten in die NM Applikation eingebracht. P1 gibt bei der Kommandoausführung die auszuführende Operation (Sub-Kommando) an. Einige Sub-Kommandos liefern Antwortdaten, andere nicht. Deshalb wird das Le-Byte teilweise vorhanden sein, aber nicht immer. Wenn es vorhanden ist, wird hier immer der Wert ‘00‘ verwendet.

5.8.1. Kommandoaufbau

Kommando APDU

Position	Länge	Wert	Beschreibung
1	1	'XX'	CLA
2	1	'16'	INS
3	1	'XX'	P1 Sub-Kommando
4	1	'00'	P2
5	1	'XX'	Lc
6	var.	'XX .. XX'	Kommandodaten
7	1	'00'	Le (falls Antwortdaten erwartet)

Antwort APDU

Die Antwortdaten sind abhängig vom gewählten Sub-Kommando.

Position	Länge	Wert	Beschreibung
1	var.	'XX .. XX'	Antwortdaten
2	2	'90 00'	Returncode

5.8.2. Secure Messaging

Secure Messaging wird durch das CLA Byte 'XC' angezeigt. Die Kommandodaten des Kommandos mit Secure Messaging enthalten optional ein Klartext Datenobjekt und / oder ein Le-Datenobjekt gefolgt von einem MAC Datenobjekt:

	Klartextdatenobjekt (Kommandodaten vorhanden)		
	1	'81'	Tag für Klartextdatenobjekt
	1 bis 3	'XX' oder '81 XX' oder '82 XX XX'	Länge, L, der Kommandodaten aus der ungesicherten Command-APDU
	L	'XX .. XX'	Kommandodaten der ungesicherten Command-APDU
	Le-Datenobjekt (wenn Antwortdaten erwartet werden).		
	1	'97'	Tag für Le-Datenobjekt
	1	'01'	Länge
	1	'00'	Le-Feld der ungesicherten Kommando-APDU
	MAC-Datenobjekt		
	1	'8E'	Tag für MAC-Object
	1	'10'	Länge des MAC
	8	'XX .. XX'	MAC_SKi (kryptographische Checksumme)

Werden Antwortdaten zurückgegeben, enthalten diese ein Klartext-Datenobjekt sowie ein Status-Datenobjekt, gefolgt von einem MAC Datenobjekt.

	Klartextdatenobjekt (wenn Antwortdaten zurückgegeben werden)		
	1	'81'	Tag für Klartextdatenobjekt
	1 bis 3	'XX' oder '81 XX' oder '82 XX XX'	Länge, La, der Antwortdaten
	L	'XX .. XX'	Antwortdaten zur Kommando-APDU
	Status-Datenobjekt.		
	1	'99'	Tag für Status-Datenobjekt
	1	'02'	Länge
	1	'SW1 SW2'	Status-Bytes
	MAC-Datenobjekt		
	1	'8E'	Tag für MAC-Object
	1	'10'	Länge des MAC
	8	'XX .. XX'	MAC_SKi (kryptographische Checksumme)

5.8.2.1. MAC über die Kommandodaten

Der Message Authentication Code (MAC) wird mit dem AES Krypto-Algorithmus im CBC-Modus unter Verwendung des Schlüssels SKi berechnet.

Der MAC wird über den mit ISO-Padding auf Blocklänge erweiterten Kommandoheader gefolgt vom Klartextdatenobjekt und dem optionalen Le-Datenobjekt berechnet.

Entspricht die Länge der Eingangsdaten nicht einem Vielfachen der Blocklänge, werden sie mit ISO-Padding auf ein Vielfaches der Blocklänge erweitert.

Folgende Daten werden zusammengestellt:

```
HDR := CLA|INS|P1|P2|'80 00 00 00 00 00 00 00 00 00 00 00'
```

```
DATA := '81'|Lc|'XX .. XX'|['97 01'|Le|]'80 [00 ... 00]'
```

Vor der MAC Berechnung wird der SSC um 1 inkrementiert.

Anschließend wird der MAC über die verketteten Daten `HDR|DATA` unter Verwendung des Sitzungsschlüssels SK_i berechnet. Als ICV dient der zuvor berechnete SSC.

