TTA Standard

정보통신단체표준(국문표준)

TTAK.KO-12.0240 제정일: 2014년 04월 10일

모바일 후불 교통카드

Mobile Deferred Payment Traffic Card

TTA 한국정보통신기술협회
Telecommunications Technology Association

정보통신단체표준(국문표준)

TTAK.KO-12.0240 제정일: 2014년 04월 10일

모바일 후불 교통카드

Mobile Deferred Payment Traffic Card

TTA 한국정보통신기술협회
Telecommunications Technology Association

서 문

1. 표준의 목적

본 표준에서는 모바일 교통 결제에 대한 후불 결제가 가능하도록 애플릿 규격을 정의한다. 본 표준은 고객이 지불 결제 관련 서비스 선택권을 가질 수 있게 하여 사용 편의성을 제공한다. 이를 통해 모바일 교통 결제서비스가 활성화되고 표준화된 지불 결제 방식을 갖추게 되어 다양한 교통 인프라에서 상호 호환성을 확보할 수 있을 것이다.

2. 주요 내용 요약

본 표준은 콤비 USIM 또는 NFC USIM에 탑재되는 후불 교통 서비스를 위한 애플릿과 모바일 선불 및 모바일 후불 교통 결제 수단을 관리하는 애플릿에 대한 규격을 정의한다.

본 표준에서 정의하는 애플릿은 콤비 USIM 및 NFC USIM에 탑재되어 교통 결제 서비스를 수행할 수 있도록 설계하였으며, 통합 결제 서비스에 참여하는 구성 요소(단말, 정산 시스템 등)간 상호 호환성이 보장되도록 설계하였다.

3. 표준 적용 산업 분야 및 산업에 미치는 영향

본 표준은 모바일 교통 결제 수단을 선불에서 후불로 확대함으로써, 모바일 교통 결제 서비스의 활성화를 기대할 수 있다. 또한 모바일 선불 및 모바일 후불 결제 수단을 통합 관리하는 애플릿을 통해 사용자의 교통 결제 수단에 대한 선택권을 제공할 수 있으며, 이를 통해 교통 인프라에서는 표준화된 지불 결제 방식을 수행할 수 있어 서비스 호환 및 인프라의 확대를 기대할 수 있다.

4. 참조 표준(권고)

4.1. 국외 표준(권고)

- ISO/IEC 7810, 'Identification cards - Physical characteristics'
- ISO/IEC 7816, 'Identification cards - Integrated circuit cards'
- ISO/IEC 10373, 'Identification cards - Test methods'
- ISO/IEC 14443, 'Identification cards- Contactless integrated circuit(s) cards - Proximity cards'

4.2. 국내 표준

- 국토교통부 고시 제2013- 465호, '교통카드 관련 장비의 전국호환성 인증 요령'
- KS X 6924 - 1, '선불IC카드 - KS X 6923 대응 사용자 카드 - 제1부 : 물리적 특성 및 기본 구조', 2009.12.
- KS X 6924 - 2, '선불IC카드 - KS X 6923 대응 사용자 카드 - 제2부 : 명령어 및 프로토콜', 2009.12.
- KS X 6924 - 3, '선불IC카드 - KS X 6923 대응 사용자 카드 - 제3부 : 암호 알고리즘', 2009.12.
- KS X 6924 - 4, '선불IC카드 - KS X 6923 대응 사용자 카드 - 제4부 : 적합성 시험', 2009.12.
- KS X ISO/IEC 7816－5, 'ID 카드－접촉식 IC 카드－제5부 : 응용 식별자의 번호 체계 및 등록 절차', 2008.12.

5. 참조 표준(권고)과의 비교

5.1. 참조 표준(권고)과의 관련성

본 표준은 'KS X 6924'의 물리적 특성 및 기본구조, 명령어 및 프로토콜, 암호 알고리즘 등이 동일하며, 이동통신사 신용카드사에서 OTA를 통한 발급 기능과 다수의 instance를 관리할 수 있는 기능이 추가되었다는 점이 다르다.

5.2. 참조한 표준(권고)과 본 표준의 비교표

TTAK.KO-12.0240	KS X 6924	비교
물리사항	물리사항	동일
파일시스템	파일시스템	수정
명령어	명령어	동일
보안 메커니즘	보안 메커니즘	동일
DF-전자지갑 서비스 파일	-	추가
DF-전자지갑 서비스 명령어	-	추가
DF-전자지갑 서비스 프로토콜	-	추가
Config - DF 파일시스템	-	추가

6. 지식 재산권 관련 사항

본 표준의 '지식 재산권 확약서' 제출 현황은 TTA 웹사이트에서 확인할 수 있다.

※본 표준을 이용하는 자는 이용함에 있어 지식 재산권이 포함되어 있을 수 있으므로, 확인 후 이용한다.

※본 표준과 관련하여 접수된 확약서 이외에도 지식 재산권이 존재할 수 있다.

7. 시험 인증 관련 사항

7.1. 시험 인증 대상 여부

- 해당 사항 없음.

7.2. 시험 표준 제정 현황

- 해당 사항 없음.

8. 표준의 이력 정보

8.1. 표준의 이력

판수	제정·개정일	제정·개정 내역
제 1 판	2014.04.10.	제정 TTAK.KO-12.0240

8.2. 주요 개정 사항

- 해당 사항 없음.

Preface

1. Purpose of Standard

The purpose of this standard is to offer payment service options and convenience of use by developing a post payment system for mobile transportation fees, compatible across various transportation infrastructures, and to revitalize mobile transportation payment and the standardized payment system.

2. Summary of Contents

This defines the standard of applets for post transportation service payment built into Combi USIM or NFC USIM and applets that manage advanced and post transportation service payment.

These applets are embedded into Combi USIM or NFC USIM to initiate transportation service payment and offer inter-compatibility among the components of integral payment services (terminal equipment, adjustment systems).

3. Applicable Fields of Industry and its Effect

Through this standard, the revitalization of mobile transportation payment service is expected through the expansion of advanced to post payment ranges. Additionally, the transportation payment system options can be provided through integral applets, which manage advanced and post mobile payment systems. This results in service compatibility and expanded infrastructure in transportation through a standard payment system.

4. Reference Standards(Recommendations)

4.1. International Standards(Recommendations)

- ISO/IEC 7810, Identification cards - Physical characteristics
- ISO/IEC 7816, Identification cards - Integrated circuit cards
- ISO/IEC 10373, Identification cards - Test methods
- ISO/IEC 14443, Identification cards- Contactless integrated circuit(s) cards - Proximity cards

4.2. Domestic Standards

- Certify national compatibility of the traffic card related equipment
- Contactless pre-paid/post pay IC card－User card－Part 1 : Physical characteristics and basic structure, 2009.12.
- Contactless pre-paid/post pay IC card－User card－Part 2 : Commands and protocols, 2009.12.
- Contactless pre-paid/post pay IC card－User card－Part 3 : Cryptogram, 2009.12.
- Contactless pre-paid/post-pay IC card－User card－ Part 4 : Conformity test, 2009.12.
- Identification cards－Integrated circuit(s) cards with contacts－Part 5 : Numbering system and registration procedure for application identifiers, 2008.12.

5. Relationship to Reference Standards(Recommendations)

5.1. Relationship of Reference Standards(Recommendations)

This Standard provides the physical properties and basic structure of KS X 6924 with same instruction, protocol and encryption algorithm. Such functions as releasing through OTA by a credit card of a carrier and managing multiple instances were added.

5.2. Differences between Reference Standard(Recommendation) and this Standard

TTAx.xx-xx.xxxx	KS X 6924	Remarks
Physical Attributes	Physical Attributes	Equivalent
File System	File System	Revised
Command	Command	Equivalent
Security Mechanism	Security Mechanism	Equivalent
DF-E-Wallet Service File	-	Added
DF-E-Wallet Service Command	-	Added
DF-E-Wallet Service Protocol	-	Added
Config - DF File System	-	Added

6. Statement of Intellectual Property Rights

IPRs related to the present document may have been declared to TTA. The information pertaining to these IPRs, if any, is available on the TTA Website.

No guarantee can be given as to the existence of other IPRs not referenced on the TTA website.

And, please make sure to check before applying the standard.

7. Statement of Testing and Certification

7.1. Object of Testing and Certification

- None

7.2. Standards of Testing and Certification

- None

8. History of Standard

8.1. Change History

Edition	Issued date	Outline
The 1st edition	2013.04.10.	Established TTAK.KO-12.0240

8.2. Revisions

- None

목 차

Contents

모바일 후불 교통카드
(Mobile Deferred Payment Traffic Card)

1. 개요

본 표준에서는 모바일 교통 결제에 대한 후불 결제가 가능하도록 애플릿 규격을 정의한다. 본 표준은 고객이 지불 결제 관련 서비스 선택권을 가질 수 있게 하여 사용 편의성을 제공한다. 이를 통해 모바일 교통 결제서비스가 활성화되고 표준화된 지불 결제 방식을 갖추게 되어 다양한 교통 인프라에서 상호 호환성을 확보를 목적으로 한다.

본 표준은 콤비 USIM 또는 NFC USIM에 탑재되는 후불 교통 서비스를 위한 애플릿과 모바일 선불 및 모바일 후불 교통 결제 수단을 관리하는 애플릿에 대한 규격을 정의한다.

본 표준에서 정의하는 애플릿은 콤비 USIM 및 NFC USIM에 탑재되어 교통 결제 서비스를 수행할 수 있도록 설계하였으며, 통합결제 서비스에 참여하는 구성 요소(단말, 정산 시스템 등) 간 상호 호환성이 보장되도록 설계하였다.

2. 표준의 구성 및 범위

본 표준은 모바일 후불 교통카드(이하 카드)에 대한 물리적 규격과 파일 시스템, 기본 명령어, 표준 거래 명령어에 대한 일반적인 사항에 대하여 정의하고 있다.

3. 참조 표준(권고)

3.1. 국외 표준(권고)

- ISO/IEC 7810, 'Identification cards - Physical characteristics'
- ISO/IEC 7816, 'Identification cards - Integrated circuit cards'
- ISO/IEC 10373, 'Identification cards - Test methods'
- ISO/IEC 14443, 'Identification cards- Contactless integrated circuit(s) cards - Proximity cards'

3.2. 국내 표준

- 국토교통부 고시 제2013- 465호, '교통카드 관련 장비의 전국호환성 인증 요령'
- KS X 6924 - 1, '선불IC카드 - KS X 6923 대응 사용자 카드 - 제1부 : 물리적 특성 및 기본 구조', 2009.12.
- KS X 6924 - 2, '선불IC카드 - KS X 6923 대응 사용자 카드 - 제2부 : 명령어 및 프로토콜', 2009.12.

- KS X 6924 - 3, '선불IC카드 - KS X 6923 대응 사용자 카드 - 제3부 : 암호 알고리즘', 2009.12.
- KS X 6924 - 4, '선불IC카드 - KS X 6923 대응 사용자 카드 - 제4부 : 적합성 시험', 2009.12.
- KS X ISO/IEC 7816-5, 'ID 카드-접촉식 IC 카드-제5부 : 응용 식별자의 번호 체계 및 등록 절차', 2008.12.

4. 용어 정의 및 약어

4.1. 용어 정의

4.1.1. 카드

모바일 후불 교통카드

4.1.2. Type A/B

'ISO/IEC 14443'에서 정의하고 있는 프로토콜로 통신할 수 있는 카드의 통신 프로토콜에 의한 분류

4.2. 약어

AFI	Application Family Identifier
ATQA	Answer To Request, Type A
ATQB	Answer To Request, Type B
CLn	Cascade Level n, Type A
DF	Dedicate File
ID	Identifier
MAC	Message Authentication Code
REAB	Request command, Type B
REQA	Request command, Type A
SAK	Select Acknowledge, Type A
SFI	Short EF Identifier
SID	Short Key File Identifier
UID	Unique Identifier, Type A

5. 물리 사항

카드는 'ISO/IEC 14443'을 준수하고 있으며, Type A, B 등의 제한을 가지지 않는다.

5.1. 카드의 특성

- 'ISO/IEC 7816'을 준수하는 접촉 통신 프로토콜 지원(선택)
- 'ISO/IEC 14443' Part 1,2,3,4를 준수하는 비접촉 통신 프로토콜 지원
- MF(Master File)를 포함한 2-Level 계층 구조 지원(Multi-Application)
- 높은 보안 기능(Dynamic 인증, Access Condition 관리, MAC 인증을 통한 Secure Messaging)
- 'ISO/IEC 7810'에서 ID 카드의 모든 형태를 위해 명시된 물리적 특성을 준수

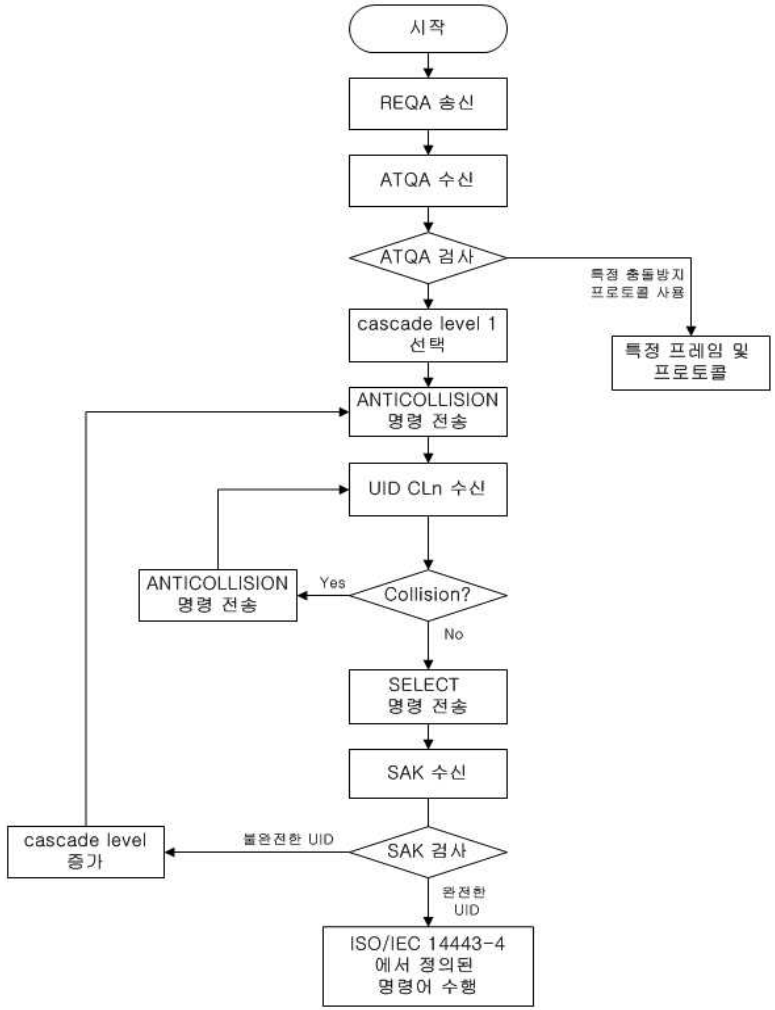
5.2. 비접촉 특성

'ISO/IEC 14443' Part 3에 기술되어 있는 초기화 및 충돌 방지 절차에 대한 사항을 준수한다.

5.2.1. Type A 비충돌식

여러 장의 Type A 카드 중 하나의 카드를 선택하는 비충돌식 절차는 다음과 같다.

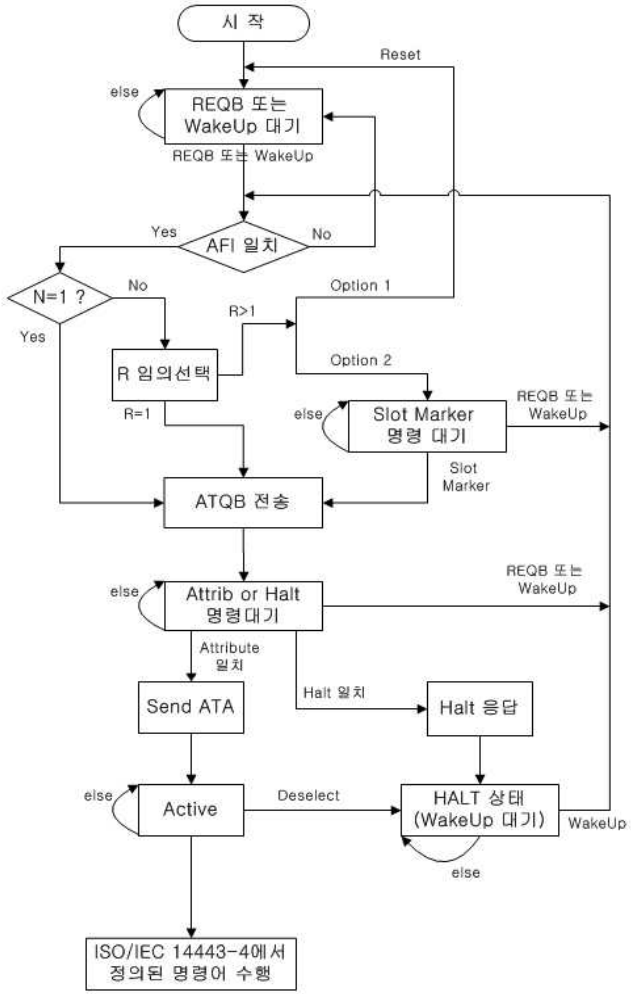
정보통신단체표준(국문표준)



(그림 5-1) 비충돌식 절차

5.2.2. Type B 비충돌식

여러 장의 Type B 카드 중 하나의 카드를 선택하는 비충돌식 절차는 다음과 같다.



