National Protective Security Authority

User Guide on DESFire EV2 Token Deployment

(EIS0009)

Disclaimer

This document has been prepared by the National Protective Security Authority (NPSA). This document is provided on an information basis only, and whilst NPSA has used all reasonable care in producing it, NPSA provides no warranty as to its accuracy or completeness. To the fullest extent permitted by law, NPSA accepts no liability whatsoever for any expense, liability, loss, damage, claim, or proceedings incurred or arising as a result of any error or omission in the document or arising from any person acting, refraining from acting, relying upon or otherwise using the document. You should make your own judgment with regard to the use of this document and seek independent professional advice on your particular circumstances.

Freedom of Information Act (FOIA)

This information is supplied in confidence and may not be disclosed other than to the agreed readership, without prior reference to NPSA. Within the UK, this material is exempt from disclosure under the relevant Freedom of Information Acts and may be subject to exemption under the Environmental Information Regulations and the Data Protection Act 1998.

Contents

1 Introduction

1.1 What is MIFARE DESFire EV2 and what is it used for?

MIFARE DESFire EV2 forms part of the wider MIFARE range of technologies. MIFARE is a contactless token technology used predominantly for providing access control as part of an Automatic Access Control System (AACS).

MIFARE DESFire EV2 is considered an appropriate option for applications that are required to transmit data fast and in a highly secured manner.
The main target applications for MIFARE DESFire EV2 are:

