# USING DESFIRE COMMANDS

VERSION 100212

## TABLE OF CONTENTS

# 1 Overview

## 1.1 Required Reading

This document assumes you have read and are familiar with the *SkyeTek Protocol V3 Reference Guide* and the *SkyeTek Protocol V3 Basic Examples* application note.

## 1.2 Document Scope

This application note describes commands used for working with Philips DESFire family of tags. These SkyeTek Protocol v3 commands let you take advantage of the advanced features of DESFire tags without requiring familiarity with detailed DESFire tag operations. DESFire tags support the MIFARE Application Directory II (MADII) standard and permit operations such as accessing applications and crediting or debiting value files.

For additional information on command formats, see

- SkyeTek Protocol v3 Reference Guide
- SkyeTek Protocol v3 Examples

**NOTE** – Most DESFire commands require that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).

**NOTE** – The request and response data fields for the DESFire commands use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format listed for each command. However, you can easily use these commands without knowledge of the ASN.1 BER standard. Each command shows the formatted template for the data field and a literal example of how to use the data field for a realistic use of the command.

# 2 Select Tag (0x0101)

To begin working with your DESFire enabled tag you must first retrieve the tag's unique identification number (TID). This is accomplished by using the SkyeProtocol select tag command. This command takes a tag type as its argument however SkyeProtocol V3 allows the use of an auto-detect tag type (0000) for initial tag identification purposes. The structure of binary mode select tag request is shown in the table below.

## 2.1 Example – Selecting a Tag Using Auto-detect

Table 2-1: Select Tag Binary Mode Request

| Start | Msg Len | Flags | Command | Tag Type | CRC |
|---|---|---|---|---|---|
| 02 | 0008 | 0020 | 0101 | 0000 | F81A |

This command will query the first available tag in the reader's field. The response data for a DESFire EV1 tag is shown below.

Table 2-2: Select Tag Response

| Start | Msg Len | Cmd Resp | Tag Type | Data Len | Data | CRC |
|---|---|---|---|---|---|---|
| 02 | 000F | 0101 | 0214 | 0007 | 045E4931E81C80 | EB1E |

The select tag request was valid and the module indicates so by the inclusion of the request command in its response. In addition, this command returns the tag type as well as the tags TID. Now that the tag type and TID are known we can use this information with the same command to start a session with the tag. For the next request use the TID flag as well as the RF flag which instructs the reader module to start a session with the tag. This command example is shown below.

Table 2-3: Binary Mode Tag Selection

| Start | Msg Len | Flags | Command | Tag Type | Data Len | Data | CRC |
|---|---|---|---|---|---|---|---|
| 02 | 0010 | 0068 | 0101 | 0214 | 07 | 045E4931E81C80 | 2C68 |

The data in this request is the TID obtained from the previous response. If the tag is successfully selected it will return a session number as shown in the following example.

Table 2-4: Tag Selection Response

| Start | Msg Len | Cmd Resp | Data Len | Data | CRC |
|---|---|---|---|---|---|
| 02 | 0007 | 0101 | 0001 | 01 | 037F |

The data returned in the response is a session number which will be used during tag authentication.

# 3 Authenticate Tag

This command authenticates to the selected application.

## 3.1 Data Field Format (ASN.1BER Encoding)

### 3.1.1 Request

```
SEQUENCE
   {
   keyNumber                    INTEGER
   key                          OCTET-STRING
   }
```

### 3.1.2 Response

(There is no response data.)

## 3.2 Example – Authenticating to Tag

### 3.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 001E | 0C28 | 0201 | 0214 | 01 | 0011 | * | E1BC |

### 3.2.2 Data Field Format

| SEQ | INT | Key Number | OCT | OCT LEN | Key | END SEQ |
|---|---|---|---|---|---|---|
| 3080 | 0201 | 00 | 04 | 08 | 0000000000000000 | 0000 |

### 3.2.3 Response

| Start | Message Length | Command Response | CRC |
|---|---|---|---|
| 02 | 0004 | 0201 | 4158 |

# 4 Create Application

The DESFire Create Application command creates a new application. It takes a three-byte application identifier, application key settings, and the number of keys to be available for this application. You must select the master application (000000) with valid authentication before executing this command.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 4.1 Data Field Format (ASN.1BER Encoding)

```
SEQUENCE
   {
   ID                                   OCTET-STRING,
   application-key-settings             SEQUENCE
         {
         access-rights                  ENUMERATED
               {
               Key 0        (0),
               key 1        (1),
               …
               Key 13       (13),
```

```
                allowAll      (14),
                denyAll       (15)
                }
        configuration-frozen                BOOLEAN,
        auth-req-create-delete-file         BOOLEAN,
        auth-req-directory-access           BOOLEAN,
        allow-app-master-key-change         BOOLEAN
        }
num-of-keys                                 INTEGER
}
```

## 4.2 Example – Creating an Application with 7 Keys

### 4.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 002A | 0C28 | 0303 | 0214 | 01 | 001F | * | 475E |

### 4.2.2 Data Field Format

| SEQ | OCT | OCT LEN | AID | Start | ENUM | A | BOOL | B |
|---|---|---|---|---|---|---|---|---|
| 3080 | 04 | 03 | 000001 | 3080 | 0A01 | 0E | 0101 | 00 |

…

| BOOL | C | BOOL | D | BOOL | E | END SEQ | INT | F | END SEQ |
|---|---|---|---|---|---|---|---|---|---|
| 0101 | 00 | 0101 | 00 | 0101 | FF | 0000 | 0201 | 01 | 0000 |

- AID is the application identifier. 0x000000 denotes the master application and an application with this ID cannot be created.
- “A” is the access rights key setting. Use a specific key number or one of the following:
  - 0x0E allows all keys

  - 0x0F denies all keys except application master key
- "B" is a boolean that indicates the configuration is not frozen.
- "C" is a Boolean that indicates authentication is required for create or delete file.
- "D" is a boolean that indicates authentication is required for directory access.
- "E" region is a boolean that indicates application master key change is allowed.
- "F" is the number of keys to allocate for the application.

