National Protective Security Authority

GUIDANCE DOCUMENT – V2

DESFire ‘EV’ Token Deployment Guide

Contents

Executive Summary

This document contains a “best practice” guide for deployment of NXP Mifare DESFire EV1 Tokens in Automatic Access Control Systems (AACS).

DESFire tokens are extremely complex to configure and can be used in a variety of ways, ranging from insecure to highly secure, depending on exact configuration. This guide is designed to ensure that only configurations that provide highly secure operation are chosen.

Section 2 of this guide provides the actual guidance and is deliberatively short.

Section 3 provides more detailed explanations of the reasoning behind each of the choices in Section 2.

Thus, Section 2 can be used to “cross-check” a deployment to ensure the correct choices have been made, and Section 3 can be used for more general guidance and to assist those performing the initial deployment.

1 Section 1 - Overview

This guide aims to bring together lessons learned through failures (and successes) of such systems. The guidance provides for two options during deployment:

- “Must Haves” - Items that must be deployed and must not be deployed in another configuration. These items will be marked “REQUIRED”.
- “Beneficial” - Items that further improve security but may be excluded if operationally infeasible. These items will be marked “RECOMMENDED” but may contain sub-entries that are “REQUIRED” if implemented.

Scope

In scope for this guide are NXP Mifare DESFire EV1 products, although similarities can be drawn for the whole range of EV products.

Out of scope for this guide are the AACS themselves, but there are some references to requirements for these systems in section 3.

2 Section 2 - TOKEN Deployment

2.1. ENCRYPTION STANDARD [REQUIRED]

Use AES encryption.

- AES should be used.
- 3DES is being retired and should not be used Single DES MUST NOT be used.
- Cryptographic keys should have good ENTROPY.

2.2. KEY DIVERSIFICATION [REQUIRED]

DIVERSIFY all other CRYPTOGRAPHIC KEYS.

- Use a diversification scheme recommended by the manufacturer. Include FILE NUMBER in diversification data for FILE keys.
- Include KEY VERSION or IDENTIFIER for non-FILE keys.
- Use a secure diversification environment & protocol.

2.3. APPLICATION (AID) PERMISSIONS [REQUIRED]

- AACS AIDs MUST NOT be DELETED or MODIFIED without authentication.
- KEY details (e.g., VERSION) MUST be accessible without authentication.

2.4. FILE PERMISSIONS [REQUIRED for AACS FILES]

- ALL FILE actions MUST be AUTHENTICATED.
- AMK[1] MUST NOT be used for FILE access.
- Unique APK[2] MUST be used for each function (READ/WRITE/DELETE/CREATE etc.).
- Directory listing MUST NOT be allowed without prior authentication with AMK or APK.
- FULLY ENCIPHERED COMMUNICATIONS MUST be used for all FILE data access.

2.5. RANDOM UID [RECOMMENDED]

Enable RANDOM UID mode.

- NON-DIVERSIFIED authentication CRYPTO KEY to request actual UID must be unique to this function [REQUIR*ED*]

[1] AMK – Application Master Key

[2] APK – Application Key

3 Section 3 - Deployment Detail

ENCRYPTION STANDARD

Single DES encryption is not considered secure as it can be brute forced on consumer grade hardware and although 3DES is vulnerable to Meet In The Middle (MITM)3[3] attack, a very large number of steps is required which may make the actual deployment of a successful attack impractical. However, as 3DES has been successfully attacked in TLS environments[4] it may be the writing on the wall for this standard, so switching to AES is advisable.

Regardless of which encryption standard is chosen, the 'entropy' of the KEYs is vital. Ideally, a good RNG (Random Number Generator) should be used to create the KEYs[5]. Keys must be protected and stored onsite in a secure manner.

KEY DIVERSIFICATION

KEY DIVERSIFICATION ensures that in the event of a key compromise, only the corresponding TOKEN/FILE is exposed, and the same key cannot be used to access other assets.

It is vital that the diversification MASTER KEYS be kept in a strictly controlled environment, and, if possible, should be generated by a process involving more than one person and obfuscated from all participants – i.e., the actual keys are never revealed to the creators, but generated by combining their individual "secrets" in some non-transparent manner.