Automatic access control systems

Identity cards

Closed-loop micropayment

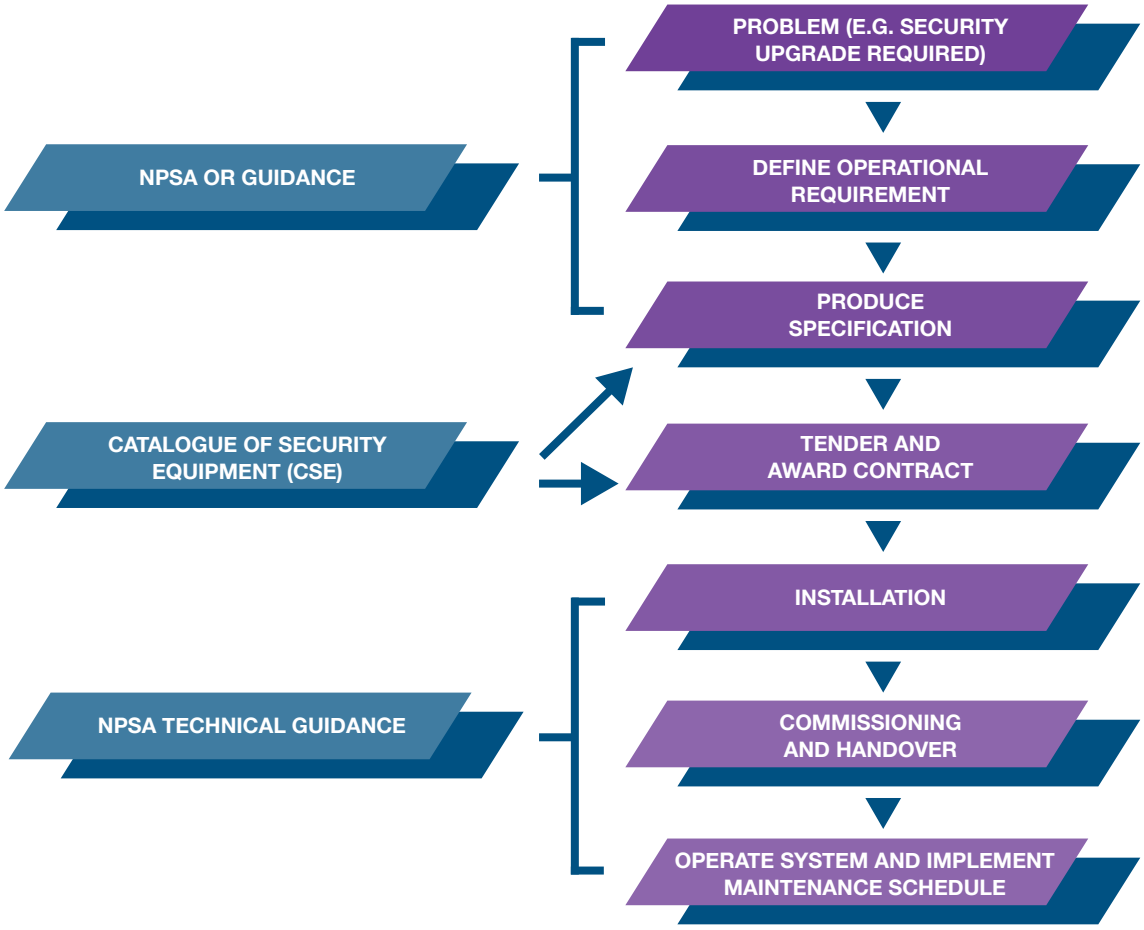
Public transportation

This document provides a brief overview of NXP MIFARE DESFire EV2 tokens in an Automatic Access Control System. It describes both the benefits and the considerations to be aware of when deploying this technology on a site or across multiple sites. DESFire EV2 tokens offer users a secure, flexible and virtual card technology solution. However, care must be taken during card configuration. If this is not done appropriately, it can compromise the level of security afforded.

The aim of this guide is to ensure that only those configurations that provide highly secure operations are selected.

1.2 Steps for Deployment

The successful specification of any security system and subsequent technology, is dependent upon the development of a detailed Operational Requirements (OR). The following diagram illustrates the stages prior to deployment.