### 4.2.3 Response

| Start | Message Length | Data | CRC |
|---|---|---|---|
| 02 | 0004 | 0303 | 7B92 |

The reader echoes the command back indicating a success.

# 5 Create File (0x0403)

This command creates a file in the selected application. To create a file you must have first created an application and then selected that application.

## 5.1 Data Field Format (ASN.1BER Encoding)

```
SEQUENCE
   {
   file ID                          INTEGER
   fileType                         ENUMERATED
         {
         standardDataFile  (0),
         backupDataFile    (1),
         valueFile         (2),
         linearRecordFile  (3),
         cyclicRecordFile  (4)
         }
   communicationsSettings           ENUMERATED
         {
         plain             (0),
         plainWithMacing   (1),
         encrypted         (2),
         }
```

```
accessRights                    SEQUENCE
      {
      readAccess                ENUMERATED
      {
      Key 0             (0),
      Key 1             (1),
      …
      Key 13            (13),
      allowAll          (14),
      denyAll           (15)
      }
      writeAccess               ENUMERATED
      {
      Key 0             (0),
      Key 1             (1),
      …
      Key 13            (13),
      allowAll          (14),
      denyAll           (15)
      }
      readWriteAccess           ENUMERATED
      {
      Key 0             (0),
      Key 1             (1),
      …
      Key 13            (13),
      allowAll          (14),
      denyAll           (15)
      }
      changeAccess              ENUMERATED
      {
      Key 0             (0),
      Key 1             (1),
      …
      Key 13            (13),
      allowAll          (14),
      denyAll           (15)
      }
  }
CHOICE
  {
  dataFile              [1]
      SEQUENCE
```

```
        {
        fileSize                    INTEGER
        }
    valueFile               [2]
        SEQUENCE
        {
        lowerLimit                  INTEGER
        upperLimit                  INTEGER
        value                       INTEGER
        limitedCreditEnabled        BOOLEAN
        }
    recordFile              [3]
        SEQUENCE
        {
        recordSize                  INTEGER
        maxRecords                  INTEGER
        }
}
```

## 5.2 Example – Creating Standard or Backup File

This example creates a standard or backup data file ID 1 and file size 4.

### 5.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0033 | 0C28 | 0403 | 0214 | 01 | 0028 | * | C1CB |

### 5.2.2 Data Field Format

| SEQ | INT | A | ENUM | B | ENUM | C | SEQ | ENUM | D | ENUM | E |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 3080 | 0201 | 01 | 0A01 | 00 | 0A01 | 00 | 3080 | 0A01 | 0E | 0A01 | 0E |

…

| ENUM | F | ENUM | G | END | CHOICE | SEQ | INT | H | END | END | END |
|---|---|---|---|---|---|---|---|---|---|---|---|

| | | | | SEQ | | | | | SEQ | CHOICE | SEQ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0A01 | 0E | 0A01 | 0E | 0000 | A180 | 3080 | 0201 | 04 | 0000 | 0000 | 0000 |

- "A" is the one-byte file ID
- "B" is the file type:
  - 00 for a standard data file
  - 01 for a backup data file
- "C" is the communication setting:
  - 00 for plain
  - 01 for plain with making
  - 03 for encrypted
- "D" specifies read access. Use a specific key number or one of the following:
  - 0x0E for allow all
  - 0x0F for deny all
- "E" specifies write access. Use a specific key number or one of the following:
  - 0x0E for allow all
  - 0x0F for deny all
- "F" specifies the read/write access. Use a specific key number or one of the following:
  - 0x0E for allow all
  - 0x0F for deny all
- "G" specifies change access. Use a specific key number or one of the following:
  - 0x0E for allow all
  - 0x0F for deny all
- "H" is the file size.

### 5.2.3 Response

| Start | Message Length | Data | CRC |
|---|---|---|---|
| 02 | 0004 | 0403 | 369A |

The reader module echoes back the command indicating a success.

## 5.3 Example – Creating a Value File

### 5.3.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 003C | 0C28 | 0403 | 0214 | 01 | 0031 | * | 4EA8 |

### 5.3.2 Data Field Format

| SEQ | INT | A | ENUM | B | ENUM | C | SEQ | ENUM | D | ENUM | E |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 3080 | 0201 | 02 | 0A01 | 02 | 0A01 | 00 | 3080 | 0A01 | 0E | 0A01 | 0E |

…

| ENUM | F | ENUM | G | END SEQ | CHOICE | SEQ | INT | H | INT | I |
|---|---|---|---|---|---|---|---|---|---|---|
| 0A01 | 0E | 0A01 | 0E | 0000 | A280 | 3080 | 0201 | 00 | 0201 | 64 |

…

| INT | J | BOOL | K | END SEQ | END CHOICE | END SEQ |
|---|---|---|---|---|---|---|
| 0201 | 64 | 0101 | 00 | 0000 | 0000 | 0000 |

- “A” is the one-byte file ID.
- “B” is the file type.
  - 02 for value
- “C” is the communication settings:
  - 00 for plain
  - 01 for plain with making
  - 03 for encrypted
- “D” specifies that there is no read access.
- “E” specifies that there is no write access.
- “F” specifies that there is no read/write access.
- “G” specifies that there is no change access.