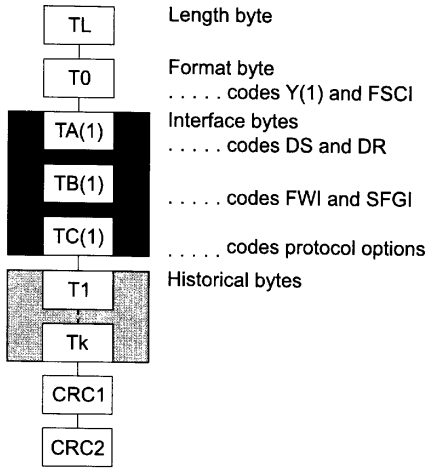
(그림 5-2) 비충돌식 절차

5.2.3. 카드 구분

5.2.3.1. Type A

Type A의 경우 카드 구분은 ATS의 Historical Bytes를 이용한다.

Structure of ATS(Answer To Select)



(그림 5-3) ATS 구조

5.2.3.2. Type B

Type B의 카드는 ATQB 중 Application Data 영역을 이용한다.

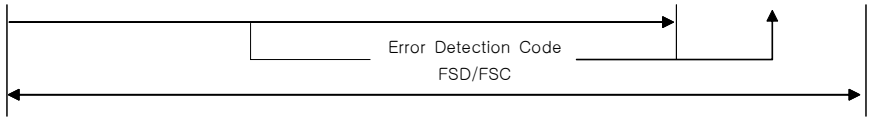
<표 5-1> ATQB Response Format

1 Byte	2~5 Byte	6~9 Byte	10~12 Byte	13~14 Byte
'50'	PUPI	Application Data	Protocol Info	CRC_B
MSB				LSB

5.2.4. 데이터 전송 프로토콜 Block format

<표 5-2> 데이터 전송 프로토콜 Block format

Prologue Field			Information Field	Epilogue Field
PCB	[CID]	[NAD]	[INF]	EDC
1Byte	1Byte	1Byte		2Bytes



Prologue field

■ Protocol Control Byte(PCB)

데이터 전달을 제어하기 위해 요구되는 정보를 전달하는 데 사용된다. 프로토콜은 블록의 3가지 기본적인 타입을 정의한다.

- Information block(I-block)은 응용 계층에서 실제 명령을 전달하는 데 사용된다.
- A receive ready block(R-block)은 ACK 또는 NACK의 정보를 전달하는 데 사용되며 R-block은 INF field를 포함하지 않는다.
- A supervisory block(S-block)은 인터페이스 장치와 카드 사이에서 제어 정보를 교환하는 데 사용된다.

■ Card Identifier field(CID)

CID는 특정한 카드를 식별하는 데 사용된다.

■ Node Address field(NAD)

NAD는 논리적으로 다른 연결을 생성하고 어드레싱하기 위해 남겨둔 값이다.

■ Information field(INF)

INF field는 선택 사항이다. INF field는 I-block의 application data와 S-block의 not-application data와 status information을 수용한다. I-block의 application data는 'ISO/IEC 7816'에서 정의되어 있는 APDU(Application Protocol Data Unit)를 의미한다.

■ Epilogue field

전송 블록의 에러 감지 코드(EDC, Error Detection Code)를 포함한다.

6. 파일 시스템

이 카드는 'ISO/IEC 7816' Part4에서 규정한 파일 구조를 따른다.

6.1. 파일 계층

이 카드의 필수 파일 구조는 아래와 같이 계층적 구조를 가지며, 그 Level은 2이다. 각 DF는 EF를 가질 수 있다.

6.2. 파일 타입

이 카드에서 정의되는 파일 타입은 다음과 같다.

6.2.1. Application Dedicated File(ADF)

Application Dedicated File은 특정 애플리케이션에 대한 EF들의 기능적인 그룹으로 이루어지며 DF Name을 제외하고는 별도의 파일 내용을 포함하지 않고 그 자체로 애플리케이션의 내용을 상징한다.

6.2.2. Elementary 파일(EF)

카드 내에서 사용되는 각 응용 데이터를 저장하는 파일로서 단말기에 의해 읽거나 쓸 수 있는 파일이다. 이 파일은 Transparent 및 Record 구조를 사용하여 구성될 수 있다.

6.2.3. Working EF(WEF)

이 파일은 카드 소지자에 대한 정보 및 전자화폐 정보 등 거래를 수행하는 데 필요한 각종 정보를 저장하며, 'ISO/IEC 7816' Part4에서 규정하는 Transparent 또는 Record 구조를 가진다.

6.2.4. Purse EF(PEF)

거래에 사용되는 잔액 정보 및 거래 내역에 대한 처리를 위한 파일로 Record 구조를 가지는 특수한 용도의 파일이다.

6.2.5. Internal EF(IEF)

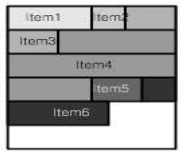
카드 내에서 보안 메커니즘을 운영하기 위해 사용되는 정보를 저장하고 있는 파일이다.

6.3. 파일 구조

카드에서 사용되는 파일들의 구조는 다음과 같다.

6.3.1. Transparent File

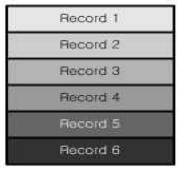
연속된 바이트로 구성되며 Offset(오프셋)을 사용하여 파일 내에 기록된 데이터를 참조한다. 일반적으로 거래에 직접 관계되지 않는 정보를 저장하는 데 사용된다. Read Binary, Update Binary 명령을 사용하여 데이터를 읽거나 기록한다.



(그림 6-1) Transparent File

6.3.2. Linear Fixed Record File

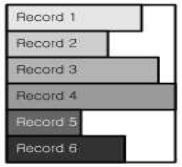
모든 레코드가 동일한 크기를 가지고 있는 레코드 파일로서 레코드 번호는 생성 순서에 따라 연속적으로 할당된다. 즉, 첫 번째 레코드가 가장 먼저 만들어진 레코드가 되며 파일 내의 레코드 개수는 카드 발급 시 결정된다. Read Record, Update Record, Append Record 명령을 통해 레코드를 읽거나 기록한다.



(그림 6-2) Linear Fixed Record File

6.3.3. Linear Variable Record File

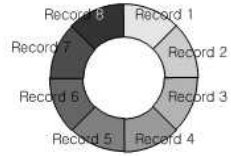
Linear Fixed Record File과 비슷한 구조이지만 모든 레코드의 크기가 동일하지는 않은 구조를 가지고 있다. 레코드 번호는 생성 순서에 따라 연속적으로 할당되지만 각 레코드의 크기는 틀리다. 파일의 크기는 카드 발급 시 결정되지만, 레코드 개수는 결정되지 않는다. Read Record, Update Record, Append Record 명령을 통해 레코드를 읽거나 기록한다.



(그림 6-3) Linear Variable Record File

6.3.4. Linear Cyclic Record File

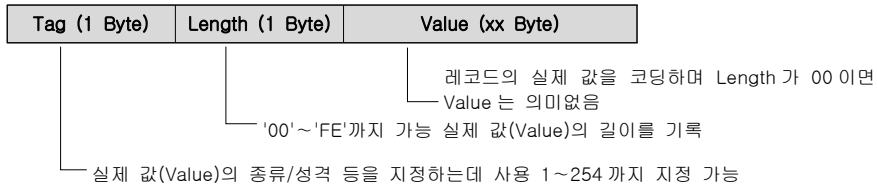
Linear Fixed Record File과 비슷하지만 링 구조로 구성되어 있다. 각 레코드 크기는 동일하며 레코드 번호는 역순에 따라 연속적으로 할당된다. 즉, 첫 번째 레코드가 가장 최근에 업데이트된 레코드이며 각 기록 프로시저에서 가장 오래된 레코드를 덮어쓰게 된다. Purse File은 이 구조를 따른다. Read Record, Update Record, Append Record 명령을 통해 레코드를 읽거나 기록할 수 있다.



(그림 6-3) Linear Cyclic Record File

■ 레코드 구조

이 규격에서는 'ISO/IEC 7816' Part4에 규정된 레코드 구조 중 Simple TLV 구조를 사용한다.



(그림 6-4) 레코드 구조

6.4. 파일 참조 방법

이 규격은 'ISO/IEC 7816' Part4에 규정된 파일 참조 방법 중 다음 3가지의 방식과 Key File의 경우를 지원한다.

6.4.1. File Identifier에 의한 참조

파일 생성 시에 지정한 File ID로 파일을 참조할 수 있다. 이 경우 선택할 수 있는 범위는 현재 속해있는 DF 내의 파일 및 현재 DF의 Parent DF, Parent DF 내의 하위 파일까지 이다. Select File 명령을 이용하여 수행된다.

6.4.2. Short EF Identifier(SFI)에 의한 참조

Select File 명령으로 선택된 파일 이외에 기타 다른 파일의 선택이 필요한 명령에서 사용되는 파일 참조 방법으로 File ID 2 바이트 중 하위 5 비트(1~30, 0x01~0x1E)만을

이용하며, 파일 선택이 필요한 명령의 파라미터 또는 데이터로 지정된다. SFI에 의한 참조가 성공하면, Current File이 변경된다.

6.4.3. DF Name 에 의한 참조(DF)

DF 선택 시에만 적용되는 방식으로 DF Name은 카드 내에서 유일하므로 DF Name이 지정되어 있는 DF 라면 카드 내의 어떤 DF 라도 선택이 가능하다. 일반적으로 DF 생성 시 애플리케이션에 따라 DF Name을 부여한다.

6.4.4. Short Key File Identifier(SID)에 의한 참조

Key File의 경우, ID로 Key Reference Number를 사용한다. SID를 지정하지 않은 경우, Key File에서 순차적인 번호체계를 자동적으로 갖는다.

7. 보안 메커니즘

교통카드는 보안 매체로서 아래와 같은 다섯 가지 보안 요구 사항이 만족되어야 하며, 이를 위해 암호 알고리즘과 비밀 키에 기반한 다양한 보안 메커니즘이 지원된다. 각 메커니즘은 카드의 파일 내에 안전하게 저장되는 Key를 기반으로 암호 알고리즘의 수행을 통해 운영된다.

Security Domain에 대한 키 인증, 암호화 및 MAC 생성은 'Global Platform 2.2.1' 규격에 준수한다.

<표 7-1>보안 메커니즘

보안 요구 사항	지원되는 보안 메커니즘
기밀성(Confidentiality)	암호 알고리즘을 이용한 데이터의 암복호화
인증(Authentication)	단말기 인증 - External Authentication 카드 인증 - Internal Authentication
접근제어(Access Control)	파일 및 명령 별 AC(Access Condition)
무결성(Integrity)	Secure Messaging(메시지 송수신 시 암복호화 및 MAC 사용) 데이터 무결성 - 암호 알고리즘을 이용한 암복호화 + MAC 메시지 무결성 보장 - MAC
부인방지(Repudiation)	Secure Messaging(메시지 송수신 시 MAC 사용)

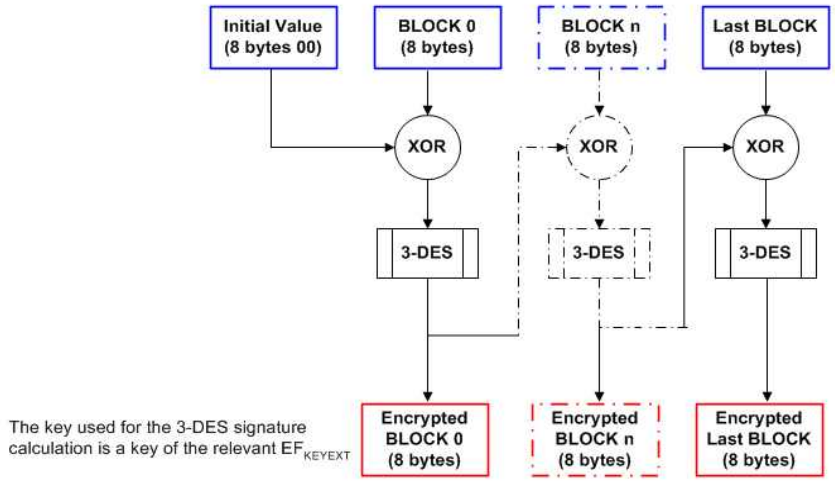
7.1. 암호 알고리즘

교통 카드 칩 운영 시스템에서 지원되는 암호 알고리즘은 3-DES 및 SEED이며, 사용되는 키 파일 내의 키 정보에 따라 알고리즘 수행 시마다 특정 암호 알고리즘을 지정하여 수행할 수 있는 구조로 되어 있다. 각 암호 알고리즘은 CBC 모드로 운영되며, 입력 데이터가 정해진 블록 크기에 맞지 않는 경우 지정된 데이터를 패딩(Padding)하여 암호 알고리즘을 수행한다.

7.1.1. 3-DES 알고리즘

3-DES는 DES 알고리즘의 빈도를 높이기 위해 키의 길이를 DES의 2배(16바이트)로 하고, DES 알고리즘을 3회 반복(암호-복호-암호)하여 사용하는 방법이다. 입력 데이터의 크기는 8 바이트이며 사용되는 키의 길이는 16 바이트이다. 3-DES 알고리즘의 ID는 10이며 본 표준 내에서의 표현 방법은 다음과 같다.

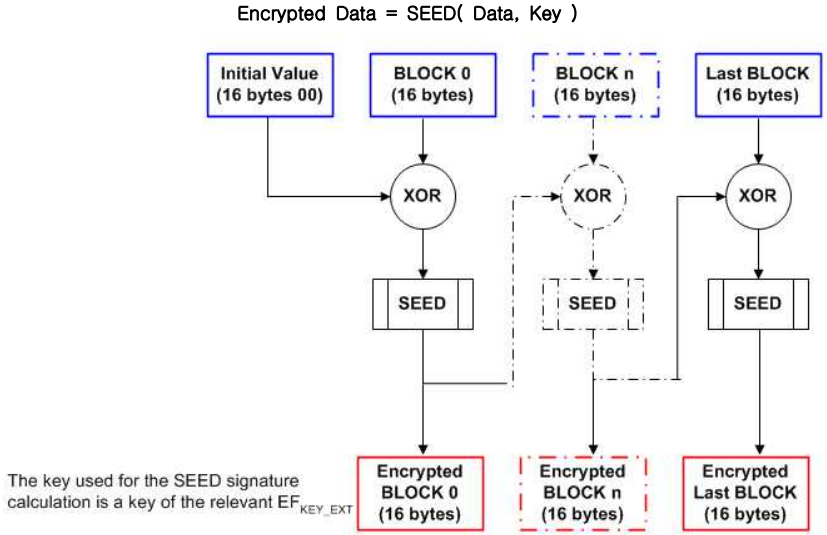
Encrypted Data = 3-DES(Data, Key)



(그림 7-1) 3-DES 알고리즘

7.1.2. SEED 알고리즘

SEED는 국내에서 개발한 128 비트 표준 암호 알고리즘('TTAS.KO-12.0004')으로 데이터와 키의 길이가 모두 16 바이트(128 비트)이다. SEED 알고리즘의 ID는 00이며 본 규격 내에서의 표현 방법은 다음과 같다.

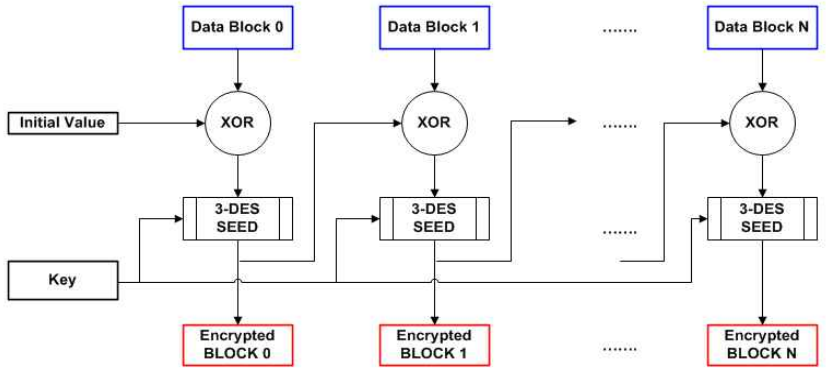


(그림 7-2) SEED 알고리즘

7.1.3. 암호 알고리즘 수행 모드

3-DES, SEED를 이용하여 한 블록(8 바이트 또는 16 바이트) 이상의 데이터를 암호화할 때, 기본적으로 CBC(Cipher Block Chaining)모드를 적용한다. 초기 입력값(IV)는 SEED의 경우 16 바이트, 3-DES의 경우 8 바이트의 '0000…¦00'±을 사용한다.