5.8.2.2. MAC über die Antwortdaten

Der MAC wird über das mit ISO-Padding auf Blocklänge erweiterte Klartextdatenobjekt (sofern vorhanden) und das Status-Datenobjekt berechnet.

```
DATA := ['81'|La|'XX .. XX'|]'99 02'|SW1 SW2|'80 [00 ... 00]'.
```

Vor der MAC Berechnung wird der SSC um 1 inkrementiert.

Anschließend wird der MAC über die Daten DATA unter Verwendung des Sitzungsschlüssels SK_i berechnet. Als ICV dient der zuvor berechnete SSC.

5.8.3. Command Chaining

Übersteigt die Länge der an die NM-Applikation zu übermittelnden Daten 255 Bytes, muss das Kommando im verketteten Modus (s. [NM]) ausgeführt werden.

5.8.4. Formale Prüfungen

Das Format der Kommando-APDU wird auf Korrektheit geprüft, d.h. die Kodierung von Lc (wenn vorhanden) und Le (wenn vorhanden) wird gemäß ISO/IEC 7816-4 geprüft. Im Fehlerfall wird die Verarbeitung des Kommandos abgebrochen und der Returncode '67 00' ausgegeben. Der Parameter INS wird auf Zulässigkeit geprüft, d.h. ob das Kommando bekannt ist. Im Fehlerfall wird die Verarbeitung des Kommandos abgebrochen und der Returncode '6D 00' ausgegeben.

Es wird geprüft, ob der Wert von CLA zulässig ist und die Kombination mit dem Wert von INS der Spezifikation entspricht. Im Fehlerfall wird die Verarbeitung des Kommandos abgebrochen und der Returncode '6E 00' ausgegeben.

Die Parameter P1/P2/Lc/Le werden im Kontext des durch CLA und INS definierten Kommandos geprüft. Hierbei werden zulässige Bereiche oder Konstanten überprüft. Im Fehlerfall wird die Verarbeitung des Kommandos abgebrochen und der Returncode '6A 86' bei unzulässigen Werten für P1/P2 oder '67 00' bei unzulässigen Werten für Lc oder Le ausgegeben.

Wenn im CLA-Byte Secure Messaging (SM) signalisiert wird, erfolgt die Prüfung des Datenobjekts für SM Datenobjekte. Im Fehlerfall wird die Verarbeitung des Kommandos abgebrochen und der Returncode '69 87' oder '69 88' oder '6A 80' oder '6A 88' ausgegeben.

Alle weiteren Parameter und Datenformate werden im Rahmen der kommandospezifischen Verarbeitung überprüft. Die Reihenfolge der beschriebenen Überprüfungen kann je nach Implementierung variieren.

5.8.5. CONFIGURE Sub-Kommandos

Sub-Kommando (P1)		*Ausgangszustand*	CONFIGURATION (CFG)						*OP*
			LOCKED	NO DATA	PKP	PUK_NM	CERT_PUK_SUBCA_NM	CERT_PUK_SUBCA_TAG	
			'FF'	'00'	'10'	'20'	'30'	'31'	
INFO	'01'		'FF'	'00'	'10'	'20'	'30'	'31'	-
AUTH	'10'		***H***[1)]	-	-		-		-
GENERATE_PKP	'02'		-	'10'	-	-	-		-
GET_PUK_NM	'03'		-	-	'20'	-	-	-	-
SET_CERT_PUK_SUBCA_NM	'04'		-	-	-	'30'		-	-
SET_CERT_PUK_SUBCA_TAG	'14'		-	-	-	-	'31'	-	
SET_CERT_PUK_NM	'05'		-	-	-	-	***OP***[2)]	***OP***[2)]	-
RESET	'FF'		-	'FF'	'FF'	'FF'	'FF'	'FF'	-
			Folgezustand						

1) ***H*** letzter gültiger, nicht-flüchtig gespeicherter Zustand.

2) ***OP*** Zustand OPERATIONAL, Konfiguration abgeschlossen

5.8.6. INFO

INFO gibt den aktuellen Zustand der Initialisierung und zusätzliche Informationen zur Ermittlung des PuK.Config aus.