- "H" specifies a lower limit of 00
- "I" specifies an upper limit of 100 (0x64)
- "J" specifies an initial value of 100 (0x64)
- "K" specifies no limited credit flag

### 5.3.3 Response

| Start | Message Length | Data | CRC |
|---|---|---|---|
| 02 | 0004 | 0403 | 369A |

The reader echoes back the command indicating a success.

## 5.4 Command Example – Create Linear or Cyclic Record File

### 5.4.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0036 | 0C28 | 0403 | 0214 | 01 | 002B | * | 3B75 |

### 5.4.2 Data Field Format

| Start | INT | A | ENUM | B | ENUM | C | SEQ | ENUM | D |
|---|---|---|---|---|---|---|---|---|---|
| 3080 | 0201 | 03 | 0A01 | 03 | 0A01 | 00 | 3080 | 0A01 | 0E |

…

| ENUM | E | ENUM | F | ENUM | G | END SEQ |
|---|---|---|---|---|---|---|
| 0A01 | 0E | 0A01 | 0E | 0A01 | 0E | 0000 |

…

| CHOICE | SEQ | INT | H | INT | I | END SEQ | END CHOICE | END SEQ |
|---|---|---|---|---|---|---|---|---|
| A380 | 3080 | 0201 | 04 | 0201 | 0A | 0000 | 0000 | 0000 |

- "A" is the one-byte file ID.
- "B" specifies a linear record file. (A value of 04 would indicate a cyclic record file.)
- "C" specifies plain communications.
- "D" specifies allow all read access
- "E" specifies allow all write access
- "F" specifies allow all read/write access
- "G" specifies allow all change access
- "H" specifies a file size of 4 bytes
- "I" specifies the number of records to be 10.

### 5.4.3 Response

| Start | Message Length | Data | CRC |
|---|---|---|---|
| 02 | 0004 | 0403 | 369A |

The reader echoes back the response indicating a success.

# 6 Get Application IDs (0x0301)

The DESFire Get Application IDs command returns a list of application identifiers. Each application ID is three-bytes long.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- You must select the master application (000000) either explicitly or implicitly before using this command.

## 6.1 Data Field Format (ASN.1BER Encoding)

### 6.1.1 Request

(There is no request data)

### 6.1.2 Response

```
SEQUENCE
    {
    application ID                          OCTET-STRING
    }
```

## 6.2 Command Example

### 6.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | CRC |
|---|---|---|---|---|---|---|
| 02 | 0009 | 0428 | 0301 | 0214 | 01 | 30FA |

### 6.2.2 Response

| Start | Message Length | Command Response | Data Length | Data | CRC |
|---|---|---|---|---|---|
| 02 | 0019 | 0301 | 0013 | * | BFAA |

### 6.2.3 Data Field Format

| SEQ | OCT | OCT LEN | AID | OCT | OCT LEN | AID | OCT | OCT LEN | AID | End |
|---|---|---|---|---|---|---|---|---|---|---|
| 3080 | 04 | 03 | 000001 | 04 | 03 | 000002 | 04 | 03 | 000005 | 0000 |

- AID is the three-byte application ID.
- The response data always starts with 3080 and ends with 0000.
- For each returned application ID, 0403 is followed by the three-byte application ID. (i.e. 000001, 000002, and 000005 are the available applications)

# 7 Select Application (0x0302)

The DESFire Select Application command switches to the specified application ID. It takes a single argument: the three-byte application identifier to switch to.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).

## 7.1 Data Field Format (ASN.1BER Encoding)

### 7.1.1 Request

```
SEQUENCE
    {
    application ID                          OCTET-STRING
    }
```

### 7.1.2 Response

(There is no response data.)

## 7.2 Command Example

### 7.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0010 | 0C28 | 0302 | 0214 | 01 | 0005 | * | 0FE0 |

Selects the master application (0x000000).

### 7.2.2 Data Field Format

| OCT | OCT LEN | AID |
|---|---|---|
| 04 | 03 | 000000 |

### 7.2.3 Response

| Start | Message Length | Data | CRC |
|---|---|---|---|

| | | | |
|---|---|---|---|
| 02 | 0004 | 0302 | 6A1B |

The reader echoes back the command response indicating a success.

# 8 Delete Application (0x0304)

The DESFire Delete Application command deletes the specified application ID. It takes a single argument: the three-byte application identifier to delete. You must select the master application (000000) and perform authentication before executing this command.

- The command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).

## 8.1 Data Field Format (ASN.1BER Encoding)

### 8.1.1 Request

**app-id** OCTET-STRING

### 8.1.2 Response

(There is no response data.)

## 8.2 Command Example

### 8.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0010 | 0C28 | 0304 | 0214 | 01 | 0005 | * | AD3A |

### 8.2.2 Data Field Format

| OCT | OCT LEN | AID |
|---|---|---|
| 04 | 03 | 000005 |

This request tells the reader to delete the application with AID = 0x000005.

### 8.2.3 Response

| Start | Message Length | Data | CRC |
|---|---|---|---|
| 02 | 0004 | 0304 | 0F2D |

The reader echoes back the command indicating a success.

# 9 Get File IDs (0x0401)

The DESFire Get File IDs command returns a list of file identifiers. Each file identifier is one-byte long.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- This command requires that you select the application containing the files you wish to list.
- This command requires that you first authenticate to the selected application.