CBC 모드는 다음과 같이 동작한다.



(그림 7-3) CBC 모드

7.1.4. 패딩 규칙

(1) MAC/서명 생성 시

입력 데이터의 마지막 블록이 알고리즘의 블록 크기(SEED: 16 바이트, 3-DES: 8 바이트)에 맞지 않는 경우 마지막 블록에 "80 00 …¦ 00"±을 블록크기가 될 때까지 채운 뒤 암호 알고리즘을 수행한다. 입력 데이터의 마지막 블록이 알고리즘의 블록 크기(SEED-16바이트, 3-DES -8바이트)와 일치할 경우 패딩 블록(SEED: 16 바이트, 3-DES: 8바이트)을 추가하지 않는다.

(2) 암호화 시

입력 데이터의 마지막 블록이 알고리즘의 블록 크기(SEED: 16 바이트, 3-DES: 8바이트)에 맞지 않는 경우 마지막 블록에 "80 00 …¦ 00"±을 블록크기가 될 때까지 채운 뒤 암호 알고리즘을 수행하고, 입력 데이터의 마지막 블록이 알고리즘의 블록 크기(SEED-16바이트, 3-DES -8바이트)와 일치할 경우 패딩 블록(SEED:16 바이트, 3-DES:8바이트)을 추가한다.

주) 데이터가 키로만 구성된 경우, 데이터의 마지막 블록의 크기가 지정 알고리즘의 블

록 크기와 일치하더라도 이에 따른 추가 패딩을 수행하지 않는다.

7.2. Access Condition

Access Condition(AC)은 파일의 생성 시 결정되며 각 파일에 사용되는 각 명령의 동작에 따라 별개로 첨부된다.

파일에 대해 특정 명령을 실행할 경우 파일을 액세스하기 전 해당 명령의 동작에 관계된 Access Condition을 만족하여야 한다. Access Condition은 보안 레벨에 따라 16단계로 나누며 특정 Access Condition의 수행 시 Access Condition 레벨과 연결된 키를 사용하여 수행된다.

7.2.1. Access Condition 할당

Access Condition은 총 2바이트의 크기를 가지며 각각 nibble 단위로 구분되어 총 4개의 AC가 할당된다. 해당 nibble에는 각 권한에 해당하는 Access Condition Level 값이 기록된다.

각 Access Condition은 레벨이 MAC 또는 AUTH인 경우 AC를 만족시키기 위해 EFAUTH_KEY파일 내의 키가 필요하게 된다. 이때 필요한 키의 번호는 nibble 단위로 지정된다.

Access Condition Level이 (E-)MAC 또는 AUTH가 아닌 경우 지정된 키 번호는 무시된다.

아래는 명령의 Operation에 따라 분류된 Access Condition과 그 용도이다.

<표 7-1> Access Condition

AC	File Operation	Access Condition 용도
AC0	Write	초기 쓰기, 데이터 추가 권한
AC1	Read	읽기 권한
AC2	Update	갱신 권한
AC3	Block / Unblock	Block/Unblock 권한

7.2.2. Access Condition Level

교통 카드 내에서 사용되는 Access Condition Level은 다음과 같으며 AC의 값으로 사용된다.

<표 7-2> Access Condition Level

AC 레벨	AC 이름	Access Condition 설명
0	ALW or FREE	특정 제한이 없이 그 수행이항상 가능하다.
1	RFU	미지정
2	MAC	정확한 Cryptogram(MAC)으로 인증된 명령만 수행이 가능하다.
3	E-MAC	수행되는 명령의 데이터가 암호화되고 명령이 MAC으로 인증이 되어야만 명령 수행이 가능하다.
4	AUTH	정확한 비밀키를 이용하여 External Authentication을 수행하고 이를 이용하여 인증된 경우에만 접근이 가능하다.
5	SD_AUTH	Security Domain으로 키 인증이 되어야만 명령 수행이 가능하다.
6~14	RFU	미지정
15	NEV	수행이 항상 불가능하다.

7.3. Authentication

교통 카드는 단말기 내에 장착된 SAM과 서로에 대해 인증을 수행하기 위한 여러 가지 절차가 필요하다. 이 절차는 카드 및 SAM 내의 Random Number Generator에서 생성된 난수와 이 난수를 이용하여 특정 알고리즘을 수행한 결과값을 서로가 교환함으로써 상대방이 특정 키를 알고 있는지 확인하는 과정이다. 카드와 SAM의 상호 인증은 교환되는 데이터 원본의 보안을 보장하며 인증 수행의 결과로 상대방이 특정 비밀 키를 소유하고 있다는 것이 입증되면 키의 노출이 없이 명령이 수행된다. 교통 카드 칩 운영 시스템은 다음과 같은 인증 메커니즘을 제공한다.

7.3.1. External Authentication(단말기 인증)

External Authentication은 카드에 대한 SAM의 신뢰성을 입증한다. 즉, 카드가 SAM이 정당한지를 판단하는 것이다.

이 과정은 카드가 생성한 난수를 Get Challenge 명령을 통하여 SAM이 수신하고 이 난수를 EFAUTH_KEY에 있는 지정된 Key를 사용하여 정확하게 암호화하여 Cryptogram을 생성한 뒤 이 Cryptogram을External Authentication 명령을 통하여 카드로 전송함으로써 수행된다. 카드는 SAM이 제공한 Cryptogram을 확인하여 인증 여부를 결정한다.

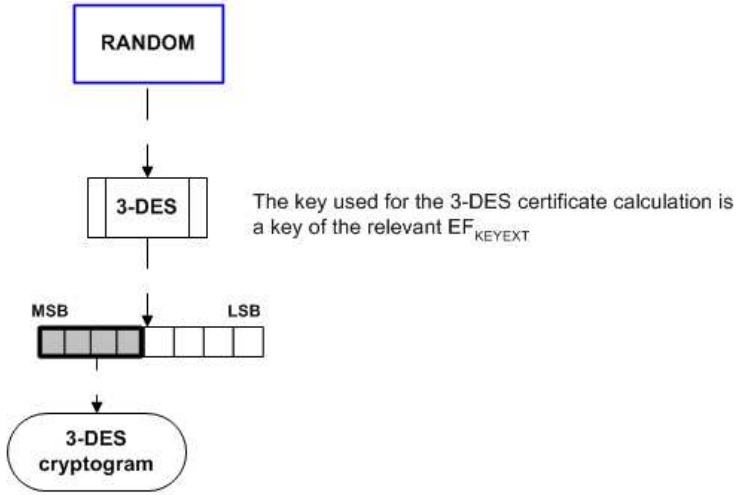
Get Challenge 명령 다음에는 즉시 External Authentication이 이루어져야 하며 사용되는 키 번호는 External Authentication 명령의 파라미터로 지정된다. 암호문 확인이 올바르게 이루어지면 해당 인증 조건이 만족된다.

확인이 실패하면 해당 Key의 시도 카운트가 감소한다. 시도 카운트가 0이 되면 해당 Key를 더 이상 사용할 수 없다. Put Key 명령어를 통해 카운트를 Reset 할 수 있다.

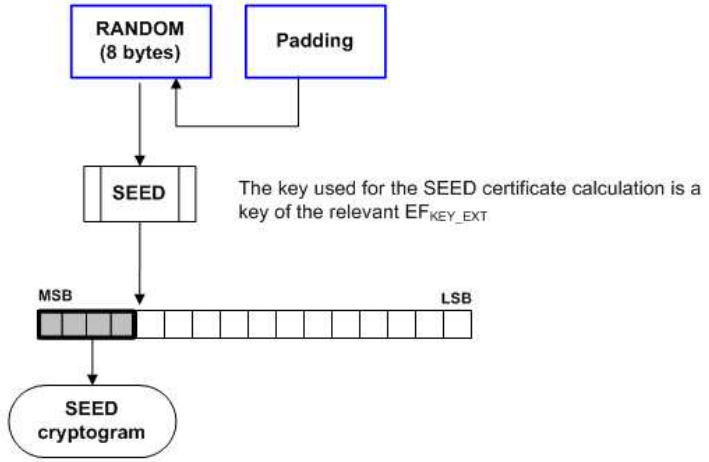
External Authentication 절차는 다음과 같다.

(그림 7-4) External Authentication 절차

< Cryptogram 계산 방법 >



(그림 7-5) 3-DES Mode



(그림 7-6) SEED Mode

7.4. Secure Messaging

카드에 명령이 전송되거나 카드로부터 응답이 발생될 때 명령/응답 메시지의 무결성 및 기밀성 보장을 위해 칩 운영체제는 다음과 같은 2가지 형태의 Secure Messaging 메커니즘을 제공한다.

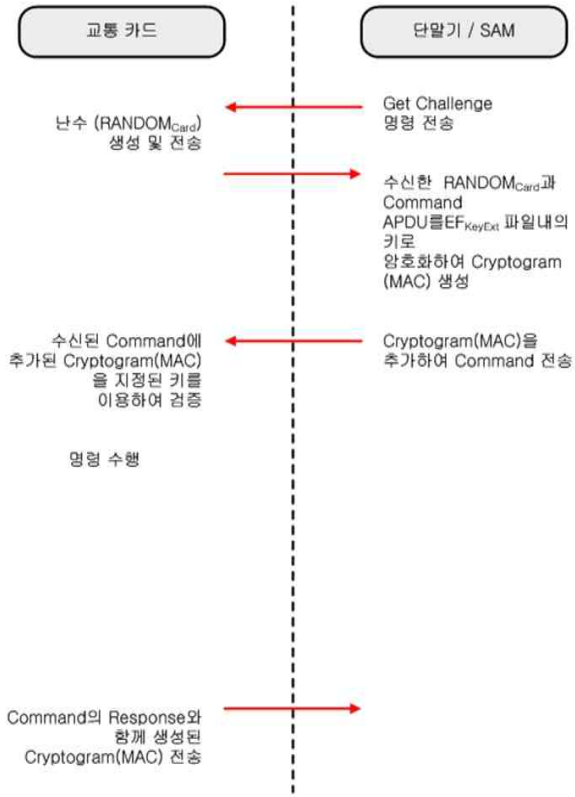
MAC Mode: 전송되는 명령/응답 메시지에 Cryptogram 추가
E-MAC Mode: 명령/응답 메시지 내의 데이터 암호화 + 전송되는 명령/응답 메시지에 Cryptogram 추가

이 메커니즘은 각각 MAC, E-MAC Access Condition Level과 연동되며 명령을 통해 접근되는 파일의 Access Condition에 따라 MAC 또는 E-MAC 모드의 암호화 및 인증 절차를 수행한다.

7.4.1. MAC Mode

전송되는 명령에 인증값(Cryptogram)을 추가하거나 명령의 수행 후 전송되는 응답에 인증값을 추가한다. 카드는 이 명령에 첨부된 인증값을 확인하여 단말기 쪽에서 보낸 명령이 유효함을 검증한다.
이 모드는 특정 파일의 Access Condition에 지정되어 사용된다. 명령의 Cryptogram 인증 시 사용되는 키는 EFAUTH_KEY에 존재한다. 이전에 실행된 명령이 Replay되는 것을 막기 위해 Cryptogram 계산에는 난수가 포함된다. 명령 교환을 방지하기 위해 명령 헤더의 데이터는 Cryptogram 계산 시 첫 번째 입력 데이터 블록에 삽입된다.

MAC 모드에서의 명령 수행 과정은 다음과 같다.



(그림 7-7) MAC Mode

■ 명령 전송 시 Cryptogram 계산 방법(단말기/SAM)

Cryptogram = 3-DES(Random + Command Header + Command Data + Padding, Key)

Cryptogram = SEED(Random + Command Header + Command Data + Padding, Key)

- Key는 EFKEY_EXT 파일 내의 키를 사용(Access Condition 내에 지정된 키 번호를 사용)

<--Random-->	<----Command Header---->					<-Command Data->	<-Padding->
Random	CLA	INS	P1	P2	P3	Data Field	Padding Block
8Byte	1Byte	1Byte	1Byte	1Byte	1Byte	Data Bytes	X Bytes

(그림 7-8) Cryptogram 계산 시 암호 알고리즘의 입력 데이터

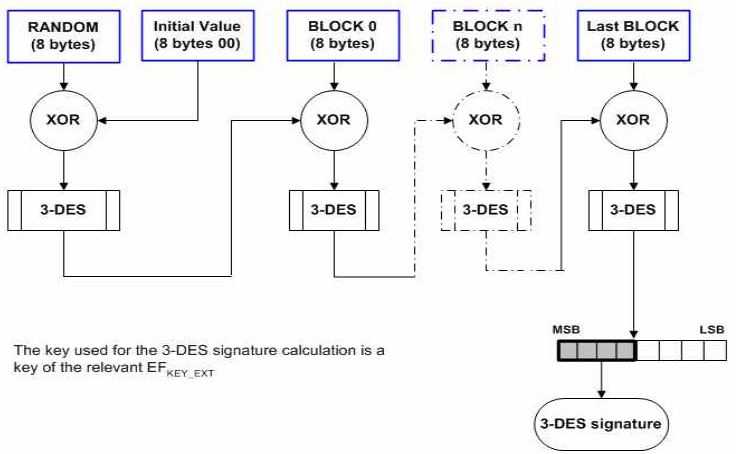
명령 전송 시 P3(Lc)는 Data length + 4(Cryptogram 크기)이며 응답 수신 시 필요한 P3(Le)는 수신할 응답의 길이 + 4(Cryptogram 크기) 이다.

Padding Block의 길이 X는 사용 암호 알고리즘에 따라 마지막 블록이 SEED 일 경우 16-Byte, 3-DES일 경우 8-Byte가 되도록 Padding Byte(80 00 …| 00)를 채워준다.

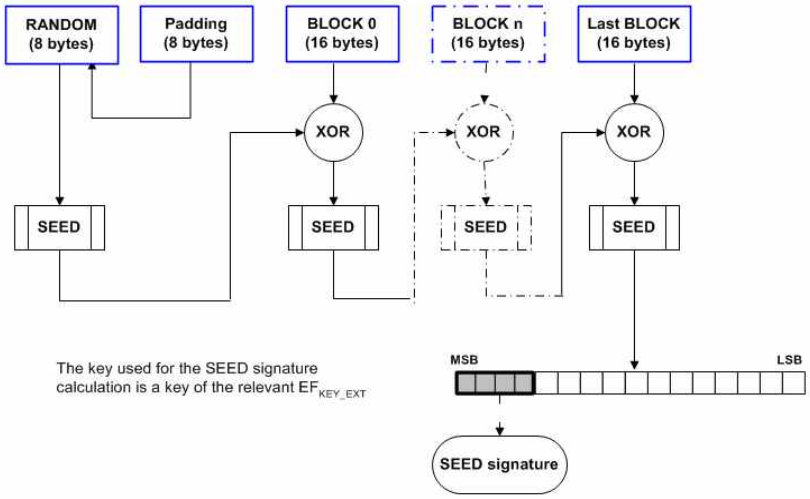
암호 알고리즘 수행 결과의 최상위 4 Bytes만 Cryptogram(MAC)으로 사용된다.

■ 응답 전송 시 Cryptogram 계산 방법(교통카드)

V1과의 호환성을 위하여 응답 MAC 값으로 '0x00 00 00 00'을 사용한다.