If possible, a SAM (Secure Access Module) should be used to store and generate KEYS in the diversification system as well as to store KEYS in the AACS reader itself, and there should be a mechanism for "rolling" keys and 'blacklisting' both specific TOKENS and whole SAMs and or MASTER KEYS.

AID (APPLICATION ID) PERMISSIONS

Each AID, including the "MASTER" (00 00 00) has a number of associated permissions including creating and deleting AIDs, as well as being able to view details such as KEY version numbers.

It is important to allow AIDs to be created without authentication to allow 3rd party applications such as Vending or Single-Sign-On to co-exist with the AACS application without the need to divulge sensitivemasterkeystothose3rd parties.

Access to KEY version numbers must be allowed prior to authentication to enable diversification schemes that include KEY version number.

FILE PERMISSIONS

Within each AID, permissions can be granted for access to the files within it (see image 1 for a generic token AID architecture). AACS AIDs should not divulge any information about their files except after authentication, but 3rd party applications with their own AIDS (e.g. Vending, Single-Sign-On etc.) can operate according to their own rules and can be disregarded with reference to this document.

[3] https://en.wikipedia.org/wiki/Meet-in-the-middle_attack
[4] https://sweet32.info/
[5] https://en.wikipedia.org/wiki/Entropy_(computing)

To avoid unnecessary distribution of the AMK, it must not be used for FILE access and should only exist in the card issuance environment.

See image 1 for an example of a token architecture and file permissions.