## 9.1 Data Field Format (ASN.1BER Encoding)

### 9.1.1 Request

(There is no request data.)

### 9.1.2 Response

```
SEQUENCE
    {
    file ID                          INTEGER
    }
```

## 9.2 Example

### 9.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | CRC |
|---|---|---|---|---|---|---|
| 02 | 0009 | 0428 | 0401 | 0214 | 01 | 0026 |

### 9.2.2 Response

| Start | Message Length | Command Response | Data Length | Data | CRC |
|---|---|---|---|---|---|
| 02 | 0013 | 0401 | 000D | * | A958 |

### 9.2.3 Data Field Format

| SEQ | INT | File ID | INT | File ID | INT | File ID | END SEQ |
|---|---|---|---|---|---|---|---|
| 3080 | 0201 | 01 | 0201 | 02 | 0201 | 03 | 0000 |

The data response field contains a sequence of integers each representing the file ID's for the selected application. (i.e. files 01, 02, and 03 exist in the selected application.)

# 10 Get File Settings (0x0404)

Get File Settings command takes a one-byte file identifier and returns the file type, communication settings, access rights, and file type specific settings.

- This command requires that you first create for the tag (i.e., include the tag ID with a Select Tag command).
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 10.1 Data Field Format (ASN.1BER Encoding)

### 10.1.1 Request

**file-id** INTEGER

### 10.1.2 Response

```
SEQUENCE {
  fileType                          ENUMERATED
        {
        standardDataFile  (0),
        backupDataFile    (1),
        valueFile         (2),
        linearRecordFile  (3),
        cyclicRecordFile  (4)
        }
  communicationSettings             ENUMERATED
        {
         plain            (0),
         plainWithMacing  (1),
         encrypted        (3)
        }
  accessRights                      SEQUENCE
        {
        readAccess                  ENUMERATED
        {
        Key 0            (0),
        Key 1            (1),
        …
        Key 13           (13),
        allowAll         (14),
        denyAll          (15)
        }
        writeAccess                 ENUMERATED
        {
        Key 0            (0),
        Key 1            (1),
        …
        Key 13           (13),
        allowAll         (14),
        denyAll          (15)
        }
```

```
            readWriteAccess                 ENUMERATED
            {
            Key 0               (0),
            Key 1               (1),
            …
            Key 13              (13),
            allowAll            (14),
            denyAll             (15)
            }
            changeAccess                    ENUMERATED
            {
            Key 0               (0),
            Key 1               (1),
            …
            Key 13              (13),
            allowAll            (14),
            denyAll             (15)
            }
      }

      CHOICE {
         dataFile               [1]
            SEQUENCE
            {
            fileSize                        INTEGER
            }
         valueFile              [2]
            SEQUENCE
            {
            lowerLimit                          INTEGER,
            upperLimit                          INTEGER,
            value                               INTEGER,
            limitedCreditEnabled                BOOLEAN
            }
         recordFile             [3]
            SEQUENCE
            {
            recordSize                          INTEGER,
            maxRecords                          INTEGER,
            numRecords                          INTEGER
            }
      }
   }
```

## 10.2 Example – Getting Settings for a Data File

### 10.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 000E | 0C28 | 0404 | 0214 | 01 | 0003 | * | A39A |

### 10.2.2 Request Data Field Format

| INT | File ID |
|---|---|
| 0201 | 01 |

This command requests the file settings for file 01.

### 10.2.3 Response

| Start | Message Length | Command Response | Data Length | Data | CRC |
|---|---|---|---|---|---|
| 02 | 002B | 0404 | 0025 | * | 2A8D |

### 10.2.4 Response Data Field Format

| Start | ENUM | A | ENUM | B | SEQ | ENUM | C | ENUM |
|---|---|---|---|---|---|---|---|---|
| 3080 | 0A01 | 00 | 0A01 | 00 | 3080 | 0A01 | 0E | 0A01 |

...

| D | ENUM | E | ENUM | F | END SEQ | CHOICE | SEQ |
|---|---|---|---|---|---|---|---|

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0E | 0A01 | 0E | 0A01 | 0E | 0000 | A180 | 3080 |

…

| INT | G | END SEQ | END CHOICE | END SEQ |
|---|---|---|---|---|
| 0201 | 04 | 0000 | 0000 | 0000 |

- "A" indicates the file type.
- "B" indicates the communication type.
- "C", "D", "E", and "F" indicate allow all access.
- "G" indicates the file size is 4 bytes.

# 11 Change File Settings (0x0405)

The DESFire Change File Settings command takes the one-byte file identifier, new communication settings, and new access rights to be set. You must perform a valid authentication to the application containing the file before executing this command.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 11.1 Data Field Format (ASN.1BER Encoding)

### 11.1.1 Request

```
SEQUENCE {
   fileID                                   INTEGER
         {
   communicationSettings                    ENUMERATED
         {
          plain              (0),
          plainWithMacing    (1),
          encrypted          (3)
```

```
        }
accessRights                                SEQUENCE
        {
        readAccess                          ENUMERATED
        {
        Key 0               (0),
        Key 1               (1),
        …
        Key 13              (13),
        allowAll            (14),
        denyAll             (15)
        }
        writeAccess                         ENUMERATED
        {
        Key 0               (0),
        Key 1               (1),
        …
        Key 13              (13),
        allowAll            (14),
        denyAll             (15)
        }
        readWriteAccess                     ENUMERATED
        {
        Key 0               (0),
        Key 1               (1),
        …
        Key 13              (13),
        allowAll            (14),
        denyAll             (15)
        }
        changeAccess                        ENUMERATED
        {
        Key 0               (0),
        Key 1               (1),
        …
        Key 13              (13),
        allowAll            (14),
        denyAll             (15)
        }
}
```

### 11.1.2 Response

(There is no response data.)