(그림 7-9) 3-DES Mode

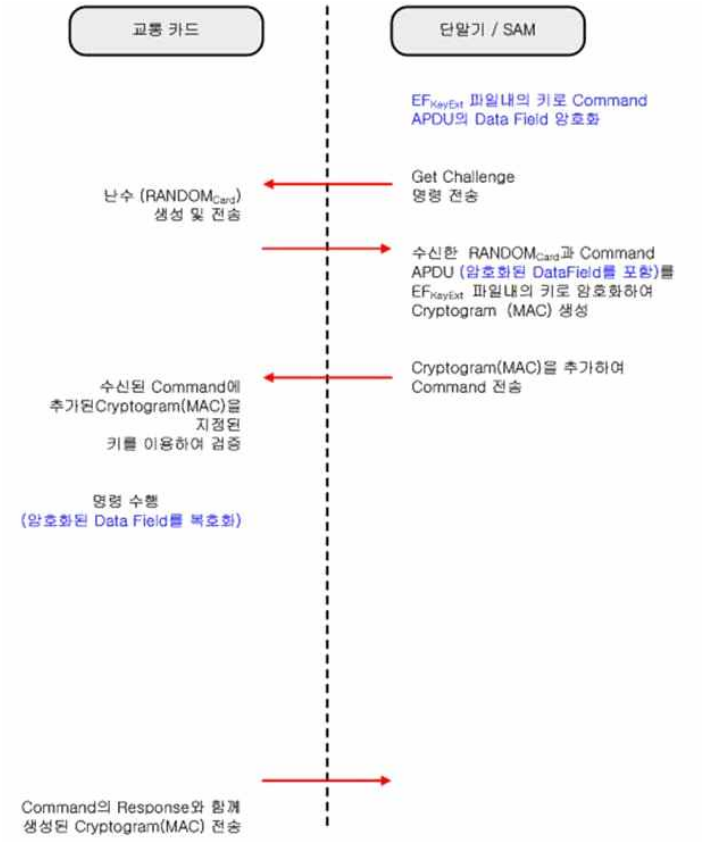


(그림 7-10) SEED Mode

7.4.2. E-MAC(Encrypted-MAC) Mode

MAC Mode와 동일하지만 전송되는 명령 또는 응답 내의 Data를 암호화한 뒤 이를 이용하여 계산된 인증값(Cryptogram)을 추가한다는 점이 MAC Mode와 틀리다.

E-MAC 모드에서의 명령 수행 과정은 다음과 같다.



(그림 7-11) E-MAC(Encrypted-MAC) Mode

■ 명령 전송 시 Cryptogram 계산 방법(단말기/SAM)

Cryptogram = 3-DES(Random + Command Header + Encrypted Command Data + Padding, Key)
Cryptogram = SEED(Random + Command Header + Encrypted Command Data + Padding, Key)

- Cryptogram 생성과 Command Data의 암호화에는 EFAUTH_KEY 파일 내의 키를 사용한다(Access Condition 내에 지정된 키 번호를 사용).

<--Random-->	<----Command Header---->					<-Command Data->	<-Padding->
Random (Padding)	CLA	INS	P1	P2	P3	Data Field **(Encrypted)**	Padding Block
8Byte (8Byte)	1Byte	1Byte	1Byte	1Byte	1Byte	Data Bytes	X Bytes

(그림 7-12) Cryptogram 계산 시 암호 알고리즘의 입력 데이터

Random 데이터는 3-DES의 경우 8 바이트를 입력 데이터의 가장 첫 블록으로 사용하고SEED의 경우는 Random에 8 바이트의 Padding Byte(80 00 …¦ 00)를 추가하여 첫 블록으로 사용한다.

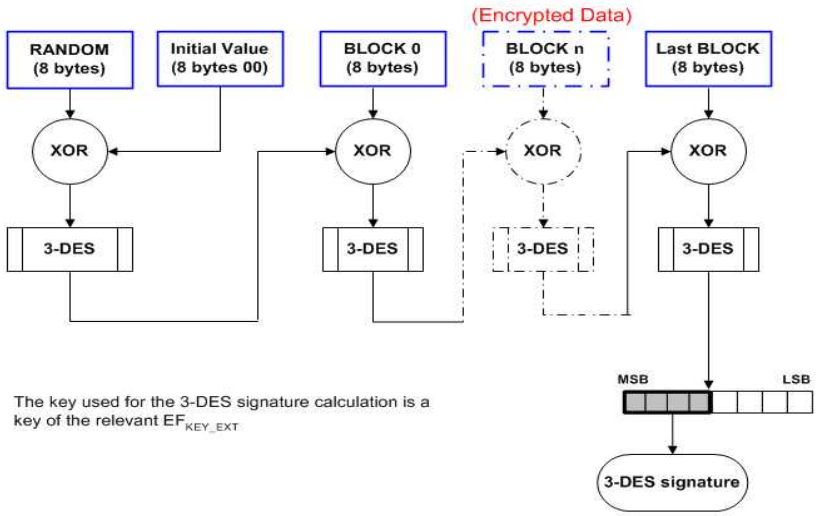
명령 전송 시 P3(Lc)는 암호화된 Data length + 4(Cryptogram 크기) 이며 응답 수신 시 필요한 P3(Le)는 암호화된 응답의 길이 + 4(Cryptogram 크기)이다.

Padding Block의 길이 X는 사용 암호 알고리즘에 따라 마지막 블록이 SEED일 경우 16-Byte, 3-DES 일 경우 8-Byte가 되도록 Padding Byte(80 00 …¦00)를 채워준다.

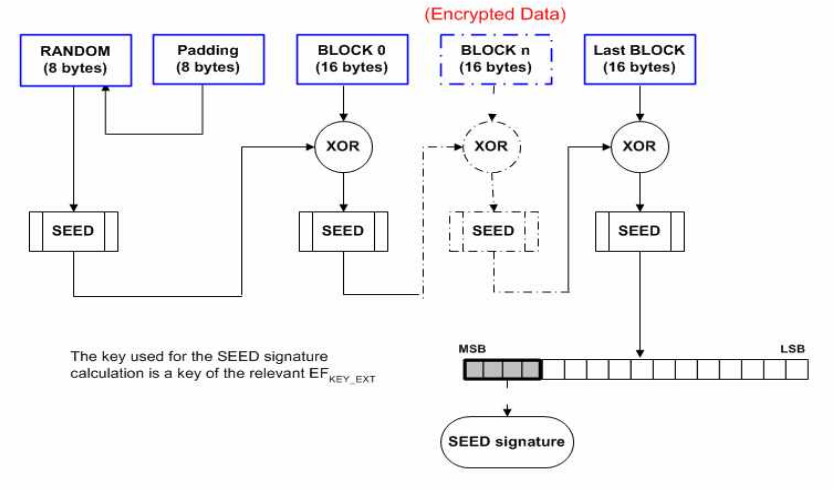
암호 알고리즘 수행 결과의 최상위 4 Bytes만 Cryptogram(MAC)으로 사용된다.

■ 응답 전송 시 Cryptogram 계산 방법(교통카드)

V1과의 호환성을 위하여 '0x00 00 00 00'을 MAC 값으로 응답한다.



(그림 7-13) 3-DES Mode



(그림 7-14) SEED Mode

8. 수명 주기

8.1. 수명 주기별 카드의 동작

발급 이전 상태에서는 거래 명령어를 제외한 모든 명령어가 사용 가능하다. 사용 상태에서는 모든 명령어(거래 명령어 포함)가 정상적으로 사용 가능하며 카드의 발급이 완료되었음을 나타낸다.

8.2. 수명 주기 상태

모바일 후불카드의 수명주기는 아래와 같으며 이전 상태로 돌아갈 수 없다.

<표 8-1> 수명 주기 상태

상태	Life Cycle
발급 이전 상태(초기 값)	03
사용 상태	07

9. APDU

9.1. APDU 구조

9.1.1. 명령어 APDU 구조

명령어 APDU는 다음과 같이 Command Header와 명령어에 따라 가변적인 Command Body로 나뉜다.

<-------- Command Header ------->				<---------- Command Body ---------->		
CLA (1Byte)	INS (1Byte)	P1 (1Byte)	P2 (1Byte)	Lc (1Byte or Empty)	Data (가변)	Le (1Byte or Empty)

(그림 8-1) 명령어 APDU 구조

<표 8-2> 명령어 APDU 구조

항 목	내 용	항 목	내 용
CLA	명령어의 분류 코드(Class)	Lc	명령 수행 시 보내는 Data 의 길이
INS	명령어 코드	Data	명령 수행 시 보내는 Data
P1	명령어 파라미터	Le	명령 수행 후 기대되는 응답의 길이
P2	명령어 파라미터		

■ CLA 바이트의 경우 다음과 같은 값이 사용된다.

<표 8-3> CLA 바이트

상위 nibble	하위 nibble	의 미
0	X	일반 명령어
0	0	암호화, MAC 사용 안함
0	4	명령 수행 시 MAC Mode 로 수행
9	X	카드의 전용 명령어
9	0	암호화, MAC 사용 안함
9	4	명령 수행 시 MAC Mode 로 수행
A	X	암호화를 사용해야 하는 명령어
A	4	명령 수행 시 E-MAC Mode 로 수행

9.1.2. 응답 APDU 구조

응답 APDU는 다음과 같이 가변적인 Response Body(응답 데이터)와 2바이트의 Trailer(상태 코드)로 나뉜다.

<---------- Response Body ---------->	<-------Trailer(Status Word)----->	
Data(가변)	**SW1(1Byte)**	**SW2(1Byte)**

Data	응답 데이터
SW1, 2	명령어 처리 상태값

(그림 9-1) 응답 APDU 구조

9.2. 케이스별 명령어

명령어와 응답 메시지는 다음과 같이 4가지 형태로 사용된다.

<표 9-1> 케이스별 명령어

Case	명령(보내는) 데이터	응답(받는) 데이터	Le 값
Case 1	없음	없음	없음
Case 2	없음	있음	정확한 값
Case 3	있음	없음	없음
Case 4	있음	있음	0x00 또는 정확한 값

특히, E-Mac 모드의 command(A4)와 MAC 모드의 command(04/94)인 경우, case 4 형태를 가지며, Le 값으로 0x00 및 0x04에 대하여 모두 정상적인 응답을 할 수 있어야 한다.

예외의 경우로, Session Key를 이용하여 환승 정보 내역을 추가하는 Append command를 수행 시, 그 응답 데이터의 사이즈는 0이므로, 이 경우는 0x00만에 대하여 정상적인 응답을 한다.

따라서, 카드에 들어가는 Case 4 형태의 APDU의 Le 값으로 0x00를 권장한다.

주) Le 값이 0x00인 경우, ISO 규정에 의하여 카드는 응답할 수 있는 모든 데이터를 보낸다. 응답한 데이터가 없는 경우(데이터의 사이즈가 0인 경우) 카드는 상태 코드(status word) 만을 내 보낸다.

주) 카드는 Read Binary/Read Record 같이 partial reading을 지원하는 경우를 제외한 모든 Case 2에 대하여, Le는 정확한 값에 대하여 동작한다.

10. 모바일 후불 파일시스템

이 카드는 'ISO/IEC 7816' Part4에서 규정한 파일 구조를 따른다.

10.1. 애플릿 정보

<표 10-1> 애플릿 정보

Applet information		
Package Name	com.moiba.applet	
Applet Name	com.moiba.applet.credittransport	
Install for Install		
Load File AID	D4 10 00 00 30	00 01 00 00
Applet AID	D4 10 00 00 30	00 01 00 00 01
Instance AID	D4 10 00 00 30	주 1) 참조
Application Privilege	00	
Install Parameters field	환승정보 내역파일 개수(1), 전자지갑 거래 파일 개수(1)	

주1) 모바일 후불 교통카드 AID

<표 10-2> 모바일 후불 교통카드 AID

구분	MOIBA RID	서비스	RFU	
Package AID	D4 10 00 00 30	00 01	00	00
Applet AID	D4 10 00 00 30	00 01	00	00 01

<표 10-3> 모바일 후불 교통카드 Instance AID 생성 규칙

구분	MOIBA RID	서비스	RFU	카드사	상품코드	No.
값	D4 10 00 00 30	00 01	00	주 2)	xx xx xx xx xx	주 3)
길이	5 Bytes	2 Bytes	1 Byte	2 Bytes	5 Bytes	1 Byte

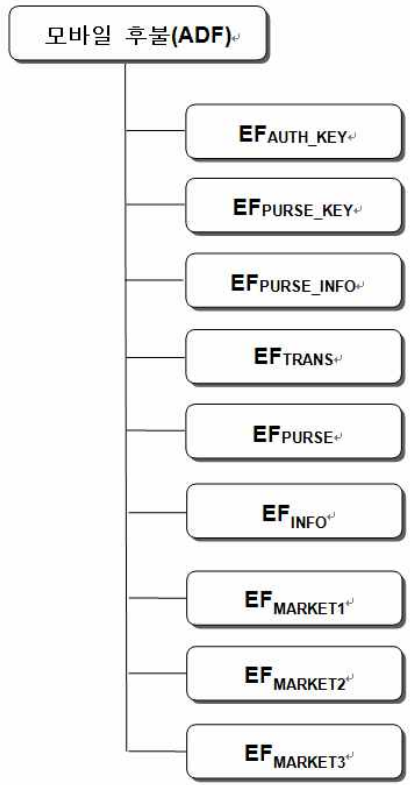
주2) 카드사 구분 코드
1st Byte : 카드사별 가나다순 할당하며, 추가되는 카드사는 마지막 번호 뒤로 할당
2nd Byte : 은행 구분이 필요할 경우 카드사에서 추가하여 관리하도록 함

<표 10-4> 카드사 구분 코드

카드사	1st Byte	2nd Byte
국민	01	00(기본값)
롯데	02	00(기본값)
BC(비씨)	03	00(기본값)
신한	04	00(기본값)

주3) 재발급 시 사용
기본값은 '00'으로 필요에 따라 증가한다.

10.2. 파일 구조



(그림 10-1) 파일 구조

이 DF는 교통카드 서비스에 대한 정보 및 internal File들을 가지는 Application DF이다.

모든 교통카드 서비스는 이 DF를 선택하고 난 후, 사용이 가능하다.

이 DF가 선택되면, 카드는 FCI 정보로 EFPURSE_INFOFile의 Record를 반환한다.

10.3. 모바일 후불 DF

<표 10-5> 모바일 후불 DF

파일식별자	AID			
내용	Write Access 조건	Update Access 조건	Read Access 조건	조건
EFPURSE_INFO, FCI	SD_AUTH	SD_AUTH	FREE	필수
EFTRANS	Key 인증 or 거래프로토콜[주 1)]	-	FREE	필수
EFPURSE	거래 프로토콜[주 1)]	-	FREE	필수

주1) 전자지갑파일의 DATA는 지불거래 프로토콜에 의해서 저장된다(별도 저장 명령어 없음).

10.3.1. FID 정의

Purse DF에서 관리하는 파일에 대한 FID는 다음과 같다.

<표 10-6> FID 정의

FID	File Type	File 구조	File Name	Description	Size	비고
	IEF	LF	EFTRANS_KEY	환승정보 키	16 x 4	
	IEF	LF	EFPURSE_KEY	Purse Key File	16 x 1	
0002	WEF	LF	EFPURSE_INFO	전자지갑 정보 파일	51 x 1	
0003	WEF	CR	EFTRANS	환승정보 내역 파일	52 x N	
0004	PEF	CR	EFPURSE	전자지갑 거래 파일	46 x N	
0011	WEF	TR	EFINFO		30	
0012	WEF	TR	EFMARKET1	유통용 데이터 파일	16	
0013	WEF	TR	EFMARKET2	유통용 데이터 파일	16	
0014	WEF	TR	EFMARKET3	유통용 데이터 파일	16	

10.3.2. EFAUTH_KEY(인증용 Key 파일)

Trans Key 파일은 현 ADF에서 환승 정보에 사용되는 Key들을 저장하고 있는 파일이다. Purse DF는 이 파일에 있는 Key들을 사용하여 정보를 저장한다.

<표 10-7> Access Condition

Operation	Command	AC Level	사용되는 Key	비고
Write	Issue Card(Initial)	SD_AUTH	-	
Read	-	NEV -	-	조회 불가
Update	-	NEV -	-	갱신 불가
Block/Unblock	-	-	-	

<표 10-8> File 구조

파일식별자			파일타입	IEF
파일크기	16 Bytes * 4		파일구조	Linear Fixed Record File(LF)
참조번호 (SID)	항목	내용	크기	비고
1	TRANSKEY	환승정보 키	16	키 인증 시도 잔여 횟수는 5
2	MARKET1KEY	유통용 1 키	16	키 인증 시도 잔여 횟수는 5
3	MARKET2KEY	유통용 2 키	16	키 인증 시도 잔여 횟수는 5
4	MARKET3KEY	유통용 3 키	16	키 인증 시도 잔여 횟수는 5

10.3.3. EFPURSE_KEY(PURSE Key 파일)

Purse Key 파일은 전자화폐 거래 시에 사용되는 Key들을 저장하고 있는 파일이다.

<표 10-9> Access Condition

Operation	Command	AC Level	사용되는 Key	비고
Write	Issue Card(Initial)	SD_AUTH	-	-
Read	-	NEV	-	조회 불가
Update	-	NEV	-	갱신 불가
Block/Unblock	-	-	-	-

<표 10-10> File 구조

파일식별자	-		파일타입	IEF
파일크기	16 Bytes *1		파일구조	Linear Fixed Record File(LF)
참조번호 (SID)	항목	내용	크기	비고
1	DPKEY	구매키	16	

EFPURSE_INFO(전자지갑 정보 파일, FCI)
Purse Information 파일은 전자지갑 관리에 필요한 정보를 저장한다.

이 파일은 FCI(File Control Information)파일로서 ADF 파일 선택 시에 응답된다.