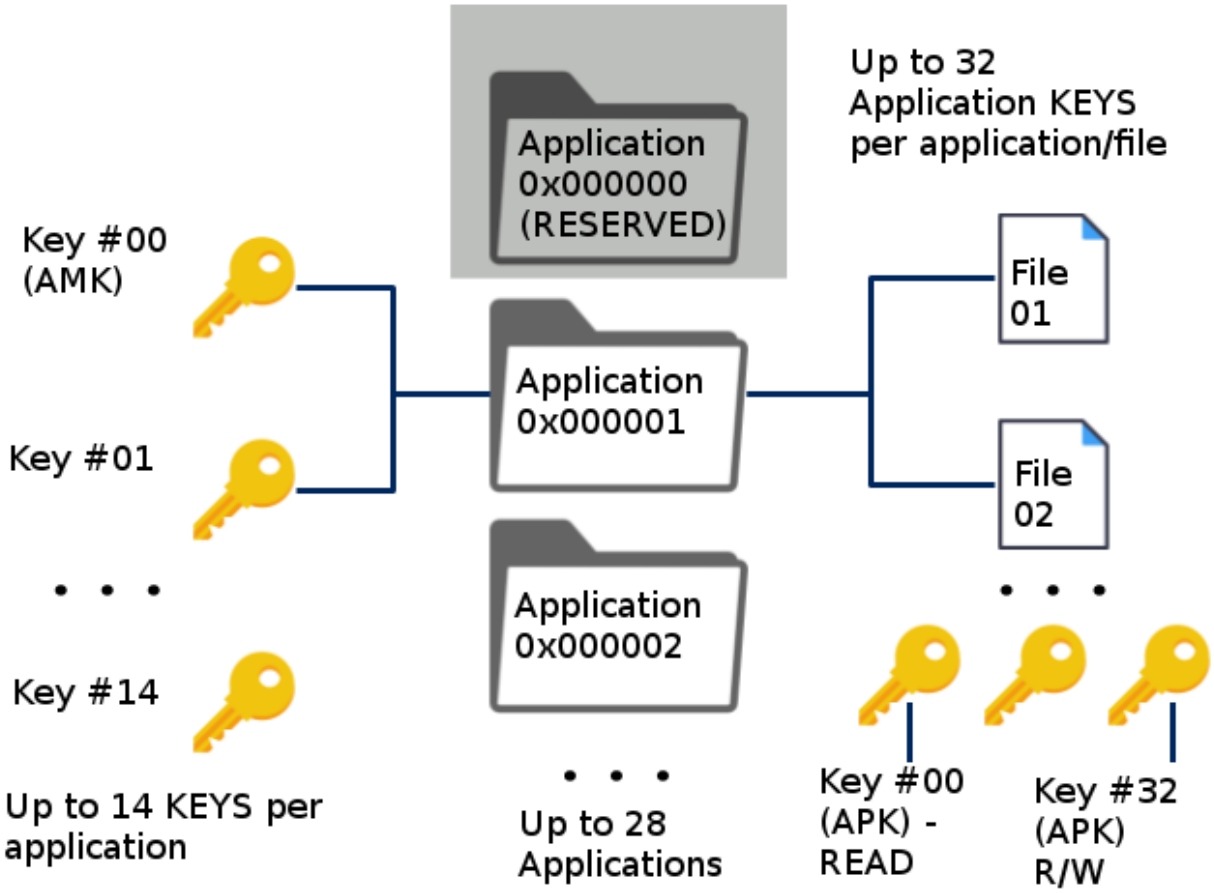

Image 1: Example of a token architecture and file permissions

To avoid misuse of KEYs, each function (READ/WRITE/DELETE etc.) should have its own unique KEY and those should only be present on devices that require the ability to perform that function, e.g. a READER that only needs READ access to one specific AID/FILE should not have a WRITE KEY for that AID/FILE stored in it.

RANDOM UID

Every DESFire token comes with a unique UID which is factory set by the manufacturer and consists of 7-byte (14 HEX digit) fixed value.

RANDOM UID mode hides the actual token UID until authentication has been performed at which point the actual UID can be requested. This prevents identification of a user via correlation with an observed UID as well as hiding part of the information required to DIVERSIFY the cryptographic keys used to access any stored data. Note that use of RANDOM UID precludes the use of non- authenticated functions such as “follow-me printing” that rely on UID alone.

Glossary

AACS - Automated Access Control System

In the fields of physical security and information security, access control (AC) is the selective restriction of access to a place or other resource and an Automated AC System does this by means of an automated check of user supplied credentials before granting access.

AID - Application ID

The unique (to that token/system) 3-byte value that identifies a DESFire application.

AES - Advanced Encryption Standard

A specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001. Also known as Rijndael.

AMK - Application Master Key

The cryptographic key used to authenticate to a specific application.

APK - Application Key

Cryptographic key used within applications to control access to file functions (e.g., READ, WRITE, CHANGE etc.).

CPA - Commercial Product Assurance
A UK initiative managed by NCSC, which evaluates commercial off-the-shelf products, and their developers, against published security and development standards.

DES/3DES - Data Encryption Standard

A symmetric-key algorithm for the encryption of electronic data, now superseded by AES.

ENTROPY - Level of randomness

In information theory, entropy is the measure of uncertainty associated with a random variable. In cryptography, this means how random the variable (usually a KEY) is.

IC - Integrated Circuit

A set of electronic circuits on one small flat piece (or "chip") of semiconductor material, normally silicon.

NCSC - National Cyber Security Centre

The NCSC was set up to help protect UK critical services from cyber-attacks, manage major incidents, and improve the underlying security of the UK Internet through technological improvement and advice to citizens and organisations.

SAM - Secure Access Module
A system consisting of a microcontroller and a reader IC to communicate with an external system and provide secure cryptographic functions.

TLS - Transport Layer Security
Cryptographic protocols designed to provide communications security over a computer network.

UID - Unique Identifier

Every NFC chip has a globally unique, manufacturer supplied, read-only identifier that can be read by most NFC readers. In most NFC chips, this UID is 4 or 7 bytes in length. An NFC tag’s UID cannot be changed or erased; it is stored in special memory in the NFC chip which does not allow the bits to be changed. They are generally issued sequentially, so a batch of tags purchased at the same time are likely to have sequential UIDs.

References

The following documents provide more detailed reference material:

- AN10969 - System level security measures for MIFARE installations
  https://www.nxp.com/docs/en/application-note/AN10969.pdf
- AN10927 - MIFARE product and handling of UIDs
  https://www.nxp.com/docs/en/application-note/AN10927.pdf
- MF3ICDx21_41_81 - MIFARE DESFire EV1 contactless multi-application IC
  https://www.nxp.com/docs/en/data-sheet/MF3ICDX21_41_81_SDS.pdf
- AN10922 - Symmetric key diversifications
  https://www.nxp.com/docs/en/application-note/AN10922.pdf
- AN10975 – Mifare SAM AV2 Documentation and Sampling
  https://www.nxp.com/docs/en/application-note/AN10975.pdf