## 11.2 Example – Changing Settings for a Specific File

### 11.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0025 | 0C28 | 0405 | 0214 | 01 | 001A | * | E61D |

### 11.2.2Data Field Format

| SEQ | INT | A | ENUM | B | SEQ | ENUM | D | ENUM | E |
|---|---|---|---|---|---|---|---|---|---|
| 3080 | 0201 | 01 | 0A01 | 00 | 3080 | 0A01 | 00 | 0A01 | 00 |

…

| ENUM | F | ENUM | G | END SEQ | END SEQ |
|---|---|---|---|---|---|
| 0A01 | 00 | 0A01 | 00 | 0000 | 0000 |

- “A” is the one-byte file ID.
- “B” is the new communication setting
- “D”, “E”, “F”, and “G” specify the new access key settings.

### 11.2.3Response

| Start | Message Length | Data | CRC |
|---|---|---|---|
| 02 | 0004 | 0405 | 53AC |

The reader module echoes back the command indicating a success.

# 12 Read File (0x0406)

The DESFire Read File command is used to read from standard and backup data files. It takes a one-byte file identifier and an offset to start from.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- This command requires that you first authenticate to the application containing the file.
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 12.1 Data Field Format (ASN.1BER Encoding)

### 12.1.1 Request

```
SEQUENCE
   {
   fileId                    INTEGER,
   offset                    INTEGER
   }
```

### 12.1.2 Response

**Data** OCTET-STRING

## 12.2 Example – Reading a Data File

### 12.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0015 | 0C28 | 0406 | 0214 | 01 | 000A | * | 2128 |

### 12.2.2 Request Data Field Format

| SEQ | INT | A | INT | B | END SEQ |
|---|---|---|---|---|---|
| 3080 | 0201 | 01 | 0201 | 00 | 0000 |

- "A" indicates the file ID
- "B" indicates the file read starting offset.

### 12.2.3Response

| Start | Message Length | Command Response | Data Length | Data | CRC |
|---|---|---|---|---|---|
| 02 | 000C | 0406 | 000g | * | 930D |

### 12.2.4 Response Data Field Format

| OCT | OCT LENGTH | Data |
|---|---|---|
| 04 | 04 | 00000000 |

- Data contains the 4 bytes read from our file.

## 13 Write File (0x0407)

The DESFire Write File command writes to standard and backup data files. It takes a one-byte file identifier, an offset to start from, and the data to be written.

**NOTE** – If the file is a backup data file, the transaction must be committed.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 13.1 Data Field Format (ASN.1BER Encoding)

### 13.1.1 Request

```
SEQUENCE
   {
   fileId                          INTEGER,
   offset                          INTEGER,
   data                            OCTET-STRING
   }
```

### 13.1.2 Response

(There is no response data.)

## 13.2 Example – Writing to a Data File

### 13.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 001B | 0C28 | 0407 | 0214 | 01 | 0010 | * | 59EA |

### 13.2.2Data Field Format

| SEQ | INT | A | INT | B | OCT | OCT LENGTH | C | END SEQ |
|---|---|---|---|---|---|---|---|---|
| 3080 | 0201 | 01 | 0201 | 00 | 04 | 04 | FFFFFFFF | 0000 |

- "A" indicates the file ID.
- "B" indicates the offset to start writing from.
- "C" indicates the data to be written.

### 13.2.3 Response

| Start | Message Length | Command Response | CRC |
|---|---|---|---|

| | | | |
|---|---|---|---|
| 02 | 0004 | 0407 | 70BE |

The reader echoes back the response indicating a success.

# 14 Delete File (0x0408)

The DESFire Delete File command deletes the specified file. It takes a single argument: the one-byte file identifier to delete. You must perform a valid authentication on the file before executing this command.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 14.1 Data Field Format (ASN.1BER Encoding)

### 14.1.1 Request

```
file-id                                    INTEGER
```

### 14.1.2 Response

(There is no response data.)

## 14.2 Example – Deleting a Specific File

### 14.2.1Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0015 | 0C28 | 0408 | 0214 | 01 | 0003 | * | E3B5 |

### 14.2.2Data Field Format

| INT | File ID |
|---|---|
| 0201 | 01 |

This example deletes file 01.

### 14.2.3Response

| Start | Message Length | Command Response | CRC |
|---|---|---|---|
| 02 | 0004 | 0408 | 8849 |

The reader echoes back the command indicating a success.

# 15 Clear File (0x0409)

The DESFire Clear File command clears the content of a specified record file. It takes a single argument: the one-byte file identifier of the record file. You must perform a valid authentication on the file before executing this command.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 15.1 Data Field Format (ASN.1BER Encoding)

### 15.1.1 Request

```
file-id                         INTEGER
```

### 15.1.2 Response

(There is no response data.)

## 15.2 Example – Clearing a Specific File

### 15.2.1Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 000E | 0C08 | 0409 | 0214 | 01 | 0003 | * | 8D5A |

### 15.2.2Data Field Format

| INT | File ID |
|---|---|
| 0201 | 03 |

### 15.2.3Response

| Start | Message Length | Command Response | CRC |
|---|---|---|---|
| 02 | 0004 | 0409 | 99C0 |

The reader echoes back the command indicating a success.