<표 10-11> Access Condition

Operation	Command	AC Level	사용되는 Key	비고
Write	Issue_Card	SD_AUTH		
Read	Read Record	FREE	-	
Update	-	NEV		
Block/Unblock	-	-	-	

<표 10-12> File 구조

파일식별자	0002		파일타입	WEF	
파일크기	51 Bytes		파일구조	Linear Fixed File(LF)	
Record No	Tag	항목	크기	Value	비고
1	6F	PurseInfo Template	49		
	B0	파일 정보 TAG	47	파일 정보	

<표 10-12> 파일 정보

항목 이름	내용	크기	형식	비고
CARDTYPE	카드 구분 코드	1	HEX	*주(1) 참조
ALG	알고리즘 ID	1	HEX	*주(2) 참조
VK	전자 화폐 키셋버전	1	HEX	Default 값 = 01(HEX)
IDCENTER	전자 화폐사 ID	1	HEX	*주(3) 참조
CSN(IDEP)	신용카드번호	8	BCD	후불 카드번호, *주(4) 참조
IDTR	카드 사용 인증 ID	5	BCD	*주(5) 참조
DISS	발급일(YYYYMMDD)	4	BCD	“°YYYYMMDD”±
DEXP	만기일(YYYYMMDD)	4	BCD	“°YYYYMMDD”±
USERCODE	카드 소지자 구분 코드	1	HEX	*주(6) 참조
DISRATE	할인 코드	1	HEX	*주(7) 참조
BALMAX	최대 저장한도 금액	4	HEX	사용한도 금액
BRA	지점 코드	2	BCD	
MMAX	1 회 거래 제한 금액	4	HEX	(후불카드) 1 회 최대 사용한도
TCODE	이동통신사 구분 코드	1	HEX	*주(8) 참조
CCODE	신용카드사 구분 코드	1	HEX	*주(9) 참조
RFU	예비	8	HEX	0x00

■ 주)

1) 카드 구분 코드(CARDTYPE)

<표 10-13> 카드 구분 코드(CARDTYPE)

카드 구분	VALUE(HEX)	내용
후불카드	10	후불카드
	11 ~ 14	RFU
	15	모바일 후불카드
	16 ~1F	RFU

2) 알고리즘 구분(ALG)

<표 10-14> 알고리즘 구분(ALG)

알고리즘 구분	VALUE(HEX)	내용
SEED	00	SEED 사용
3-DES	10	3-DES 사용
Others		RFU

모바일 후불 교통카드는 3-DES만 사용한다.

3) 전자화폐사 ID(IDCENTER)

IDCENTER는 한국전자지불산업협회에서 지정한 교통 사업자의 고유 번호이다. 교통 호환 사업자는 고유의 IDCENTER값을 가지고 있으며 이를 표시하는 영역이다. 자세한 사항은 한국전자지불산업협회 또는 IDCENTER 관리기관의 전자화폐 고유 식별번호를 참조한다.

<표 10-15> 전자화폐사 ID(IDCENTER)

IDCENTER	사업자	IDCENTER	사업자
0x00	Reserved	0x07	한국도로공사
0x01	금융결제원	0x08	한국스마트카드
0x02	에이캐시	0x09	코레일네트웍스
0x03	마이비	0x0A	Reserved
0x04	Reserved	0x0B	이비
0x05	브이캐시	0x0C	서울특별시버스운송사업조합
0x06	몬덱스코리아	0x0D	카드넷

2009년 11월 한국전자지불산업협회에 등록된 전자화폐 고유 식별번호

4) 전자화폐ID(IDEP)

<표 10-16> 전자화폐ID(IDEP)

Byte	형식	내 용
1~3rd Byte	BCD	발행사 ID
4~8th Byte	BCD	카드 일련 번호

5) 카드 사용 인증ID(IDTR)

카드의 사용을 체크하기 위한 인증 ID이다. 각 발행사별로 부여한다.

6) 카드 소지자 구분 코드(USERCODE)

<표 10-17> 카드 소지자 구분 코드(USERCODE)

코드	내용	코드	내용
01	일반	06	경로
02	어린이	07~0E	RFU
03	RFU	0F	테스트
04	청소년	FF	비활성
05	RFU		

7) 할인 코드(DISRATE)

<표 10-18> 할인 코드(DISRATE)

분야	할인 코드값	설명
장애	0x10	기본
	0x11	동반 무임
	0x12~0x1F	RFU
유공	0x20	기본
	0x21	동반 무임
	0x22~0x2F	RFU

8) 이동통신사 구분 코드(TCODE)

<표 10-19> 이동통신사 구분 코드(TCODE)

이동통신사	값
SKT	01
KT	02
LGU+	03

9) 신용카드사 구분 코드(CCODE)

<표 10-20> 신용카드사 구분 코드(CCODE)

신용카드사	값	신용카드사	값
국민	01	CITI(씨티)	07
농협	02	외환	08
롯데	03	우리	09
BC(비씨)	04	하나 SK	0A
삼성	05	현대	0B
신한	06		

10.3.5. EFTRANS(환승 정보 내역 파일)

환승 정보 내역 파일은 교통 카드 이용 시, 환승 관련 로그 정보를 관리한다.

이 파일은 Cyclic Record 구조를 가지고 있으며, Record의 개수는 발급 시에 결정된다.

<표 10-21> Access Condition

Operation	Command	AC Level	사용되는 Key	비고
Write	Append Record	MAC	EFAUTH_KEY 의 Key1	
	Protocol		지불 Protocol 의 Session Key	참고 1)
Read	Read Record	FREE	-	
Update	-	-	-	
Block/Unblock	-	-	-	

참고1) 정상 지불 거래 후에는 거래 프로토콜에서 생성된 Session Key에 의하여 업데이트가 가능하다.

<표 10-22> File 구조

파일식별자	0003		파일타입	WEF	
파일크기	52 Bytes * N		파일구조	Linear Cyclic Record File(CR)	
Record No	Tag	항목	크기	Value	비고
1	01	환승 템플릿	50	파일 정보	

<표 10-23> 파일 정보

항목 이름	내 용	크기	형식	비고
RFU	예비 주 1) 참조	50	HEX	

■ 주)

1) 예비

환승 관련 세부 정보는 별도로 정의한다.

10.3.6. EFPURSE(전자 지갑 파일)

전자 지갑 파일은 전자지갑 거래를 위한 Cyclic Record 구조를 가지는 특수한 File이다.

거래 관련 명령을 통해서만 레코드 기록이 가능하며 충전 및 구매를 모두 포함한 거래 내역이 순차적으로 파일 내에 기록된다. 가장 최근의 거래 내역에 포함된 잔액 정보가 현재 카드의 잔액을 나타낸다.

<표 10-24> Access Condition

Operation	Command	AC Level	사용되는 Key	비고
Write	-	NEV	-	참고 1)
Read	Read Record	FREE	-	-
Update	-	NEV	-	-
Block/Unblock	-	-	-	-

참고1) 전자지갑파일의 DATA는 지불거래 프로토콜에 의해서 저장된다(별도 저장 명령어 없음).

<표 10-25> File 구조

파일식별자	0004		파일타입	PEF	
파일크기	46 Bytes *N		파일구조	Linear Cyclic Record File(CR)	
Record No	Tag	항목	크기	Value	비고
1	06	거래유형(TRT) 후불거래	44	파일 정보	

<표 10-26> File 정보

항목 이름	내 용	크기	형식	비고
BALEP	전자 지갑 사용 총액	4	HEX	주 1) 참조
NTEP	전자 지갑 거래 일련번호	4	HEX	주 2) 참조
M	거래 금액	4	HEX	주 3) 참조
IDSAM	SAM ID	8	BCD	주 4) 참조
NTSAM	SAM 거래 일련번호	4	HEX	주 5) 참조
TIME	거래 일시	7	BCD	주 6) 참조
RFU	Reserved for Future Used	13		

■ 주)

1) 잔액(BALEP)
후불 : 현재까지의 누적 사용 금액(4 Bytes), 사용 시 잔액 증가

2) 거래 번호(NTEP)
전자화폐 거래 발생 시 증가하는 4 Bytes 카운트. 애플리케이션에서 값을 읽을 수는 있지만 다시 Reset은 불가능하다.

3) 거래 금액(M)
각 구매 거래 등에서 사용된 가치

4) SAM ID(IDSAM)
각 구매 등의 거래에 사용된 SAM의 유일한 일련번호

5) SAM 거래 일련번호(NTSAM)
전자화폐 거래 발생 시 증가하는 4 Bytes 카운트로써, 각 SAM에서 생성된 카운트

6) 거래 일시(TIME)
전자화폐 거래 일시(YYYYMMDDhhmmss)

10.3.7. EFINFO(정보 파일)

후불 모바일 카드의 신용카드사의 발급 정보를 저장한다.

<표 10-27> Access Condition

Operation	Command	AC Level	사용되는 Key	비고
Write	-	-	-	
Read	Read Binary	FREE	-	-
Update	Update Binary	SD_AUTH	-	-
Block/Unblock	-	-	-	-

<표 10-28> File 구조

파일식별자	0011	파일타입	WEF
파일크기	30 Bytes	파일구조	TransParent File(TR)

<표 10-29> 파일 정보

항목 이름	내 용	크기	형식	비고
Issuer Code	발급사 정보	2	HEX	주 1)
Product Code	상품번호	6	BCD	발급사에서 임의 부여
Membership	멤버십정보	16	BCD	발급사에서 임의 부여
RFU		6	BCD	

주1) 발급사의 정보를 나타내며 자세한 사항은 향후에 정의한다. AID에서 발급사 정보를 표시하는 경우 이 값을 따른다.

10.3.8. EFMARKET1(유통용 데이터 파일)

전자 지갑 내부에 부가정보 서비스를 사용하기 위한 예비 파일이다.

<표 10-30> Access Condition

Operation	Command	AC Level	사용되는 Key	비고
Write	-	-	-	
Read	Read Binary	FREE	-	-
Update	Update Binary	MAC	$MARKET1_{KEY}$	-
Block/Unblock	-	-	-	-

<표 10-31> File 구조

파일식별자	0013		파일타입	WEF	
파일크기	16 Bytes		파일구조	TransParent File(TR)	
Record No	Tag	항목	크기	Value	비고
-	-	-	16	-	

10.3.9. EFMARKET2(유통용 데이터 파일)

전자 지갑 내부에 부가정보 서비스를 사용하기 위한 예비 파일이다.

<표 10-32> Access Condition

Operation	Command	AC Level	사용되는 Key	비고
Write	-	-	-	
Read	Read Binary	FREE	-	-
Update	Update Binary	MAC	$MARKET2_{KEY}$	-
Block/Unblock	-	-	-	-

<표 10-33> File 구조

파일식별자	0013		파일타입	WEF	
파일크기	16 Bytes		파일구조	TransParent File(TR)	
Record No	Tag	항목	크기	Value	비고
-	-	-	16	-	

10.3.10. EFMARKET3(유통용 데이터 파일)

전자 지갑 내부에 부가정보 서비스를 사용하기 위한 예비 파일이다.

<표 10-34> Access Condition

Operation	Command	AC Level	사용되는 Key	비고
Write	-	-	-	
Read	Read Binary	FREE	-	-
Update	Update Binary	MAC	$MARKET3_{KEY}$	-
Block/Unblock	-	-	-	-

<표 10-35> File 구조

파일식별자	0014		파일타입	WEF	
파일크기	16 Bytes		파일구조	TransParent File(TR)	
Record No	Tag	항목	크기	Value	비고
-	-	-	16	-	

11. 모바일 후불 명령어

11.1. 상태 코드(SW, Status Word) 정의

카드의 Application에서 사용되는 상태 코드는 다음과 같다.

<표 11-1> 상태 코드(SW, Status Word) 정의

SW1	SW2	의미	비고
90	00	성공적으로 수행함	ISO, KSX 6924
61	XX	성공적으로 명령어를 수행함. XX 길이(HEX)의 응답 데이터가 존재하며, Get Response 를 이용하여 응답 데이터를 획득할 수 있음.	ISO
63	CX	허용되는 오류 회수가 X 회(HEX) 남았음.	ISO
65	81	메모리가 손상됨.	ISO
67	00	잘못된 길이정보가 수신됨.	ISO, KSX 6924
69	81	명령어와 파일 구조가 호환되지 않음	ISO
	82	Access Condition 을 만족하지 않음	ISO, KSX 6924
	83	인증 명령어(Authentication Command)가 Block 됨	ISO, KSX 6924
	84	접근하고자 하는 데이터가 유용하지 않음.	ISO
	85	선행 명령어 또는 작업이 수행되지 않았음.	ISO
	86	허용되지 않는 명령어(no Current EF)	ISO
	87	기대된 보안 메시지(MAC 등) 없음.	ISO
	88	보안 메시지(Secure Message: MAC, Sign 등) 검증 실패	ISO, 일반명령어의 SW 만 해당
6A	80	데이터 필드 내 파라미터 오류	ISO
	81	지원되지 않는 기능임	ISO
	82	파일 또는 Application 이 존재하지 않음	ISO
	83	레코드가 존재하지 않음	ISO, Read/Update Record 시 사용
	84	메모리 공간이 충분하지 않음.	ISO
	85	TLV 구조가 아님	ISO
	86	P1, P2 오류	ISO, KSX 6924
	88	참조하고자 하는 데이터를 찾을 수 없음	7816-4, 7.5.1
6B	00	P1, P2 오류	ISO
6C	XX	Le 길이 오류., XX 는 예상되는 정상 Le 정보.	ISO, KSX 6924
6D	00	INS(Instruction)가 정의되어 있지 않음.	ISO, KSX 6924
6E	00	CLA(Class)가 정의되어 있지 않음.	ISO, KSX 6924
91	01	오류 데이터, 금액 오류(거래 금액이 카드 잔액보다 큼)	KSX 6924
	03	명령어의 순서 오류	KSX 6924
	04	거래 카운터 초과	KSX 6924
	0B	저장한도 초과	KSX 6924
	0F	서명 오류	KSX 6924
	10	등록되지 않은 알고리즘	KSX 6924
	11	키 버전 불일치	KSX 6924
	20	거래 상태 오류	KSX 6924
	21	등록되지 않은 IDCENTER	KSX 6924
	22	직전 거래 로그 IDSAM 불일치	KSX 6924
	23	직전 거래 로그 NTEP 불일치	KSX 6923
	24	직전 거래 로그 MPDA 불일치, 직전 거래 로그 불일치(M)	KSX 6924
	25	직전 거래 로그 IDEP 불일치	KSX 6923

11.2. INS(Instruction) 정의

11.2.1. INS별 설명

<표 11-2> INS별 설명

INS	Command Name	Short Description	See
A4	Select File	모바일 후불 DF 를 선택한다.	
C0	Get Response	카드로부터 전송되지 않은 데이터를 획득한다.	
84	Get Challenge	카드로부터 난수(Random Number)를 획득한다.	
82	External Auth	지정된 키를 이용하여 외부 인증을 획득한다.	GP 2.1.1.
50	Initialize Update	카드와 외부 호스트의 상호 인증하기위한 명령어	GP 2.1.1.
12	Issue Card	카드 발급 정보 저장	
CA	Get Data	카드로부터 CSN(Chip Serial Number) 또는 AppCode(Application Code)를 획득한다.	
B0	Read Binary	지정된 Binary 파일에 저장된 데이터를 읽는다.	
D6	Update Binary	지정된 Binary 파일에 데이터를 갱신한다.	
B2	Read Record	지정된 파일의 Record 에 저장된 데이터를 읽는다.	
DC	Update Record	지정된 파일의 Record 의 데이터를 갱신한다.	
E2	Append Record	지정된 파일에 새로운 Record 를 추가한다.	
4C	Read Balance	전자지갑 파일의 누적 거래 금액 정보를 읽어온다.	
30	Block	현재 선택된 DF 의 동작을 정지한다.	
32	Unblock	현재 정지된 DF 를 동작시킨다.	
E8	SetLifeCycle	Life Cycle 변경	
02	INITIALIZE CARD	KS X 6924 후불 거래/취소거래/재거래의 초기화 명령어	
04	PURCHASE CARD	KS X 6924 후불 거래/취소거래/재거래 명령어	

11.3. 일반 명령어

11.3.1. Select File

Select File 는 ADF를 선택한다. 이 명령어에 의해 선택된 파일이 현재의 파일이 된다.

<표 11-3> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	A4	04	00	XX	AID	VAR

<표 11-4> Response APDU

Response Data	SW1	SW2
FCI 데이터	XX	XX

11.3.2. Update Binary

Update Binary Command는 현재의 EF 또는 Parameter로 지정된 EF에 Data를 Update한다.

Transparent 구조의 파일에만 적용된다.