5.8.6.1. Kommando- und Antwortnachricht

Kommando APDU

Position	Länge	Wert	Beschreibung
1	1	'00'	CLA
2	1	'16'	INS
3	1	'00'	P1 = '00' für INFO
4	1	'00'	P2
5	1	'00'	Le

Antwort APDU

Nach erfolgreicher Ausführung gibt das Kommando die Antwortdaten gefolgt vom Returncode '90 00' aus.

Position	Länge	Wert	Beschreibung
1	21 oder 26	'XX .. XX'	Antwortdaten
2	2	'90 00'	Returncode

Antwortdaten

Position	Länge	Wert	Beschreibung
1	1	'XX'	Aktueller Zustand (s.5.3)
2	8	'XX … XX'	CAR der Root-CA [VDVSAM]
3	6	'XX … XX'	Schlüssel-ID des PrK.Config
4	6	'XX … XX'	Wurde der PuK.NM bereits ausgelesen, wird die beim Auslesen eingebrachte appInstanz_ID zurückgegeben.

5.8.7. AUTH

Mit dem Sub-Kommando AUTH wird die NM Applikation in den flüchtigen Sicherheitszustand AUTH (s. 5.3) überführt. Dafür muss der Aufrufer über den Schlüssel PuK.Config verfügen und das Kryptogramm CD.Auth erzeugen (5.4.3), das in den Kommandodaten übergeben wird. Gleichzeitig wird der für die Konfiguration erforderliche Sitzungsschlüssel SKi sowie der SSC festgelegt. Die NM-Applikation belegt ihre Authentizität mit der Quittung MAC.Auth, die in den Antwortdaten ausgegeben wird.

5.8.7.1. Kommando- und Antwortnachricht

Kommando APDU

Position	Länge	Wert	Beschreibung
1	1	'X0'	CLA mit Command Chaining
2	1	'16'	INS
3	1	'10'	P1 = '10' für AUTH
4	1	'00'	P2
5	1	var.	Lc
6	var.	'XX .. XX'	Kommandodaten
7	1	'00'	Le

Antwort APDU

Nach erfolgreicher Ausführung gibt das Kommando die Antwortdaten gefolgt vom Returncode '90 00' aus.

Position	Länge	Wert	Beschreibung
1	16	'XX .. XX'	MAC.Auth
2	2	'90 00'	Returncode

5.8.7.2. Ablaufbeschreibung

Nach dem Empfang des letzten Teilkommandos entschlüsselt die NM Applikation das empfangene Kryptogramm CD.Auth mit dem PrK.Config. Schlägt die Entschlüsselung fehl, wird das Kommando mit dem Returncode '6A 80' abgebrochen.

Die Applikation speichert den im Klartext vorliegenden Schlüssel SKi flüchtig für die weitere Verwendung und setzt den SSC := C.Config. Anschließend erhöht die Applikation den SSC um 1 und berechnet den MAC.Auth, indem der CBC-MAC (ICV := SSC) mit dem SKi über C.Config gebildet wird. Das Ergebnis der Berechnung wird in den Antwortdaten ausgegeben.

5.8.8. GENERATE_PKP

Das Sub-Kommando GENERATE_PKP erzeugt das RSA Schlüsselpaar der NM-Applikation. Das Schlüsselpaar wird in der sicheren Umgebung der NM Applikation ("on-Card") generiert. Nach erfolgreicher Ausführung des Kommandos kann der öffentliche Schlüssel PuK.NM ausgelesen werden.

5.8.8.1. Kommando- und Antwortnachricht

Kommando APDU

Position	Länge	Wert	Beschreibung
1	1	'0C'	CLA
2	1	'16'	INS
3	1	'02'	P1 = '02' für GENERATE_PKP
4	1	'00'	P2
5	1	'10'	Lc
6	1	'8E'	Tag für MAC Datenobjekt
7	1	'10'	Länge
8	16	'XX .. XX'	MAC

Antwort APDU

Im Erfolgsfall gibt die Applikation den Returncode '90 00' aus.