# 16 Credit Value File

The DESFire Credit Value File command is used to credit a value file. It takes a one-byte file identifier and a value.

**NOTE**- As with all operations on a value file, this transaction must be committed.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).

## 16.1 Data Field Format (ASN.1BER Encoding)

### 16.1.1 Request

```
SEQUENCE
   {
   fileId                          INTEGER,
   value                           INTEGER
   }
```

### 16.1.2 Response

(There is no response data.)

## 16.2 Example – Crediting a Value to a Specific File

### 16.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0015 | 0C28 | 040A | 0214 | 01 | 000A | * | BA97 |

### 16.2.2Data Field Format

| SEQ | INT | File ID | INT | Value | END SEQ |
|---|---|---|---|---|---|
| 3080 | 0201 | 02 | 0201 | 03 | 0000 |

### 16.2.3Response

| Start | Message Length | Command Response | CRC |
|---|---|---|---|
| 02 | 0004 | 040A | AB5B |

# 17 Debit Value File (0x040B)

The DESFire Debit Value File command debits a value file. It takes a one-byte file identifier and a value.

**NOTE** – As with all operations on a value file, this transaction must be committed.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 17.1 Data Field Format (ASN.1BER Encoding)

Request

```
SEQUENCE
    {
    fileId                    INTEGER,
    value                     INTEGER
    }
```

Response

(There is no response data.)

## 17.2 Example – Deleting a Value from a Specific File

### 17.2.1Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0015 | 0C28 | 040B | 0214 | 01 | 000A | * | 6E5B |

### 17.2.2 Data Field Format

| SEQ | INT | File ID | INT | Value | END SEQ |
|---|---|---|---|---|---|
| 3080 | 0201 | 02 | 0201 | 02 | 0000 |

### 17.2.3Response

| Start | Message Length | Data | CRC |
|---|---|---|---|
| 02 | 0004 | 040B | BAD2 |

## 18 Limited Credit Value File (0x040C)

The DESFire Limited Credit Value File command credits back part of a value that was debited within the same transaction. An example of using this command could be a DESFire card that allows use of a coupon to partially refund all or part of a purchase amount while withholding permission to grant credit beyond the current debit amount. This could allow a vending machine to debit the card and add a limited credit but prevent putting greater amounts of money back onto the card. This command takes a one-byte file identifier and a value.

**NOTE** – As with all operations on a value file, this transaction must be committed.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- This command requires that you first authenticate to the application containing the value file you would like to limit.
- This command requires that you first perform a debit transaction.

### 18.1 Data Field Format (ASN.1BER Encoding)

Request

```
SEQUENCE
{
```

```
fileId                          INTEGER,
value                           INTEGER
}
```

Response

(There is no response data.)

## 18.2 Example – Setting a Limiting Credit Value for a Specific File

### 18.2.1Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0015 | 0C28 | 040C | 0214 | 01 | 000A | * | 3F01 |

### 18.2.2Data Field Format

| SEQ | INT | File ID | INT | Value | SEQ END |
|---|---|---|---|---|---|
| 3080 | 0201 | 01 | 0201 | 01 | 0000 |

### 18.2.3Response

| Start | Message Length | Command Response | CRC |
|---|---|---|---|
| 02 | 0004 | 040C | CE6D |

# 19 Get Value (0x040D)

The DESFire Get Value command gets the current value of a value file. It takes a one-byte file identifier.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).

- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 19.1 Data Field Format (ASN.1BER Encoding)

### 19.1.1 Request

**file-id** INTEGER

### 19.1.2 Response

**value** INTEGER

## 19.2 Example – Getting the Value from a Specific Value File

### 19.2.1Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 000E | 0C28 | 040D | 0214 | 01 | 0003 | * | 91AD |

### 19.2.2 Request Data Field Format

| INT | File ID |
|---|---|
| 0201 | 01 |

### 19.2.3Response

| Start | Message Length | Command Response | Data Length | Data | CRC |
|---|---|---|---|---|---|
| 02 | 0009 | 040D | 0003 | * | 6D96 |

### 19.2.4 Response Data Field Format

| INT | Value |
|---|---|
| 0201 | 32 |

# 20 Commit Transaction (0x040E)

The DESFire Commit Transaction command commits the current transaction for a backup data file or record file. It doesn't take any arguments or return any data.

**NOTE** – This command returns an error if you perform the commit when there is no pending transaction.

## 20.1 Command Example

### 20.1.1Request

| Start | Message Length | Flags | Command | Tag Type | Session | CRC |
|---|---|---|---|---|---|---|
| 02 | 0009 | 0428 | 040E | 0214 | 01 | B2DF |

### 20.1.2Response

| Start | Message Length | Data | CRC |
|---|---|---|---|
| 01 | 0004 | 040E | Ed7f |

# 21 Abort Transaction (0x040F)

The DESFire Abort Transaction command aborts the current transaction for a backup data file or record file. It doesn't take any arguments or return any data.

**NOTE** – This command returns an error if you perform the commit when there is no pending transaction.

## 21.1 Command Example

### 21.1.1Request

| Start | Message Length | Flags | Command | Tag Type | Session | CRC |
|---|---|---|---|---|---|---|
| 02 | 0009 | 0428 | 040F | 0214 | 01 | AE64 |

### 21.1.2Response

| Start | Message Length | Data | CRC |
|---|---|---|---|
| 02 | 0004 | 040F | FCF6 |

# 22 Read Records (0x0410)

The DESFire Read Records command reads from linear and cyclic record files. It takes a one-byte file identifier and a record number from which to start.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).