<표 11-5> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	D6	XX	XX	XX	Update Data	-
04					Update Data ‖ « MAC	-

<표 11-6> Input Data 구조

P1(HEX)	P2(HEX)	Lc(HEX)	Data	Le(HEX)
P1 의 b8 = 1 이면 P1 의 b7 과 b6 은 0, b5 ~ b1 은 SFI. P2 는 갱신할 Offset		Data 길이	Update Data(‖ « MAC)	-

<표 11-7> Response APDU

Response Data	SW1	SW2
(응답 MAC 값)	XX	XX

11.3.3. Read Binary

Read Binary 는 현재의 EF 또는 Parameter로 지정된 EF에서 Byte 데이터를 읽는다. Transparent EF 파일 구조에만 적용 가능하다.

<표 11-8> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	B0	XX	XX	-	-	XX

※ Read Binary는 Partial Reading이 가능하다.

<표 11-9> Input Data 구조

P1(HEX)	P2(HEX)	Lc(HEX)	Data	Le(HEX)
P1 의 b8 = 1 이면 P1 의 b7 과 b6 은 0, b5 ~ b1 은 SFI, P2 는 Read 할 Offset		-	-	XX 읽어올 Response Data 의 길이

<표 11-10> Response APDU

Response Data	SW1	SW2
Read Data From Binary File(Le Bytes)	XX	XX

11.3.4. Read Record

Read Record 는 현재의 EF 또는 Parameter로 지정된 EF에서 특정 Record의 내용을 읽어온다. 단, Record 구조의 파일에만 적용 가능하다.

<표 11-11> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	B2	XX	XX	-	-	XX

<표 11-12> Input Data 구조

P1(HEX)	P2(HEX)	Lc(HEX)	Data	Le(HEX)
Record 번호	b8~b4 는 SFI b3~b1 = 04 일 경우, P1 의 레코드 번호 이용	-	-	Read 될 바이트 수

<표 11-13> Response APDU

Response Data	SW1	SW2
Read Record Data(Le Bytes)	XX	XX

주) Read Record Command를 수행하고자 하는 파일에 Record가 정의되어 있지 않는 경우, 카드는 Error Code(0x6A83)를 응답한다.

11.3.5. Get Data

Get Data 는 카드로부터 필요한 정보를 읽어 온다.

<표 11-14> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	CA	01	01	-	-	08
			02			04
			03			01

<표 11-15> Input Data 구조

P1(HEX)	P2(HEX)		Lc(HEX)	Data 설명(DEC)	Le(HEX)
01	01	Chip Serial Number	-	-	08
	02	Application Code			04
	03	Card Status Code		사고자 코드주 1)	01

주1) 사고자 코드는 Put Data에 의해서 기록된다.

<표 11-16> Response APDU

Response Data	SW1	SW2
Output Data	XX	XX

<표 11-17> Output Data 구조

P1 ‖ « P2	항목	내용	크기 (Byte)	Le (HEX)
01 01	CSN	카드의 유일한 Chip Serial Number	8	08
01 02	AppCode	규격버전(2Bytes)||Applet 제조사버전(2Bytes)	4	04
01 03	Card Status Code	사고자 코드		

11.3.6. Put Data

Put Data는 카드의 사고자 코드를 기록하며 SD에 의해서 인증이 완료되어야 수행가능하다.

<표 11-18> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	DA	01	03	01	Card Status Code	-

<표 11-19> Input Data 구조

P1(HEX)	P2(HEX)		Lc(HEX)	Data 설명(DEC)	Le(HEX)
01	03	Card Status Code	01	사고자 코드주 1)	-

주1) 사고자 코드는 카드의 분실, 해지, 정지 등의 정보가 기록되며 구체적인 값은 발급사가 임의로 결정한다.

<표 11-20> Response APDU

Response Data	SW1	SW2
-	XX	XX

11.3.7. Initialize Update

카드와 외부 호스트가 상호 인증하기 위해서 사용하는 명령어로 카드는 자신의 난수를 생성하여 Card Cryptogram을 생성한다. 이때 생성된 난수와 세션 키는 본 명령어가 재실행되거나 카드 Reset 혹은 Applet이 Deselect되면 사라진다.

<표 11-21> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
80	50	Key Version Number 주 1)	Key Identifier 주 2)	08	Host Random Number(HRN)	00

주1) Key Version Number 는 'Global Platform 2.1.1' 참조.
주2) Key Identifier는 'Global Platform 2.1.1' 참조

<표 11-22> Response APDU

Response Data	SW1	SW2
Output Data	XX	XX

<표 11-23> Output Data 구조

P1 ‖ P2	항목	내용	크기(Byte)	Le(HEX)
XX XX	data	Key diversification data	10	0A
		Key information data	2	02
		Card challenge	8	08
		Card cryptogram	8	08

11.3.8. Get Challenge

Get Challenge 는 보안 관련 메커니즘에서 사용하기 위해, 카드에 Random Number 발생 요청 시 사용한다.

<표 11-24> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	84	00	00	-	-	08/10

<표 11-25> Input Data 구조

P1(HEX)	P2(HEX)	Lc(HEX)	Data	Le(HEX)
00	00	-	-	08/10

<표 11-26> Response APDU

Response Data	SW1	SW2
Output Data	XX	XX

<표 11-27> Output Data 구조

P1 ‖ P2	항목	내용	크기(Byte)	Le(HEX)
00 00	Random Number	카드(3-DES 사용)가 생성한 난수	8	08

11.3.9. External Authentication(SD)

카드와 외부 호스트가 상호 인증하기 위해서 사용하는 명령어로 카드는 외부로부터 입력된 Host Cryptogram을 검증한다. 이 명령어가 성공적으로 실행되면 상호 인증 상태는 카드가 Reset 혹은 Applet이 Deselect되거나 Initialize Update 명령어가 재실행될 때까지 유지된다.

<표 11-28> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
84	82	Security Level 주1)	00	10	Host Cryptogram || MAC 주2)	-

주1) Security Level
Security Level은 'Global Platform 2.1.1' 참조
주2) Host Cryptogram 과 MAC 생성 방법
Host Cryptogram과 MAC 생성방법은 'Global Platform 2.1.1' 참조

<표 11-29> Response APDU

Response Data	SW1	SW2
-	XX	XX

11.3.10. Issue Card

Initialize Update, ExternalAuthentication에 의해 인증이 된 후에 카드의 데이터 발급 및 키를 주입하는 명령이다.

데이터의 암호화 및 MAC 여부는 External Authentication(SD)의 Security Level에 따른다.

<표 11-30> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
80 84	12	XX 주1)	Block number	Var	발급정보 주2)	-

주1) 'Global Platform 2.1.1' store data 참조

<표 11-31> 발급정보

DGI	크기	항목	내용	갱신항목
0001	33	EFPURSEINFO	전자지갑 정보파일	X
0002	1E	EFINFO	신용카드사 발급정보파일	X
0011	13	TRANS KEY || KCV 주 3)	환승정보 키	X
0012	13	MARKET1 KEY || KCV	유통용 1 키	X
0013	13	MARKET2 KEY || KCV	유통용 2 키	X
0014	13	MARKET3 KEY || KCV	유통용 3 키	X
0021	14	VKDP_KEY || DP KEY || KCV	구매 키	X
0101	01	USERCODE	카드소지자 구분코드	O
0102	01	DISRATE	할인코드	O
0103	04	BALMAX	최대 저장한도금액	O
0104	02	BRA	지점 코드	O
0105	04	MMAX	1 회 거래 제한 금액	O
0106	08	RFU	예비	O

주3) KCV
Issue Card 명령 수행 시 데이터로 보내는 값으로써, 복호화 한 키를 아래의 절차로 통하여 획득한 값으로 계산 후 값이 일치할 경우 KEY를 갱신한다.
① 데이터는 8/16바이트의 '00'을 사용하고 키는 16 바이트를 사용한다.
② 사용하는 알고리즘은 3-DES 이며, 계산된 결과의 상위 3 바이트를 KCV로 사용한다.

주 4) 발급정보의 구성
DGI(2) || 길이(1) || 발급데이터(Variable)

<표 11-32> Input Data 구조

P1(HEX)	P2(HEX)	Lc(HEX)	Data	Le(HEX)
00 : More blocks 80 : Last block	Block number	Var	DGI || 발급데이터	-

<표 11-33> Response APDU

Response Data	SW1	SW2
-	XX	XX

11.3.11. Append Record

Append Record 는 모든 Record 구조의 파일에 하나의 레코드를 추가한다. 레코드 포인터는 현재 EF의 새로 만든 레코드를 가리킨다.

만약 Record가 최대로 생성된 Linear File에 Record를 추가하고자 할 경우에는 "메모리 부족" 에러가 발생한다. 그러나, Cyclic Record File의 경우에는 가장 오래된 Record 대신에 Append하고자 하는 Record가 기록되며, 이 Record의 번호가 1이다.

<표 11-34> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	E2	00	XX	XX	Append Data	-
04					Append Data ‖« MAC	-

주) Le값이 0x00이므로 ISO 규정에 따라 응답할 수 있는 Data 값을 보낸다.

<표 11-35> Input Data 구조

P1(HEX)	P2(HEX)	Lc(HEX)	Data	Le(HEX)
00	b8~b4 : SFI b3~b1 : 000	Data 의 길이	Append Data(‖« MAC)	-

<표 11-36> Response APDU

Response Data	SW1	SW2
(응답 MAC 값)	XX	XX

11.3.12. Get Response

Get Response 는 카드를 접촉으로 T=0 프로토콜에 의하여 사용되는 경우, 이전에 실행된 Command가 응답하는 데이터 중 아직 전송되지 않은 데이터를 전송 받을 수 있다.

<표 11-37> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	C0	00	00	-	-	XX

<표 11-38> Input Data 구조

P1(HEX)	P2(HEX)	Lc(HEX)	Data	Le(HEX)
00	00	-	-	XX

<표 11-39> Response APDU

Response Data	SW1	SW2
Output Data	XX	XX

<표 11-40> Output Data 구조

P1 ‖ P2	항목	내용	크기(Byte, DEC)	Le(HEX)
00 00	Response Data	카드가 응답하고자 하는 data	카드가 응답하고자 하는 데이터의 길이	XX

11.3.13. Block

Block은 현재 선택된 DF를 Block시켜 후불카드 거래 명령어를 사용할 수 없도록 한다.

Block의 Access Codition은 SD_AUTH 또는 환승 정보 키에 의한 MAC이다.

SD에 의한 인증 후 Block 수행 시 데이터의 MAC 여부는 Secure Domain 인증 시의 Security Level에 따라 정해진다.

<표 11-41> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
90	30	00	00	-	-	-
94				08/04	MAC	

<표 11-42> Input Data 구조

P1(HEX)	P2(HEX)	Lc(HEX)	Data	Le(HEX)
00	00	-	-	-
00	00	08 : Secure Domain 인증	MAC	-
00	00	04 : 환승정보키 인증(MAC 모드)	MAC	-

<표 11-43> Response APDU

Response Data	SW1	SW2
-	XX	XX

11.3.14. Unblock

Unblock은 Block되어 있는 DF를 Unblock하기 위하여 사용된다.

Unblock의 Access Codition은 SD_AUTH 또는 환승 정보 키에 의한 MAC이다.

SD에 의한 인증 후 Unblock 수행 시 데이터의 MAC 여부는 Secure Domain 인증 시의 Security Level에 따라 정해진다.

<표 11-44> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
90	32	00	00	-	-	-
94				08/04	MAC	

<표 11-45> Input Data 구조

P1(HEX)	P2(HEX)	Lc(HEX)	Data	Le(HEX)
00	00	-	-	-
00	00	08 : Secure Domain 인증	MAC	-
00	00	04 : 환승정보키 인증(MAC 모드)	MAC	-

<표 11-46> Response APDU

Response Data	SW1	SW2
-	XX	XX

11.3.15. Set Life Cycle

카드의 Life Cycle 상태를 설정하며 이전 상태로 되돌아 갈 수 없다.

데이터의 MAC 길이 여부는 Secure Domain 인증 또는 키 인증에 따라 정해진다.

<표 11-47> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
90	E8	00	07	-	-	-
94				08/04	MAC	

<표 11-48> Input Data 구조

P1(HEX)	P2(HEX)	Lc(HEX)	Data	Le(HEX)
00	07 : 사용상태	-	-	-
00	07 : 사용상태	08 : Secure Domain 인증	MAC	-
00	07 : 사용상태	04 : 환승정보키 인증(MAC 모드)	MAC	-

<표 11-49> Response APDU

Response Data	SW1	SW2
-	XX	XX

11.4. 전자지갑 서비스 일반 명령어

11.4.1. Read Balance

Read Balance는 전자지갑 파일의 누적 거래 금액 정보를 읽어온다.

<표 11-50> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
90	4C	00	00	-	-	04

■ Input Data 구조

<표 11-51> Response APDU

Response Data	SW1	SW2
BALEP	XX	XX

<표 11-52> Data 내용

P1 ‖ « P2	DATA			비고
	항목	내용	크기	
00 00	BALEP	전자 화폐의 현재 누적 거래 금액	4	

11.5. 후불 거래 명령어

11.5.1. INITIALIZE CARD(KS X 6924)

KS에 대응하는 명령어이다. Initialize Card는 전자지갑 거래를 하기 전에 각 거래를 초기화 하는 명령이다. Purchase, Re-Purchase 명령을 수행하기 전에 실행한다.

<표 11-53> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
90	02	XX	00	04	거래 금액	XX

주) Le값이 0x00이면, ISO 규정에 따라 응답할 수 있는 Data 값을 보낸다.

<표 11-54> Input Data 구조

P1(HEX)		P2(HEX)	Lc(HEX)	Data	Le(HEX)
10	후불카드 거래	00	04	거래 금액	17
11	후불카드(초기화 후) 거래	00			
12	후불카드 직전거래 취소	00	-	-	27

<표 11-55> Response APDU

Response Data	SW1	SW2
Output Data	XX	XX

<표 11-56> Output Data

P1 ‖« P2	Output Data			비고
	항목	내용	크기	
10 00	ALGEP	전자화폐사에서 사용되는 알고리즘	1	17
	VKEP	키 버전	1	
	BALEP	누적 거래 금액	4	
	IDCENTER	전자화폐사 ID	1	
	IDEP	전자화폐 ID	8	
	NTEP	전자화폐 거래 일련번호	4	
	Sign1	전자화폐에서 계산한 서명	4	
11 00	ALGEP	전자화폐사에서 사용되는 알고리즘	1	17
	VKEP	키 버전	1	
	BALEP	누적 거래 금액	4	
	IDCENTER	전자화폐사 ID	1	
	IDEP	전자화폐 ID	8	
	NTEP	전자화폐 거래 일련번호	4	
	Sign1	전자화폐에서 계산한 서명	4	
12 00	ALGEP	전자화폐사에서 사용되는 알고리즘	1	27
	VKEP	키 버전	1	
	BALEP	누적 거래 금액	4	
	IDCENTER	전자화폐사 ID	1	
	IDEP	전자화폐 ID	8	
	NTEP	전자화폐 거래 일련번호	4	
	IDSAM'	직전 거래 SAM ID	8	
	MPDA- EP'	직전 거래 금액	4	
	NTEP'	직전 거래 일련번호	4	
	Sign1	전자화폐에서 계산한 서명	4	

11.5.2. PURCHASE CARD('KS X 6924')

Purchase Card는 전자지갑 내의 누적 금액을 증가 시키고, 실제 거래를 수행하는 명령어이다. 거래를 위해 초기화 명령이 수행되어야 한다.

<표 11-57> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
90	04	XX	XX	XX	XX	04

주) Le값이 0x00이면, ISO 규정에 따라 응답할 수 있는 Data 값을 보낸다.

<표 11-58> Input Data 구조

P1(HEX)		P2(HEX)		Lc (HEX)	Data			Le (HEX)
					항목	내용	크기	
10	후불 거래	00	부가 정보 없을 경우	12	IDSAM	지불 SAM ID	8	04
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
		XX	부가 정보 없을 경우	Data 길이	IDSAM	지불 SAM ID	8	
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
					부가정보	저장하고자 하는 부가데이터	N	
11	(초기화 후) 후불 거래	00	부가 정보 없을 경우	12	IDSAM	지불 SAM ID	8	04
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
		XX	부가 정보 있을 경우	Data 길이	IDSAM	지불 SAM ID	8	
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
					부가정보	저장하고자 하는 부가데이터	N	

P1(HEX)		P2(HEX)		Lc (HEX)	Data			Le (HEX)
					항목	내용	크기	
12	직전 후불 거래 취소	00	부가 정보 없을 경우	12	IDSAM	지불 SAM ID	8	04
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
		XX	부가 정보 있을 경우	Data 길이	IDSAM	지불 SAM ID	8	
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
					부가정보	저장하고자 하는 부가데이터	N	
20	후불 거래	00	부가 정보 없을 경우	19	IDSAM	지불 SAM ID	8	04
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
					TIME	거래시간	7	
		XX	부가 정보 있을 경우	Data 길이	IDSAM	지불 SAM ID	8	
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
					TIME	거래시간	7	
					부가정보	저장하고자 하는 부가데이터	N	
21	(초기화 후) 후불 거래	00	부가 정보 없을 경우	19	IDSAM	지불 SAM ID	8	04
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
					TIME	거래시간	7	
		XX	부가 정보 있을 경우	Data 길이	IDSAM	지불 SAM ID	8	
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
					TIME	거래시간	7	
					부가정보	저장하고자 하는 부가데이터	N	

P1(HEX)		P2(HEX)		Lc (HEX)	Data			Le (HEX)
					항목	내용	크기	
22	직전 후불 거래 취소	00	부가 정보 없을 경우	19	IDSAM	지불 SAM ID	8	04
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
					TIME	거래시간	7	
		XX	부가 정보 있을 경우	Data 길이	IDSAM	지불 SAM ID	8	
					NTSAM	지불 SAM 거래 일련번호	4	
					SCSAM	지불 SAM의 구매 상태 코드	2	
					Sign2	지불 SAM에서 계산된 서명	4	
					TIME	거래시간	7	
					부가정보	저장하고자 하는 부가데이터	N	

주) P2가 XX인 경우, 상위 3 bits는 부가 데이터를 저장하고자 하는 파일의 Type을 의미하며, 하위 5 bits는 파일의 SFI를 의미한다.