Position	Länge	Wert	Beschreibung
1	2	'90 00'	Returncode

5.8.8.2. Ablaufbeschreibung

Die Applikation prüft die Kommandodaten formal (s. 5.8.4).

Die Applikation prüft, ob das Sub-Kommando im aktuellen Zustand zugelassen ist. Ist dies nicht der Fall, wird die Ausführung mit dem Returncode '69 85' abgebrochen.

Die Applikation erhöht den erhöht den intern, flüchtig gespeicherten SSC-Wert um Eins und prüft den MAC über die Kommandodaten mit ICV = SSC. Schlägt die Prüfung fehl, wird das Kommando mit dem Returncode '69 88' abgebrochen.

Liegt das Schlüsselpaar der Applikation bereits vor, wird der Returncode '90 00' ausgegeben. Ansonsten generiert die Applikation das Schlüsselpaar PkP.NM, speichert die Daten nicht-flüchtig und gibt im Anschluss den Returncode '90 00' aus.

Nach erfolgreicher Ausführung des Sub-Kommandos befindet sich die Applikation im nicht-flüchtigen Zustand PKP.

5.8.9. GET_PUK_NM

Das Sub-Kommando GET_PUK_NM dient zum gesicherten Auslesen des öffentlichen Schlüssels PuK.NM der Applikation. Gleichzeitig wird die appInstanz_ID der Applikation gesetzt, die zur eindeutigen Identifikation der Applikation im Verlauf der Initialisierung Verwendung finden kann.

5.8.9.1. Kommando- und Antwortnachricht

Kommando APDU

Position	Länge	Wert	Beschreibung
1	1	'0C'	CLA
2	1	'16'	INS
3	1	'03'	P1 = '02' für GET_PUK_NM
4	1	'00'	P2
5	1	'1A'	Lc
6	1	'81'	Klartext Datenobjekt appInstanz_ID
7	1	'06'	Länge
8	6	'XX .. XX'	appInstanz_ID
9	1	'8E'	Tag für MAC-Datenobjekt
10	1	'10'	Länge
11	16	'XX .. XX'	MAC
12	1	'00'	Le

Antwort APDU

Im Erfolgsfall gibt die Applikation folgende Antwortdaten gefolgt vom Returncode '90 00' aus.

Position	Länge	Wert	Beschreibung
1	1	'81'	Klartext Datenobjekt Schlüsseldaten
2	2	'81 8D'	Länge
3	2	'7F 49'	Tag Datenobjekt PuK.NM
4	2	'81 89'	Länge
5	1	'81'	Tag Modulus
6	2	'81 80'	Länge Modulus
7	128	'XX .. XX'	Modulus
8	1	'82'	Tag öffentlicher Exponent
9	1	'04'	Länge öffentlicher Exponent
10	4	'XX .. XX'	Öffentlicher Exponent
11	1	'8E'	Tag für MAC Datenobjekt
12	1	'10'	Länge
13	16	'XX .. XX'	MAC

5.8.9.2. Ablaufbeschreibung

Die Applikation prüft die Kommandodaten formal (s. 5.8.4).

Die Applikation prüft, ob das Sub-Kommando im aktuellen Zustand zugelassen ist. Ist dies nicht der Fall, wird die Ausführung mit dem Returncode '69 85' abgebrochen.

Die Applikation erhöht den intern, flüchtig gespeicherten SSC-Wert um Eins und prüft den MAC über die Kommandodaten mit ICV = SSC. Schlägt die Prüfung fehl, wird das Kommando mit dem Returncode '69 88' abgebrochen.

Die Applikation übernimmt die appInstanz_ID aus dem Klartext Datenobjekt und speichert diese nicht-flüchtig.

Anschließend stellt die Applikation den PuK.NM im Klartext Datenobjekt der Antwortdaten bereit, erhöht den SSC-Wert um Eins, berechnet den MAC über das Klartext Datenobjekt (s. 5.8.2.2) mit ICV = SSC und gibt die Antwortdaten gefolgt vom Returncode '90 00' aus.

Nach erfolgreicher Ausführung des Sub-Kommandos befindet sich die Applikation im nicht-flüchtigen Zustand PUK_NM.