## 22.1 Data Field Format (ASN.1BER Encoding)

Request

```
SEQUENCE {
  fileId            INTEGER,
  value             INTEGER
}
```

Response

```
OCTET-STRING
```

## 22.2 Example – Reading Records from a Specific File

### 22.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0015 | 0C28 | 0410 | 0214 | 01 | 000A | * | B4A5 |

### 22.2.2Data Field Format

| SEQ | INT | File ID | INT | Offset | END SEQ |
|---|---|---|---|---|---|
| 3080 | 0201 | 03 | 0201 | 00 | 0000 |

### 22.2.3 Response

| Start | Message Length | Command Response | Data Length | Data | CRC |
|---|---|---|---|---|---|
| 02 | 0010 | 0410 | 000A | * | 983F |

### 22.2.4 Data Field Format

| OCT | OCT LENGTH | Record |
|---|---|---|
| 04 | 08 | 003C3812003C3812 |

# 23 Write Record (0x0411)

The DESFire Write Record command writes a record to a linear or cyclic record file. It takes a one-byte file identifier, a record offset to start from, and the data to be written.

**NOTE** – The transaction must be committed.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 23.1 Data Field Format (ASN.1BER Encoding)

### 23.1.1Request

```
SEQUENCE{
   fileID                    INTEGER,
   offset                    INTEGER,
   data                      OCTET-STRING
}
```

### 23.1.2Response

(There is no response data.)

## 23.2 Example – Writing to a Specific Record File

### 23.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0027 | 0C28 | 0411 | 0214 | 01 | 001C | * | B4E4 |

### 23.2.2 Request Data Field Format

| SEQ | INT | File ID | INT | Offset | OCT | OCT LENGTH | Record | END SEQ |
|---|---|---|---|---|---|---|---|---|
| 3080 | 0201 | 01 | 0201 | 00 | 04 | 10 | AABBCCDDEEFF00000000000000000000 | 0000 |

### 23.2.3 Response

| Start | Message Length | Command Response | CRC |
|---|---|---|---|
| 02 | 0004 | 0411 | 0509 |

# 24 Change Key Settings

The DESFire Change Key Settings command changes either the application or the master key settings.

- For application key settings, this command takes:
  - The access rights
  - Configuration frozen flag
  - A flag indicating authentication is required to create or delete files
  - A flag indicating authentication is required for directory access
  - A flag indicating if changes are allowed to the application master key
- For the master key settings, this command takes:
  - Configuration frozen flag
  - A flag indicating authentication is required to create or delete files
  - A flag indicating authentication is required for directory access
  - A flag indicating if changes are allowed to the master key
- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 24.1 Data Field Format (ASN.1BER Encoding)

### 24.1.1 Request

```
CHOICE {
  application-key-settings      [1]
   SEQUENCE {
        access-rights
              ENUMERATED {
              key0     (0)
```

```
                ..
                key13   (13),
                allowAll(14),
                denyAll (15)
                }
            configuration-frozen                    BOOLEAN,
            auth-req-create-delete-file             BOOLEAN,
            auth-req-directory-access               BOOLEAN,
            allow-app-master-key-change             BOOLEAN
    }
    master-key-settings             [2]
     SEQUENCE
            {
            configuration-frozen                BOOLEAN,
            auth-req-create-delete-app          BOOLEAN,
            auth-req-app-directory-access       BOOLEAN,
            allow-picc-master-key-change        BOOLEAN
            }
}
```

### 24.1.2 Response

(There is no response data.)

## 24.2 Example – Changing Application Key Settings

### 24.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 0022 | 0C28 | 0412 | 0214 | 01 | 0014 | * | 3D20 |

### 24.2.2 Data Field Format

| CHOICE | SEQ | ENUM | A | BOOL | B |
|---|---|---|---|---|---|
| A180 | 3080 | 0A01 | 0E | 0101 | 00 |

…

| BOOL | C | BOOL | D | BOOL | E | SEQ END | CHOICE END |
|---|---|---|---|---|---|---|---|
| 0101 | FF | 0101 | FF | 0101 | FF | 0000 | 0000 |

- "A" indicates allow all access rights.
- "B" indicates configuration is not frozen.
- "C" indicates authentication required to create/delete files.
- "D" indicates authentication required for directory access.
- "E" indicates allow master key change.

### 24.2.3 Response

| Start | Message Length | Command Response | CRC |
|---|---|---|---|
| 02 | 0004 | 0412 | 3792 |

The reader echoes back the command indicating a success.

## 24.3 Example – Changing Master Key Settings

### 24.3.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 001F | 0C28 | 0412 | 0214 | 01 | 0014 | * | 9730 |

### 24.3.2 Data Field Format

| CHOICE | SEQ | ENUM | A | BOOL | B |
|---|---|---|---|---|---|
| A280 | 3080 | 0A01 | 0E | 0101 | 00 |

…

| BOOL | C | BOOL | D | BOOL | E | SEQ END | CHOICE END |
|---|---|---|---|---|---|---|---|
| 0101 | FF | 0101 | FF | 0101 | FF | 0000 | 0000 |

- "A" indicates allow all access rights.
- "B" indicates configuration is not frozen.
- "C" indicates authentication required to create/delete files.
- "D" indicates authentication required for directory access.
- "E" indicates allow master key change.

### 24.3.3 Response

| Start | Message Length | Command Response | CRC |
|---|---|---|---|
| 02 | 0004 | 0412 | 3792 |

The reader echoes back the command indicating a success.