주) P1이 20,21,22 이면 거래시간이 포함된다.

<표 11-59> Response APDU

Response Data	SW1	SW2
Sign3	XX	XX

<표 11-60> Output Data

P1 ‖ « P2	Output Data			비고
	항목	내용	크기	
-	Sign3	서명값.	4	

12. ConfigDF 파일시스템

12.1. Config DF(AID: A0 00 00 04 52 00 01)

교통 호환 ADF의 정보를 외부에 제공하는 역할을 하며 최초의 거래 시 필수적으로 Config DF를 선택하여야 한다.

12.1.1. Applet AID

Config DF의 Applet AID는 아래와 같다.

<표 12-1> Applet AID

Applet AID	A0 00 00 04 52 00 01

12.1.2. Package AID

Config DF의 Package AID는 아래와 같다.

<표 12-2> Package AID

Package AID	A0 00 00 04 52 00

12.1.3. Install Parameters

Config DF의 Install Parameters는 3 Bytes또는 11 Bytes로 아래와 같이 이루어진다.

<표 12-3> Install Parameters

AC	Access 조건1)
UAC	주카드 설정 Access 조건 2)
CLC	EF Config List 수
PIN	PIN data 값3)

1) Access 조건 : Update List와 Delete List 명령의 수행 조건을 표시한다.

<표 12-4> Access 조건

Parameters 값	Update Access 조건
0x01	키 인증
0x02	PIN1 인증
0x03	인증 or PIN1 인증
0x83	Key 인증 and PIN1 인증

2) 주 카드 설정 Access 조건: Update EF Config의 수행 조건을 표시한다.

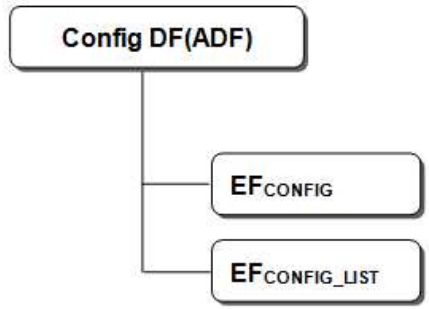
<표 12-5> 주 카드 설정 Access 조건

Parameters 값	Update Access 조건
0x00	Free
0x01	Key 인증
0x02	PIN2 인증
0x03	Key 인증 or PIN2 인증
0x83	Key 인증 and PIN2 인증

3) PIN data값 : PIN1과 PIN2의 초기값을 같은 값으로 입력한다. PIN값을 입력하지 않으면 임시 PIN값으로 초기화 한다.

- 임시 PIN : 0x3132333435363738

12.2. 파일 구조



(그림 12-1) 파일구조

12.3. Config DF

Config DF 아래에는 EFCONFIG와 EFCONFIG_LIST가 있으며 EFCONFIG는 교통 호환 정보를 저장하고 있으며, EFCONFIG_LIST는 EFCONFIG에 등록 가능한 교통 호환 정보를 갖는 지불수단을 지정된 개수만큼 저장한다.

<표 12-6> Config DF

AID	AID		
내용	Update Access 조건	Read Access 조건	조건
EFCONFIG	AC의 조건에 따라	FREE	필수
EFCONFIG_LIST	AC의 조건에 따라	FREE	필수

ConfigDF 선택 시 응답으로EFCONFIG의 내용을 응답하는 경우 아래의 형태를 따른다.

<표 12-7> FCI File 구조

파일식별자			파일타입	WEF	
파일크기	var		파일구조	LF	
Record No	Tag	항목	크기	Value	비고
1	6F	FCI Template	var		필수
	84	DF Name	var		필수
	A5	FCI Proprietary Template	var		필수
	50	카드 규격 및 선후불 구분	2	01 00(선불), 11 00(후불)	필수
	47	지원항목	2		필수
	43	IDCENTER	1		필수
	11	잔액조회명령	5		필수
	4F	교통 호환 ADF AID	5~16		필수
	9F10	부가정보파일	3*N		필수
	45	카드 소지자(카드타입) 정보	1		필수
	5F24	유효기간	2		필수
	12	카드일련번호	8		필수
	13	카드관리번호	8		선택
	BF0C	카드사업자 임의정보	17		선택

12.3.1. FID 정의

Config DF에서 관리하는 파일에 대한 FID는 다음과 같다.

<표 12-8> FID 정의

FID	File Type	File 구조	File Name	Description	Size (Size x list 수)
01	WEF,FCI	LF	EFCONFIG	Config 정보	N x 1
03	WEF	CR	EFCONFIG_LIST	Config 정보 목록	N x CLC

12.3.2. EFCONFIG(교통 호환용 Application 정보 파일)

Config DF 하위에 존재하며 실제 교통 호환용 Application에 대한 정보를 저장하고 있다. Read Record 명령으로 파일 내용의 정보가 조회 가능하다.

EFCONFIG를 통하여 얻을 수 있는 정보는 아래와 같다.

- 카드가 지원하는 “호환 카드”규격 버전
- “호환 카드” 규격 중 카드가 지원하는 항목
- 교통 호환 ADF의 AID
- 교통 호환 ADF에서의 존재하는 부가정보 파일의 정보
- 카드 소지자에 대한 정보
- 각 교통 호환 카드 사업자의 임의의 정보
- 카드 유효기간
- 잔액 조회 명령(선택 사항)

DF Select 시 FCI로 응답된다. FCI응답으로 EFCONFIG 파일의 1번 레코드의 데이터를 포함하는 경우 FCI Proprietary Template Tag인 ‘A5’ Tag에 포함 되어야 한다.

<표 12-9> Access Condition

Operation	Command	AC Level	사용되는 Key	비고
Read	Read Record		FREE	
Update	Update EF config		SD 인증 또는 PIN2 인증	

<표 12-10> File 구조

파일식별자	01		파일타입	WEF	
파일크기	var		파일구조	LF	
Record No	Tag	항목	크기	Value	비고
1	87	Config Data Template	var		필수
	50	카드 규격 및 선후불 구분	2	01 00(선불), 11 00(후불)	필수
	47	지원항목	2		필수
	43	IDCENTER	1		필수
	11	잔액조회명령	5		필수
	4F	교통 호환 ADF AID	5~16		필수
	9F10	부가정보파일	3*N		필수
	45	카드 소지자(카드타입) 정보	1		필수
	5F24	유효기간	2		필수
	12	카드일련번호	8		필수
	13	카드관리번호	8		선택
	BF0C	카드사업자 임의정보	17		선택

1) EFCONFIG 구성

주) 이 파일의 쓰기 권한은 각 교통 호환 사업자가 결정하며 16 바이트 이상의 키에 의해서 보호되거나 금지되어 있어야 한다.

카드는 위의 정보의 항목을 상기의 순서로 저장하도록 권고 하며 카드의 정보 파악을 위해서 위의 순서로 정보가 정렬되어 있다고 간주해서는 안된다.

EFCONFIG의 정보는 Config DF 선택 시 FCI로 응답할 수 있다.

2) 교통 호환 정보

교통 호환 정보는 EFCONFIG 파일의 1번 레코드에 저장되며 필수 항목이다. 교통 호환을 위한 데이터를 저장하고 있다.

● 카드 규격 및 선후불 구분

카드가 지원하는 호환 카드의 규격의 버전 및 선후불 구분 코드를 저장한다. 최상위 4 비트는 선불 및 후불 구분 코드로 사용되며 0의 경우 선불, 1의 경우 후불 카드를 나타낸다. 상위 바이트의 하위 4 비트로 메이저 버전을 하위 바이트로 마이너 버전을 표시한다.

현재의 규격이 1.0이므로 선불카드의 경우 '01 00'의 값을 갖고 후불카드의 경우 '11 00'의 값을 갖는다.

<표 12-11> 선후불 카드 규격

상위 바이트								하위 바이트								내용
b7	b6	b5	b4	b3	b2	b1	b0	b7	b6	b5	b4	b3	b2	b1	b0	
0	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	0x xx 의 경우 선불
0	0	0	1	x	x	x	x	x	x	x	x	x	x	x	x	1x xx 의 경우 후불
선/후불 구분				메이저 버전				마이너버전(1)				마이너버전(2)				

● 지원 항목

카드가 규격의 일부만을 지원하는 경우 이를 표시하는 영역이다.

각 항목을 지원하는 경우 각 비트에 '1'의 값을 지원하지 않는 경우에는 '0'의 값을 설정한다.

<표 12-12> 지원항목

비트								사업자
b7	b6	b5	b4	b3	b2	b1	b0	
–	–	–	–	–	–	–	1	ISO 14443-3 의 준수
–	–	–	–	–	–	1	–	ISO 14443-4 의 준수
–	–	–	–	–	1	–	–	Config DF 의 준수
–	–	–	–	1	–	–	–	하이패스의 지원 여부
x	x	x	x	–	–	–	–	예비 영역

● IDCENTER

IDCENTER는 한국전자지불산업협회에서 지정한 교통 사업자의 고유 번호이다. 교통 호환 사업자는 고유의 IDCENTER값을 가지고 있으며 이를 표시하는 영역이다. 자세한 사항은 한국전자지불산업협회 또는 IDCENTER 관리기관의 전자화폐 고유 식별번호를 참조한다.

IDCENTER	사업자	IDCENTER	사업자
0x00	Reserved	0x07	한국도로공사
0x01	금융결제원	0x08	한국스마트카드
0x02	에이캐시	0x09	코레일네트웍스
0x03	마이비	0x0A	Reserved
0x04	Reserved	0x0B	이비
0x05	브이캐시	0x0C	서울특별시버스운송사업조합
0x06	몬덱스코리아	0x0D	카드넷

2009년 11월 한국전자지불산업협회에 등록된 전자화폐 고유 식별번호

● 잔액 조회 명령

단말기에서 카드의 잔액을 조회하기 위해 사용할 명령이다. 잔액 조회 명령에 표시될수 있는 명령은 Case 2 명령으로 제한한다. 잔액 조회의 명령에 서명값 등 상수로 표현이 불가능한 경우에는 표시를 하지 않으며 또한 카드에서 잔액을 조회할 수 있는 명령이 없는 경우 표시하지 않는다. 또한 응답의 구조는 4 바이트 이하의 16진수 값만이 가능하다.

● 교통 호환 ADF AID

교통 호환 사업자 별 고유의 ADF의 AID이다. 교통 호환 ADF를 선택하기 위해서 사용한다.

● 부가 정보 파일

교통 호환 사업자가 부가 서비스를 하기 위하여 기록하는 부가 정보 파일의 정보를 가지고 있다. 부가 정보 파일의 종류, SFI 및 저장 가능한 최대 길이에 관한 정보를 나타낸다. 부가 정보 파일이 레코드 파일의 경우에는 길이에 Tag Length를 포함한 저장 가능한 최대 길이를 표시한다. 이 항목의 구성은 아래와 같다.

<표 12-13> 부가 정보 파일

Tag	Length	Value
	'03' * N	File Type(3bit) || SFI(5bit) || Max Length(2Byte) || [File Type(3bit) || SFI(5bit) || Max Length(2Byte)]....

부가 정보 파일은 1개 이상 존재해야 하며, 파일 하나당 3 바이트로 표현한다. 교통 사업자는 이 중 어떤 파일에 호환 정보를 기록 하여도 상관없으나 일관된 파일을 사용하여 정보를 조회 및 기록해야 한다. 또한, 호환카드 규격상 필요한 최대 길이 이상을 포함한 정보 파일이 1개 이상 존재해야 하며 카드 발급 시 처음 기록된 부가 정보 파일에 이를 표시하도록 권고한다.

주) File Type이 Record의 경우 Max Length는 Tag Length를 포함한 저장 가능한 순수 데이터의 길이를 나타낸다.

부가 정보 파일은 아래의 파일 구조 중 하나를 지원해야 한다.

<표 12-14> 부가 정보 파일

바이트(1)				바이트(2)	바이트(3)	File Type	지원 명령
File Type			SFI	MAX Length			
b7	b6	b5	b4 ~ b0	b7 ~ b0	b7 ~ b0		
0	0	0	x	x		RFU	-
0	0	1	x	x		Transparent	Read Binary
0	1	0	x	x		RFU	-
0	1	1	x	x		RFU	-
1	0	0	x	x		RFU	-
1	0	1	x	x		RFU	-
1	1	0	x	x		RFU	-
1	1	1	x	x		Cyclic Record	Read Record

● 소지자 정보

카드를 소지하고 있는 사람의 정보를 표시하며 자세한 사항은 아래와 같다.

<표 12-15> 소지자 정보

Value	설 명	Value	설 명
01	일반	11	버스
02	어린이	12	화물차
03	청소년	13	
04	경로	14	
05	장애인	15	

● 유효기간

카드 유효기간 정보를 표시하며, 형식은 'YYMM'으로 한다.

● 카드일련번호

키 변형 시 사용되는 카드번호 데이터 8 자리를 기록한다. 거래 초기화 시에 나오는 정보와 동일해야 한다.

● 카드관리번호

카드일련번호와 동일할 수도 동일하지 않을 수도 있으며, 사업자가 관리하는 카드번호이다.

12.3.3. EFCONFIG_LIST(교통 호환용 Application 정보 파일 목록)

Config DF 하위에 존재하며 교통 호환용 Application 에 대한 목록 정보를 저장하고 있다. Read Record 명령으로 파일 내용의 정보가 조회 가능하다.

EFCONFIG_LIST를 통하여 얻을 수 있는 각 목록의 정보는 아래와 같다.

- 카드가 지원하는 “호환 카드”규격 버전
- “호환 카드” 규격 중 카드가 지원하는 항목
- 교통 호환 ADF의 AID
- 교통 호환 ADF에서의 존재하는 부가정보 파일의 정보
- 카드 소지자에 대한 정보
- 각 교통 호환 카드 사업자의 임의의 정보
- 카드 유효기간
- 잔액 조회 명령(선택 사항)

<표 12-15> Access Condition

Operation	Command	AC Level	사용되는 Key	비고
Read	Read Record / Read List		Free	
Update	Update List		SD 인증 또는 PIN1 인증	
Delete	Delete List		SD 인증 또는 PIN1 인증	

<표 12-16> File 구조

파일식별자	03		파일타입	WEF	
파일크기	var		파일구조	LF	
Record No	Tag	항목	크기	Value	비고
1	87	Config Data Template	var		필수
	50	카드 규격 및 선후불 구분	2	01 00(선불), 11 00(후불)	필수
	47	지원항목	2		필수
	43	IDCENTER	1		필수
	11	잔액조회명령	5		필수
	4F	교통 호환 ADF AID	5~16		필수
	9F10	부가정보파일	3*N		필수
	45	카드 소지자(카드타입) 정보	1		필수
	5F24	유효기간	2		필수
	12	카드일련번호	8		필수
	13	카드관리번호	8		선택
	BF0C	카드사업자 임의정보	17		선택

1) EFCONFIG_LIST 구성

주) 이 파일의 쓰기 권한은 각 교통 호환 사업자가 결정하며 16 바이트 이상의 키 또는 PIN에 의해서 보호 되거나 금지되어 있어야 한다.

13. ConfigDF 명령어

13.1. 메시지 구조

13.1.1. 명령어 APDU 구조

명령어 APDU는 다음과 같이 Command Header와 명령어에 따라 가변적인 Command Body로 나눠진다.