5.8.10. SET_CERT_PUK_SUBCA_NM

Mit dem Sub-Kommando SET_CERT_PUK_SUBCA_NM wird das Zertifikat Cert-PuK-Sub-CA-NM über den öffentlichen Schlüssel der Sub-CA, die das Zertifikat über den öffentlichen Schlüssel des Nutzermediums ausstellt, verifiziert und eingebracht. Das Zertifikat ist ein CV-Zertifikat mit Signatur gemäß PKCS#1_v1.5.

Ferner wird der öffentliche Schlüssel aus dem Zertifikat so abgelegt, dass dieser im OPERATIONAL Zustand durch das Kommando VERIFY CERTIFICATE der NM-Applikation direkt in einer einstufigen Verifikation eines SAM-Authentisierungszertifikats verwendet werden kann.

Das Kommando verwendet Command-Chaining und Secure Messaging.

5.8.10.1. Kommando- und Antwortnachricht

Kommando APDU

Position	Länge	Wert	Beschreibung
1	1	'XC'	CLA '0C' für einfachen oder letzten Kommandoaufruf im verketteten Modus '1C' für den ersten bis vorletzten Kommandoaufruf im verketteten Modus
2	1	'16'	INS
3	1	'04'	P1 = '04' für SET_CERT_PUK_SUBCA_NM
4	1	'00'	P2
5	1	'XX'	Lc
6	1	'81'	Klartext Datenobjekt
7	1 od. 2	['81'] 'XX'	Länge des Klartext Datenobjekts
8	*var.*	'XX .. XX'	Zertifikatsdaten
9	1	'8E'	Tag für MAC-Datenobjekt
10	1	'10'	Länge des MAC
11	16	'XX .. XX'	MAC

Antwort APDU

Im Erfolgsfall gibt die Applikation den Returncode '90 00' aus.

Position	Länge	Wert	Beschreibung
1	2	'90 00'	Returncode

5.8.10.2. Ablaufbeschreibung

Die Applikation prüft die Kommandodaten formal (s. 5.8.4).

Die Applikation prüft, ob das Sub-Kommando im aktuellen Zustand zugelassen ist. Ist dies nicht der Fall, wird die Ausführung mit dem Returncode '69 85' abgebrochen.

Die Applikation erhöht den intern, flüchtig gespeicherten SSC-Wert um Eins und prüft den MAC über die Kommandodaten mit ICV=SSC. Schlägt die Prüfung fehl, wird das Kommando mit dem Returncode '69 88' abgebrochen.

Wurde ein Teilkommando empfangen, übernimmt die Applikation die im Klartext Datenobjekt enthaltenen Daten für die spätere Prüfung und gibt den Returncode '90 00' aus.

Wurde das letzte Teilkommando empfangen, prüft die Applikation das Zertifikat. Hierzu wird die Signatur mit dem in den Config-Daten empfangenen PuK.Root überprüft. Schlägt die Prüfung fehl, wird das Kommando mit dem Returncode '63 CF' abgebrochen.

War die Prüfung des Zertifikats erfolgreich, wird das Zertifikat nicht-flüchtig gespeichert und der Returncode '90 00' ausgegeben.

Nach erfolgreicher Ausführung des Sub-Kommandos befindet sich die Applikation im nicht-flüchtigen Zustand CERT_PUK_SUBCA_NM.

5.8.11. SET_CERT_PUK_SUBCA_TAG

Mit dem Sub-Kommando SET_CERT_PUK_SUBCA_TAG wird das Zertifikat Cert-PuK-Sub-CA-Tag über den öffentlichen Schlüssel der Sub-CA, die das Zertifikat über den öffentlichen Schlüssel der VDV-KA Tags ausstellt, verifiziert und eingebracht. Das Zertifikat ist ein CV-Zertifikat mit Signatur gemäß PKCS#1_v1.5.

Ferner wird der öffentliche Schlüssel aus dem Zertifikat so abgelegt, dass dieser im OPERATIONAL Zustand durch das Kommando Verify Certificate der NM-Applikation direkt in einer einstufigen Verifikation eines Tag-Zertifikats verwendet werden kann.

Das Kommando verwendet Command-Chaining und Secure Messaging.