## 25 Get Key Settings (0x0413)

Get Key Settings command gets either the application or master key settings.

- For application key settings, this command returns:
  - The access rights
  - Configuration frozen flag
  - A flag indication authentication is required to create or delete files
  - A flag indicating authentication is required for directory access
  - A flag indication if changes are allowed to the application master key
- For the master key settings, this command returns:
  - Configuration frozen flag
  - A flag indicating authentication is required to create or delete files
  - A flag indicating authentication is required for directory access
  - A flag indicating if changes are allowed to the master key
- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 25.1 Data Field Format (ASN.1BER Encoding)

### 25.1.1 Request

None

### 25.1.2 Response

```
SEQUENCE {
  CHOICE {
    application-key-settings    [1]
        SEQUENCE {
      access-rights
            ENUMERATED {
            key0     (0)
            ..
            key13    (13),
            allowAll(14),
            denyAll  (15)
            }
      configuration-frozen                BOOLEAN,
      auth-req-create-delete-file         BOOLEAN,
      auth-req-directory-access           BOOLEAN,
      allow-app-master-key-change         BOOLEAN
    }
    master-key-settings         [2]
        SEQUENCE {
      configuration-frozen                BOOLEAN,
      auth-req-create-delete-app          BOOLEAN,
      auth-req-app-directory-access       BOOLEAN,
      allow-picc-master-key-change        BOOLEAN
    }
  },
  numOfKeys                               INTEGER
}
```

## 25.2 Example – Getting Application Key Settings

### 25.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | CRC |
|---|---|---|---|---|---|---|
| 02 | 0009 | 0428 | 0413 | 0214 | 01 | FAF1 |

### 25.2.2Response

| Start | Message Length | Command Response | Data Length | Data | CRC |
|---|---|---|---|---|---|
| 02 | 0024 | 0413 | 001E | * | 5D50 |

### 25.2.3Data Field Format

| SEQ | CHOICE | SEQ | ENUM | A | BOOL | B |
|---|---|---|---|---|---|---|
| 3080 | A180 | 3080 | 0A01 | 0E | 0101 | 00 |

…

| BOOL | C | BOOL | D | BOOL | E | SEQ END | CHOICE END |
|---|---|---|---|---|---|---|---|
| 0101 | FF | 0101 | FF | 0101 | FF | 0000 | 0000 |

…

| INT | F | END SEQ |
|---|---|---|
| 0201 | 07 | 0000 |

- "A" indicates allow all access rights.
- "B" indicates configuration is not frozen.
- "C" indicates authentication required to create/delete files.

- "D" indicates authentication required for directory access.
- "E" indicates allow master key change.
- "F" indicates the number of application keys.

# 26 Get Key Version (0x0414)

The DESFire Get Key Version command takes the key number and returns its version number. The version number is stored in the eighth bit, which was once used for parity bits of the Triple-DES key. (DES and later Triple-DES reserved every eighth bit as a parity bit. DESFire and other recent cipher implementations ignore these parity bits.)

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).
- The request and response data fields use ASN.1 BER (Tag Length Value) encoding, as specified in the Data Field Format, although you do not need knowledge of the ASN.1 BER standard to use the command.

## 26.1 Data Field Format (ASN.1BER Encoding)

Request

| | |
|---|---|
| **keyNO** | INTEGER |

Response

| | |
|---|---|
| **version** | INTEGER |

## 26.2 Example – Getting the Key Version

### 26.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 000E | 0C28 | 0414 | 0214 | 02 | 0003 | 020100 | 4D87 |

### 26.2.2Response

| Start | Message Length | Command Response | Data Length | Data | CRC |
|---|---|---|---|---|---|
| 02 | 0009 | 0414 | 0003 | 020100 | 4D87 |

The data contains an integer containing the key version (0x00 in this example).

# 27 Change Key (0x0415)

The DESFire Change Key command changes a key. It takes the key number, new key version, new key, and (if authentication was done with this key number) the current key.

**NOTE** – Valid authentication must be performed before executing this command.

- This command requires that you first create a session for the tag (i.e., include the tag ID with a Select Tag command).

## 27.1 Data Field Format (ASN.1BER Encoding)

### 27.1.1 Request

```
SEQUENCE {
   keyNo                    INTEGER
   keyVersion               INTEGER
   key                      OCTET-STRING
   currentKey               OCTET-STRING OPTIONAL
}
```

### 27.1.2 Response

(There is no response data.)

## 27.2 Example – Changing Key While Authenticated to Master Key

### 27.2.1 Request

| Start | Message Length | Flags | Command | Tag Type | Session | Data Length | Data | CRC |
|---|---|---|---|---|---|---|---|---|
| 02 | 001F | 0C28 | 0415 | 0214 | 01 | 0014 | * | 6966 |

### 27.2.2 Data Field Format

| SEQ | INT | A | INT | B | OCTET |
|---|---|---|---|---|---|
| 3080 | 0201 | 00 | 0201 | 00 | 04 |

…

| OCT LENGTH | INT | C | END SEQ |
|---|---|---|---|
| 08 | 0201 | FFFFFFFFFFFFFFFF | |

- "A" indicates key number.
- "B" indicates key version.
- "C" indicates key.

### 27.2.3 Response

| Start | Message Length | Command Response | CRC |
|---|---|---|---|
| 02 | 0004 | 0415 | 432D |

# 28 Revision History

| Revision | Author | Change |
|---|---|---|
| 100212 | Ryan Smith | Initial release. |
| | | |

Table 28-1: Revision History