bwCard

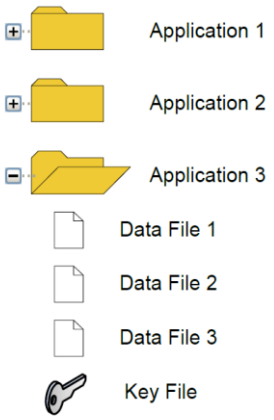
Steinbuch Centre for Computing (SCC)
Projekt bwCard
Ansprechpartner: Axel Maurer
Telefon: 0721 608-48752
E-Mail: axel.maurer@kit.edu
Stand: 01.07.2021

bwCard App
V 1.1 / 01. Juli 2021

Inhalt

Einführung

Zur übergreifenden Nutzung der bwCard ist es erforderlich einen gemeinsamen Standard zu setzen, anhand dessen jede bwCard im Anerkennungsraum individuell ist. Dieses Arbeitspapier bezieht sich derzeit nur auf eine entsprechende Kodierung einer bwCard, die einen Chip „mifare DESFire“ hat. Weitere Chiptechnologien werden hier nicht betrachtet.
Die DESFire-Karte hat grundsätzlich folgenden Aufbau:



Jede Karte hat eine individuelle vom Hersteller vergebene Seriennummer (UID). Diese liegt immer in der App „0“, File „0“ und kann nicht verändert werden. Es gibt allerdings die Möglichkeit diese UID vor unberechtigtem Auslesen zu schützen indem ein sogenannter PICC-Masterkey (proximity integrated circuit card) vergeben wird und die Funktion Random-Uid auf der Karte eingeschaltet wird. Ist dies der Fall, wird ohne Leseschlüssel eine „Random-ID“ ausgegeben, die aber als solche zu erkennen ist. Die UID selbst kann in diesem Fall immer ausgelesen werden, wenn sich gegen die Karte mit einem gültigen Schlüssel authentifiziert wurde. Das kann sowohl die PICC als auch der Schlüssel einer App sein. (s.u.)
Vergleichbar einem Filesystem, wie man es aus anderen Betriebssystemen kennt, ist die DESFire Karte hierarchisch aufgebaut.
Die erste Ebene wird als Application bezeichnet, der Bezeichner (Name) dazu nennt sich Application ID (AID). Diese Application ID besteht aus 3 Byte und ist frei wählbar. Es ist aber auch möglich, sich vom Hersteller NXP eine AID zuweisen zu lassen, die dann zumindest in diesem Umfeld eindeutig ist. Da jedoch die meisten Einrichtungen dies nicht vornehmen, ist damit die Eindeutigkeit nicht vollständig gewährleistet. Auf dem Kartentyp EV1 lassen sich bis zu 28 Apps anlegen, auf dem Kartentyp EV2 gibt es keine Beschränkung der Anzahl. Um das festzustellen, gibt es die Möglichkeit das Verzeichnis der Apps auf der Karte auszulesen.
Auf der zweiten und gleichzeitig letzten Ebene können in jeder App sogenannte „Data Files“ angelegt werden. Ein File ist dabei nichts anderes als eine Byte-Folge. Natürlich lässt sich dieser Byte-String beliebig strukturieren, aber das wird durch die Karte nicht explizit unterstützt und ist daher anwendungsspezifisch.
Zugriffsrechte gibt es auf Ebene der Karte - „root-Verzeichnis“ - und auf Ebene der App. Die Rechte werden durch entsprechende Schlüssel gesichert. Die Daten auf der Karte werden verschlüsselt. Als Verschlüsselungsalgorithmen werden AES, 3DES und weitere angeboten. Bei neuen Anwendungen wird die Verwendung von AES 128 empfohlen. Die Karte gibt es in verschiedenen Speichergrößen. Üblich sind inzwischen 8KByte.

Status Quo

Alle Karten, die im Verbund der Produktionsgemeinschaft bwCard personalisiert werden, bekommen eine App mit der AID „0xBECAAD“ aufgebracht. Diese enthält einen File „0“ mit 12 Byte. In diesem File wird die individuell vergebene Kartennummer gespeichert. Diese Nummer ist wie folgt aufgebaut:

<StaLa>	<Prüfziffer>	<Nummer>
1580	1	2547777

StaLa: Signatur der Hochschule gemäß dem Schlüsselverzeichnis für die Hochschulstatistik
Prüfziffer: Berechnet modulo 11 ohne Gewichtung
Nummer: 7-stellige fortlaufende Nummer

Die Nummer wird bei den meisten Einrichtungen auch auf die Karte aufgedruckt. Diese App wird NICHT als bwCard App genutzt.