5.8.11.1. Kommando- und Antwortnachricht

Kommando APDU

Position	Länge	Wert	Beschreibung
1	1	'XC'	CLA '0C' für einfachen oder letzten Kommandoaufruf im verketteten Modus '1C' für den ersten bis vorletzten Kommandoaufruf im verketteten Modus
2	1	'16'	INS
3	1	'14'	P1 = '14' für SET_CERT_PUK_SUBCA_TAG
4	1	'00'	P2
5	1	'XX'	Lc
6	1	'81'	Klartext Datenobjekt
7	1 od. 2	['81'] 'XX'	Länge des Klartext Datenobjekts
8	*var.*	'XX .. XX'	Zertifikat Daten
9	1	'8E'	Tag für MAC-Datenobjekt
10	1	'10'	Länge des MAC
11	16	'XX .. XX'	MAC

Antwort APDU

Im Erfolgsfall gibt die Applikation den Returncode '90 00' aus.

Position	Länge	Wert	Beschreibung
1	2	'90 00'	Returncode

5.8.11.2. Ablaufbeschreibung

Die Applikation prüft die Kommandodaten formal (s. 5.8.4).

Die Applikation prüft, ob das Sub-Kommando im aktuellen Zustand zugelassen ist. Ist dies nicht der Fall, wird die Ausführung mit dem Returncode '69 85' abgebrochen.

Die Applikation erhöht den intern, flüchtig gespeicherten SSC-Wert um Eins und prüft den MAC über die Kommandodaten mit ICV = SCC. Schlägt die Prüfung fehl, wird das Kommando mit dem Returncode '69 88' abgebrochen.

Wurde ein Teilkommando empfangen, übernimmt die Applikation die im Klartext Datenobjekt enthaltenen Daten für die spätere Prüfung und gibt den Returncode '90 00' aus.

Wurde das letzte Teilkommando empfangen, prüft die Applikation das Zertifikat. Hierzu wir die Signatur mit dem in den Config-Daten empfangenen PuK.Root überprüft. Schlägt die Prüfung fehl, wird das Kommando mit dem Returncode '63 CF' abgebrochen.

War die Prüfung des Zertifikats erfolgreich, wird das Zertifikat nicht-flüchtig gespeichert und der Returncode '90 00' ausgegeben.

Nach erfolgreicher Ausführung des Sub-Kommandos befindet sich die Applikation im nicht-flüchtigen Zustand CERT_PUK_SUBCA_TAG.

5.8.12. SET_CERT_PUK_NM

Mit dem Sub-Kommando SET_CERT_PUK_NM wird das Zertifikat Cert-PuK-NM über den öffentlichen Schlüssel PuK.NM (bzw. das Zertifikat Cert-PuK-Tag über den öffentlichen Schlüssel PuK.Tag im Falle einer Tag-Applikation) verifiziert und in die Applikation eingebracht. Nach erfolgreicher Einbringung wird die Applikation in den nicht-flüchtigen Zustand OPERATIONAL (s. 5.1) überführt.

5.8.12.1. Kommando- und Antwortnachricht

Kommando APDU

Position	Länge	Wert	Beschreibung
1	1	'XC'	CLA '0C' für einfachen oder letzten Kommandoaufruf im verketteten Modus '1C' für den ersten bis vorletzten Kommandoaufruf im verketteten Modus
2	1	'16'	INS
3	1	'05'	P1 = '05' für SET_CERT_PUK_NM
4	1	'00'	P2
5	3	'XX'	Lc
6	1	'81'	Tag für Klartext Datenobjekt
7	3	'81 XX'	Länge des Klartext Datenobjekts
8	376	'XX .. XX'	CV-Zertifikat des NM gemäß [VDVSAM]
9	1	'8E'	Tag für MAC-Datenobjekt
10	1	'10'	Länge des MAC
11	16	'XX .. XX'	MAC

Antwort APDU

Im Erfolgsfall gibt die Applikation den Returncode '90 00' aus.

Position	Länge	Wert	Beschreibung
1	2	'90 00'	Returncode

5.8.12.2. Ablaufbeschreibung

Die Applikation prüft die Kommandodaten formal (s. 5.8.4).