<---------- Command Header --------->				<---------- Command Body ---------->		
CLA (1Byte)	INS (1Byte)	P1 (1Byte)	P2 (1Byte)	Lc (1Byte or Empty)	Data (가변)	Le (1Byte or Empty)

(그림 13-1) 명령어 APDU 구조

<표 13-1> 명령어 APDU 구조

항 목	내 용	항 목	내 용
CLA	명령어의 분류 코드(Class)	Lc	명령 수행 시 보내는 Data 의 길이
INS	명령어 코드	Data	명령 수행 시 보내는 Data
P1	명령어 파라미터	Le	명령 수행후 기대되는 응답의 길이
P2	명령어 파라미터		

■ CLA 바이트의 경우 다음과 같은 값이 사용된다.

<표 13-2> CLA 바이트

상위 nibble	하위 nibble	의 미
0	X	일반 명령어
0	0	암호화, MAC 사용 안 함
0	4	명령 수행 시 MAC Mode 로 수행
8	X	카드의 전용 명령어
8	0	암호화, MAC 사용 안 함
8	4	명령 수행 시 MAC Mode 로 수행
A	X	암호화를 사용해야 하는 명령어
A	4	명령 수행 시 E-MAC Mode 로 수행

13.1.2. 응답 APDU 구조

응답 APDU는 다음과 같이 가변적인 Response Body(응답 데이터)와 2 바이트의 Trailer(상태코드)로 나뉜다.

<---------- Response Body ---------->	<-------Trailer(Status Word)----->	
Data(가변)	SW1(1Byte)	SW2(1Byte)

Data	응답 데이터
SW1, 2	명령어 처리 상태값

(그림 13-2) 응답 APDU 구조

13.2. 케이스별 명령어

명령어와 응답 메시지는 다음과 같이 4가지 형태로 사용된다.

<표 13-3> 케이스별 명령어

Case	명령(보내는) 데이터	응답(받는) 데이터	Le 값
Case 1	없음	없음	없음
Case 2	없음	있음	정확한 값
Case 3	있음	없음	없음
Case 4	있음	있음	0x00 또는 정확한 값

특히, E-Mac 모드의 command(A4)와 Mac 모드의 command(04/94)인 경우, case 4 형태를 가지며, Le 값으로 0x00 및 0x04에 대하여 모두 정상적인 응답을 할 수 있어야 한다.

예외의 경우로, Session Key를 이용하여 환승정보 내역을 추가하는 Append command를 수행 시, 그 응답 데이터의 사이즈는 0이므로, 이 경우는 0x00만에 대하여 정상적인 응답을 한다.

따라서, 카드에 들어가는 Case 4 형태의 APDU의 Le 값으로 0x00를 권장한다.

주) Le 값이 0x00인 경우, ISO 규정에 의하여 카드는 응답할 수 있는 모든 데이터를 보낸다. 응답할 데이터가 없는 경우(데이터의 사이즈가 0인 경우) 카드는 상태코드(status word)만을 내 보낸다.

주) 카드는 Read Binary / Read Record 같이 partial reading을 지원하는 경우를 제외한 모든 Case 2에 대하여, Le는 정확한 값에 대하여 동작한다.

13.3. 상태 코드(SW, Status Word) 정의

카드의 Application에서 사용되는 상태 코드는 다음과 같다.

<표 13-4> 상태 코드

SW1	SW2	의미	비고
90	00	성공적으로 명령어를 수행함	
61	XX	성공적으로 명령어를 수행함. XX 길이(HEX)의 응답 데이터가 존재하며, Get Response를 이용하여 응답 데이터를 획득할 수 있음	
63	CX	허용되는 오류 회수가 X회(HEX) 남았음.	
65	81	메모리가 손상됨.	
67	00	잘못된 길이정보가 수신됨.	
69	81	명령어와 파일 구조가 호환되지 않음	
	82	Access Condition을 만족하지 않음. 접근 권한이 없음	
	83	인증 명령어(Authentication Command)가 Block됨 명령 실행 불가 수명 상태	
	84	접근하고자 하는 데이터가 유용하지 않음. 명령 실행 불가 모드	
	85	선행 명령어 또는 작업이 수행되지 않았음.	
	88	보안 메시지(Secure Message: MAC, Sign 등) 검증 실패	
6A	82	파일 또는 Application이 존재하지 않음	
	83	레코드가 존재하지 않음(Read/Update Record 시 사용)	
	84	메모리 공간이 충분하지 않음.	
	86	P1, P2 오류	
	88	참조하고자 하는 데이터를 찾을 수 없음	7816-4, 7.5.1
6B	00	P1, P2 오류	7816-5, 5.1.3
6C	XX	Le 길이 오류. XX는 예상되는 정상 Le 정보.	
6D	00	INS(Instruction)가 정의되어 있지 않음.	
6E	00	CLA(Class)가 정의되어 있지 않음.	

13.4. INS(Instruction) 정의

<표 13-5> INS(Instruction) 정의

INS	Command Name	Short Description	See
A4	Select File	DF/EF 등을 선택한다.	
B2	Read Record	지정된 파일의 Record에 저장된 데이터를 읽는다.	
CA	Get Data	카드로부터 AppCode(Application Code) 또는 EFconfig list 잔여 개수를 획득한다.	
82	External Auth	지정된 키를 이용하여 외부 인증을 획득한다.	GP 2.1.1.
50	Initialize Update	카드와 외부 호스트의 상호 인증하기위한 명령어	GP 2.1.1.
20	Verify PIN	PIN 인증을 한다.	.
24	Change PIN	Pin data 갱신한다.	
42	Read List	지정된 AID의 EFconfig list read한다	
44	Update List	지정된 AID의 EFconfig list Update한다	
46	Delete List	지정된 AID 의 EFconfig list를 delete한다	
48	Update EFconfig	지정된 AID의 EFconfiglist data를 EFconfig로 update 한다.	

13.5. 명령어

13.5.1. Select File

Select File Command는 ADF나 EF를 선택한다. 이 명령어에 의해 선택된 파일이 현재의 파일이 된다.

<표 13-6> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	A4	04	00	XX	AID	VAR

<표 13-7> Response APDU

Response Data	SW1	SW2
FCI 데이터	XX	XX

13.5.2. Read Record

Read Record Command는 현재의 EF 또는 Parameter로 지정된 EF에서 특정 Record의 내용을 읽어온다. 단, Record 구조의 파일에만 적용 가능하다.

<표 13-8> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	B2	XX	XX	-	-	XX

※ Read Record는 Partial Reading이 가능하다.

<표 13-9> Input Data 구조

P1(HEX)	P2(HEX)	Lc(HEX)	Data	Le(HEX)
Record 번호	b8~b4 는 SFI b3~b1 = 04 일 경우, P1 의 레코드 번호 이용	-	-	Read 될 바이트 수

<표 13-10> Response APDU

Response Data	SW1	SW2
Read Record Data(Le Bytes)	XX	XX

주) Read Record Command를 수행하고자 하는 파일에 Record가 정의되어 있지 않는 경우, 카드는 Error Code(0x6A83)를 응답한다.

13.5.3. Get Data

Get Data Command는 카드로부터 필요한 정보를 읽어 온다.

<표 13-11> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	CA	01	xx	-	-	XX

<표 13-12> Input Data 구조

P1(HEX)	P2(HEX)		Lc(HEX)	Data 설명(DEC)	Le(HEX)
01	01	Application Code	-	-	04
01	02	EFconfig list 잔여 개수	-	-	01

<표 13-13> Response APDU

Response Data	SW1	SW2
Output Data	XX	XX

<표 13-14> Output Data 구조

P1 ‖ P2	항목	내용	크기(Byte)	Le(HEX)
01 01	AppCode	규격버전(2Bytes)||Applet 제조사버전(2Bytes)	4	04
01 02	EF Config List	EF config list 남은 개수	1	01

13.5.4. Initialize Update

카드와 외부 호스트가 상호 인증하기 위해서 사용하는 명령어로 카드는 자신의 난수를 생성하여 Card Cryptogram을 생성한다. 이때 생성된 난수와 세션 키는 본 명령어가 재실행되거나 카드 Reset 혹은 Applet이 Deselect되면 사라진다.

<표 13-15> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
80	50	Key Version Number 주 1)	Key Identifier 주 2)	08	Host Random Number(HRN)	1C

주1) Key Version Number 는 'Global Platform 2.1.1' 참조.
주2) Key Identifier는 'Global Platform 2.1.1' 참조

<표 13-16> Response APDU

Response Data	SW1	SW2
Output Data	XX	XX

<표 13-17> Output Data 구조

P1 ‖« P2	항목	내용	크기(Byte)	Le(HEX)
XX XX	data	Key diversification data	10	0A
		Key information data	2	02
		Card challenge	8	08
		Card cryptogram	8	08

13.5.5. External Authentication

카드와 외부 호스트가 상호 인증하기 위해서 사용하는 명령어로 카드는 외부로부터 입력된 Host Cryptogram을 검증한다. 이 명령어가 성공적으로 실행되면 상호 인증 상태는 카드가 Reset 혹은 Applet이 Deselect되거나 Initialize Update 명령어가 재실행될 때까지 유지된다.

<표 13-18> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
84	82	Security Level 주 1)	00	10	Host Cryptogram || MAC 주 2)	-

주1) Security Level : 'Global Platform 2.1.1' 참조
주2) Host Cryptogram 과 MAC 생성 방법: 'Global Platform 2.1.1' 참조

<표 13-19> Response APDU

Response Data	SW1	SW2
-	XX	XX

13.5.6. Verify PIN

Verify Command는 단말기로부터 전송된 PIN 값을 비교·검증하는 명령어이다

<표 13-20> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
00	20	00	XX	00	-	-
				08	PIN Data	

주) Le값이 0x00이므로 ISO 규정에 따라 응답할 수 있는 Data 값을 보낸다.

<표 13-21> Input Data 구조

P1(HEX)	P2(HEX)		Lc(HEX)	Data 설명(DEC)	Le(HEX)
00	01	PIN1	08	Updatelist 인증 시 필요한 PIN data	-
00	02	PIN2	08	Update EFconfig 인증 시 필요한 PIN data	-

<표 13-22> Response APDU

Response Data	SW1	SW2
-	XX	XX

13.5.7. Change PIN

Change PIN command는 PIN Data를 갱신하는 명령어이다. Change PIN 명령어는 PIN이 Block 되었을 경우 unblock 된다. SDkey 인증 또는 PIN 인증 후 명령이 가능하다.

<표 13-23> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
80	24	00	XX	08	Update PIN Data	-

주) Le값이 0x00이므로 ISO 규정에 따라 응답할 수 있는 Data 값을 보낸다.

<표 13-24> Input Data 구조

P1(HEX)	P2(HEX)		Lc(HEX)	Data 설명(DEC)	Le(HEX)
00	01	PIN1	08	Updatelist 인증 시 필요한 PIN data 변경	-
00	02	PIN2	08	Update EFconfig 인증 시 필요한 PIN data 변경	-

<표 13-25> Response APDU

Response Data	SW1	SW2
-	XX	XX

13.5.8. Read List

Read List Command는 단말기로부터 전송된 교통 호환 ADF AID로 EF config list에 있는 레코드 중 AID가 일치하는 특정 Record의 내용을 읽어온다.

<표 13-26> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
80	42	00	00	XX	교통 호환 ADF AID	XX
		00	01	-	-	XX
		01				

주) Le값이 0x00이므로 ISO 규정에 따라 응답할 수 있는 Data 값을 보낸다.

<표 13-26> Response APDU

Response Data	SW1	SW2
Output Data	XX	XX

<표 13-26> Output Data 구조

P1 ‖« P2	항목	내용	크기(Byte)	Le(HEX)
00 00	Read Record Data	교통 호환 ADF AID 에 의한 Record Data	X	XX
00 01	All record AID -First occurrence	EF config list 에 존재하는 유효한 Record 의 AID 주 1) 참조	X	XX
01 01㈜	All record AID -Next occurrence	EF config list 에 존재하는 유효한 Record 의 AID 주 1) 참조 First 가 실행되지 않은 경우 실행되지 않음	X	XX

주1) status word가 0x6310일 경우에 명령이 가능하다.

<표 13-26>

항목	값	내용	크기	
Tag	4F	교통 호환 ADF AID 의 Tag	1	유효한 레코드의 개수(Count)만큼 반복됨
Len	5-16	교통 호환 ADF AID 의 길이	1	
AID		교통 호환 ADF AID	5-16	

13.5.9. Update List

Update List Command는 단말기로부터 전송된 교통 호환 ADF AID로 EF config list에서 특정 Record의 내용을 갱신 또는 생성한다. EF config list에 있는 레코드 중 AID가 일치하는 특정 Record의 갱신하고 AID가 일치하는 Record가 존재하지 않을 경우 새로 생성한다. EF config로 등록된 list일 경우 EF config의 내용까지 갱신한다. Access Condition에 의해서 Key 인증이 필요한 경우 데이터의 암호화 및 MAC 여부는 External Authentication(SD)의 Security Level에 따른다.

<표 13-28> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
80	44	00	00	XX	Update record Data	-
84					Update record Data||MAC	-

주) Le값이 0x00이므로 ISO 규정에 따라 응답할 수 있는 Data 값을 보낸다.

<표 13-29> Response APDU

Response Data	SW1	SW2
-	XX	XX

13.5.10. Delete List

Delete List Command는 단말기로부터 전송된 교통 호환 ADF AID로 EF config list에서 특정 Record의 내용을 삭제한다. Access Condition에 의해서 Key 인증이 필요한 경우 데이터의 암호화 및 MAC 여부는 External Authentication(SD)의 Security Level에 따른다.

<표 13-30> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
80	46	00	00	XX	교통 호환 ADF AID	-
84					교통 호환 ADF AID || MAC	-

주) Le값이 0x00이므로 ISO 규정에 따라 응답할 수 있는 Data 값을 보낸다.

<표 13-31> Response APDU

Response Data	SW1	SW2
-	XX	XX

13.5.11. Update EF config

Update EF config Command는 단말기로부터 전송된 교통 호환 ADF AID로 EF config list에서 특정 Record의 내용을 찾아 EF config로 갱신한다. 단, AID가 EF config list 파일에 존재해야만 적용 가능하다. Access Condition에 의해서 Key 인증이 필요한 경우 데이터의 암호화 및 MAC 여부는 External Authentication(SD)의 Security Level에 따른다.

<표 13-32> Command APDU

CLA	INS	P1	P2	Lc	Data	Le
80	48	00	00	XX	교통 호환 ADF AID	-
84					교통 호환 ADF AID || MAC	-

주) Le값이 0x00이므로 ISO 규정에 따라 응답할 수 있는 Data 값을 보낸다.

<표 13-33> Response APDU

Response Data	SW1	SW2
-	XX	XX

14. 프로토콜

14.1. 1차 발급

<표 14-1> 1차 발급

이동통신사	Config DF
→SELECT Card manager	
	←RESPONSE (FCI 정보)
→Initialize Update	
	←RESPONSE (난수 | MAC)
→External Authentication	
	←RESPONSE (인증 결과)
→install for load	
→load applet	
→install for install	

14.2. 2차 발급

<표 14-2> 2차 발급

이동통신사	Config DF
→SELECT ConfigDF	
	←RESPONSE (FCI 정보)
→Initialize Update	
	←RESPONSE (난수 | MAC)
→External Authentication	
	←RESPONSE (인증 결과)
→update list(반복) (update record data)	
→update EF config (AID)	
	←RESPONSE

표준 작성 공헌자

표준 번호 : TTAK.KO-12.0240

이 표준의 제정·개정 및 발간을 위해 아래와 같이 여러분들이 공헌하였습니다.

구분	성명	위원회 및 직위	연락처 (E-mail 등)	소속사
과제 제안	최영준	선임	02-405-6635 yjchoi@kisa.or.kr	KISA
표준 초안 제출	노용래	팀장	070-8765-8320 ylno2000@moiba.or.kr	MOIBA
초안 검토 및 표준안 심의	최재혁	매니져	jaehyuck.choi@sk.com	SK플래닛
	강유진	매니져	02-3495-4378 jinastory@kt.com	KT
	장철운	매니져	cwjang@lguplus.co.kr	LGU+
	신제춘	과장	02-6936-2615 zechoony@kbcard.com	KB국민카드
	이상희	대리	neonobless@lottecard.co.kr	롯데카드
	김현석	과장	02-3475-8016 phihop@bccard.com	비씨카드
	고영민	차장	02-6950-7389 uky75@shinhan.com	신한카드
	최웅조	책임	02-2028-9023 wjchoi@lotte.net	이비카드
사무국 담당	김영화	정보기술부 부장	031-724-0110 ykim@tta.or.kr	TTA
	오흥룡	정보기술부 선임	031-724-0083 hroh@tta.or.kr	TTA
	조은주	정보기술부 선임	031-724-0117 jej@tta.or.kr	TTA

정보통신단체표준(국문표준)

모바일 후불 교통카드
(Mobile Deferred Payment Traffic Card)

발행인 : 한국정보통신기술협회 회장
발행처 : 한국정보통신기술협회
463-824, 경기도 성남시 분당구 분당로 47
Tel : 031-724-0114, Fax : 031-724-0109
발행일 : 2014.04.