Beschreibung bwCard App

Application ID
Die AID der bwCard App ist 0xF58860. Diese wurde bei NXP registriert.

Schlüssel der App
Der PICC-Masterkey der Karte und die Schreib- und Änderungsschlüssel der App können von der Einrichtung frei vergeben werden. Sie müssen für jede Einrichtung eindeutig sein. Der Leseschlüssel (Key 1) für die bwCard-App wird gemeinsam vergeben und vom KIT verwaltet (bwCard Key). Er wird bei Bedarf an die Mitglieder der Föderation weitergegeben. Der bwCard-Key muss als Verschlusssache aufbewahrt werden.

Files
Die Filegrößen wurden festgelegt auf:
File 0: bwcardNummer (20 Byte)
File 1: scope der Einrichtung (40 Byte)
File 2: UID in HEX Format (10 Byte)
File 3: ESCN (european student card number) (64 Byte)

bwcardNummer
Die bwCardNummer kann durch die Einrichtung frei vergeben werden. Sie kann höchsten 20 Byte umfassen und muss für jede Karte der Einrichtung eindeutig sein. Eine Wiederverwendung der bwCardNummer ist nicht vorgesehen. Jede neu ausgegebene Karte muss auch eine neue bwCardNummer erhalten.
Die bwCardNummer sollte als ASCII String und mit angehängten Nullwerten (0x0) kodiert werden.
Beispiel:
bwCardnummer: *158095325086*
Kodierung (Hex):
0x31353830393533323530383600000000000000000

Scope
Scope der Einrichtung ist die Domain der Einrichtung, wie sie in bwIDM verwendet wird. (Domain name according to RFC 1035)
Der Scope sollte als ASCII String und mit angehängten Nullwerten (0x0) kodiert werden.
Beispiel:
Scope: *kit.edu*
Kodierung (Hex)
0x6b69742e656475000000000000000000000000000000000000000000000000000000000000000000000000

UID
UID ist die Seriennummer der Karte. Im Fall, dass die Random-UID Funktion eingeschaltet ist, muss hier die UID abgelegt werden, die sich nach Authentifizierung an der Karte auslesen lässt (echt UID)
Sie wird in „most siginificant byte order“ kodiert. Im Gegensatz zu den anderen Files werde hier die Daten mit führenden Nullenwerten kodiert.
Beispiel:
UID: 04 0B 46 8A 1F 55 80
Kodierung (Hex):
0x000000040B468A1F5580

ESCN
Die „european student card number (ESCN)” wird aufgenommen, um eine zukünftige Umsetzung der ESC Kodierung (DEUINFO) vorzubereiten. Sie kann selbstverständlich auch für weitere Zwecke genutzt werden.
Sie wird als ASCII String kodiert und mit angehängten Nullwerten (0x0).
Beispiel:
ESCN: f4140375-d665-1038-b5cc-007990797674
Kodierung (Hex):
0x0000000000000000000000000000000000000000000000000000000000006634313430333735 2d643636352d313033382d623563632d3030373939303739373 63734

File 2 und 3 sind optional können auch mit „0“ gefüllt werden.
Platzbedarf 192 Byte (Overhead) + 96 Byte (Schlüssel) + 192 Byte (Nutzdaten in 32 Byte) = 480 Byte

Technische Spezifikation

Name=bwCard
AID=0xF58860 → bei “NXP MIFARE Application Directory” registriert

AppMasterKeySettings=0B

- Configuration changeable if authenticated with masterkey
- Retrieving FileIds, FileSettings and KeySettings without master key authentication
- Application master key changeable

CryptoMethod=AES (128)

IsoFileSupport=false

File 0 bwCardNumber
- StdData, 20 byte
- ReadBeforeWriting=1
- KeyDivAlg=0
- Keys: AES
- Communication type: enciphered
- Key1: read (bwCard Key)

File 1 scope (Domain name according to RFC 1035)
- StdData, 40 byte
- ReadBeforeWriting=1
- KeyDivAlg=0
- Keys: AES
- Communication type: enciphered
- Key1: read (bwCard Key)

File 2 UID (UID HEX as Binary))
- StdData, 10 byte
- ReadBeforeWriting=1
- KeyDivAlg=0
- Keys: AES
- Communication type: enciphered
- Key1: read (bwCard Key)

File 3 ESCN (european student card number)
- StdData, 64 byte
- ReadBeforeWriting=1
- KeyDivAlg=0
- Keys: AES
- Communication type: enciphered
- Key1: read (bwCard Key)