Die Applikation prüft, ob das Sub-Kommando im aktuellen Zustand zugelassen ist. Ist dies nicht der Fall, wird die Ausführung mit dem Returncode '69 85' abgebrochen.

Die Applikation erhöht den intern, flüchtig gespeicherten SSC-Wert um Eins und prüft den MAC über die Kommandodaten mit ICV=SSC. Schlägt die Prüfung fehl, wird das Kommando mit dem Returncode '69 88' abgebrochen.

Wurde ein Teilkommando empfangen, übernimmt die Applikation die im Klartext Datenobjekt enthaltenen Daten für die spätere Prüfung und gibt den Returncode '90 00' aus.

Wurde das letzte Teilkommando empfangen, prüft die Applikation das empfangene Zertifikat.

Hierzu wir die Signatur mit dem zuvor eingebrachten öffentlichen Schlüssel der Sub-CA geprüft. Schlägt die Prüfung fehl, wird das Kommando mit dem Returncode '63 CF' abgebrochen.

Die in den Zertifikatsdaten enthaltene appInstanz_ID wird mit der beim Auslesen des PuK.NM eingebrachten (s. 5.8.9) verglichen. Sind die Daten unterschiedlich, wird das Kommando mit dem Returncode '6A 80' abgebrochen.

War die Prüfung des Zertifikats erfolgreich, wird das Zertifikat nicht-flüchtig gespeichert und der Returncode '90 00' ausgegeben.

Nach erfolgreicher Ausführung des Sub-Kommandos wird die Applikation nicht-flüchtig in den Zustand OPERATIONAL (s. 5.1) überführt.

5.8.13. RESET

Das Sub-Kommando RESET setzt den Konfigurationszustand auf NO DATA (s. 5.3) zurück. Alle im Rahmen der Konfiguration eingebrachten oder generierten Daten werden gelöscht. Nach erfolgreicher Ausführung des Kommandos, wird der flüchtige Sicherheitszustand auf LOCKED gesetzt und alle temporären Schlüssel gelöscht.

5.8.13.1. Kommando- und Antwortnachricht

Kommando APDU

Position	Länge	Wert	Beschreibung
1	1	'0C'	CLA
2	1	'16'	INS
3	1	'FF'	P1 = 'FF' für RESET
4	1	'00'	P2
5	1	'12'	Lc
6	1	'8E'	Tag für MAC-Datenobjekt
7	1	'10'	Länge
8	16	'XX .. XX'	MAC

Antwort APDU

Im Erfolgsfall gibt die Applikation den Returncode '90 00' aus.

Position	Länge	Wert	Beschreibung
1	2	'90 00'	Returncode

5.8.13.2. Ablaufbeschreibung

Die Applikation prüft die Kommandodaten formal (s. 5.8.4).

Die Applikation prüft, ob das Sub-Kommando im aktuellen Zustand zugelassen ist. Ist dies nicht der Fall, wird die Ausführung mit dem Returncode '69 85' abgebrochen.

Die Applikation erhöht den intern, flüchtig gespeicherten SSC-Wert und prüft den MAC über die Kommandodaten mit ICV=SSC. Schlägt die Prüfung fehl, wird das Kommando mit dem Returncode '69 88' abgebrochen.

Alle im Rahmen der Konfiguration eingebrachten Daten werden gelöscht. Bei erneuter Durchführung der Konfiguration müssen die Daten erneut eingebracht werden. Die Root Daten werden nicht gelöscht.

Der nicht-flüchtige Subzustand NO DATA (s. 5.3) wird gesetzt und der flüchtige Sicherheitszustand mit LOCKED überschrieben. Die Applikation stellt sicher, dass SKi und SSC ungültig sind.

6 Anhang

6.1. Glossar

Abkürzung	Bedeutung
NM	Nutzermedium
VDV	Verband Deutscher Verkehrsunternehmen
KA	Kernapplikation

6.2. Referenzen

[SPEC_NM] VDV-Kernapplikation, Spezifikation Nutzermedium, V1.106, Mai 2008

[SPEC_SAM] VDV-Kernapplikation, Spezifikation SAM, V1.106, Mai 2008