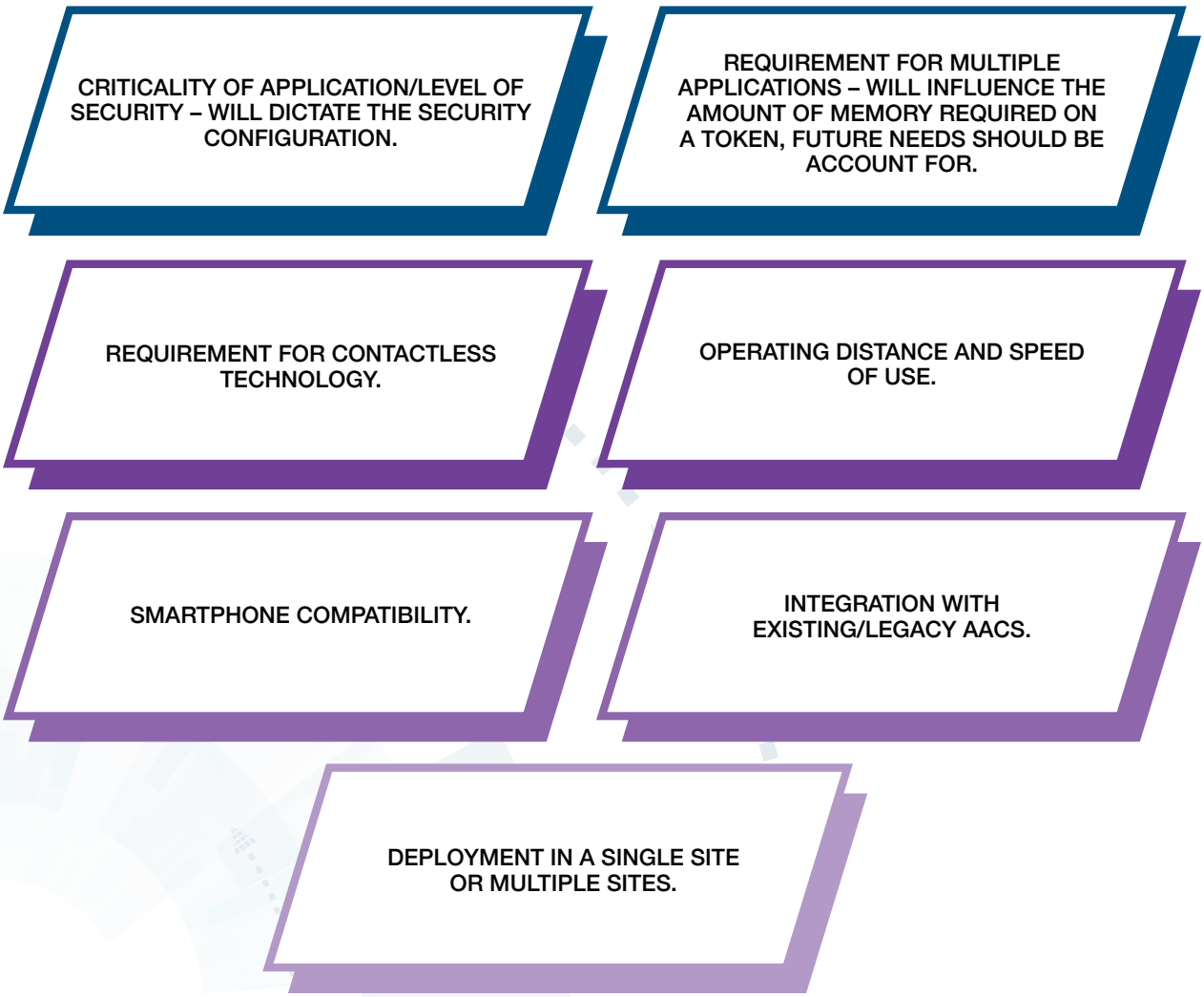


2 Selection

2.1 Factors

Several factors play a role in determining the suitability and design of an AACS and corresponding token technology. They relate to the operational requirements as well as the local circumstances in which the AACS will operate.

Selecting and developing a token architecture is a complex and costly process. However, the benefits of seeking advice and selecting an appropriate solution far outweigh the difficulties and problems associated with changing an incorrectly deployed token family. Before selecting a specific token technology for AACS, a detailed operational requirement (OR) needs to be produced. Some of the factors that ought to be considered include



3 Token Deployment

3.1 DESFire EV2 Features

The DESFire EV2 token has several advantages compared to its predecessors, such as improved range, unlimited number of applications on the card, larger storage, virtual card architecture and improved security. Prior to describing how these tokens should be deployed in any application, this sub-section details the unique features of the DESFire EV2 token.

RANDOM UNIQUE IDENTIFIER (UID)

- Each DESFire token has a unique number or data sequence, referred to as UID. In the case of DESFIRE EV2, this UID is longer and more secured than prior technologies.
- The Random UID feature hides the actual token UID until authentication has been performed.
- Random UID is the most secured configuration, however, in practice it is not always preferred as it can limit third party applications such as printing, if printing is done in a legacy manner i.e. utilising the UID of the card without authentication. Random UID prevents identification of an individual and hides part of the information required to access any stored data.

SECURITY AND ENCRYPTION

- DESFire EV2 has a number of built-in hardware encryption options. AES (Advanced Encryption Standard) 128 or 256 algorithms are the preferred options.
- AES is a symmetric encryption algorithm, which operates on blocks of data of a fixed size of 128 bits or 256 bits. Because of its high degree of security this encryption algorithm is used by governments worldwide to encrypt top secret or sensitive data.

APPLICATION ID (AID) PERMISSIONS

- Each DESFire application has a unique identification or AID. AIDs have a number of associated permissions including creating and deleting AIDs, as well as being able to view details such as key version numbers.
- AACS AIDs should not divulge any information about their files except after authentication, but third party applications with their own AIDs (e.g. vending, single-sign-on etc.) can operate according to their own rules. This hides part of the information required to access any stored data.

VIRTUALISATION

- The DESFire EV2 introduced virtual card architecture. This allows the token's credentials to be stored securely onto a mobile phone or similar devices. As a result, users can utilise their mobile phones to unlock AACS doors. This virtual architecture also allows users to have multiple credentials on their mobile phone allowing for greater flexibility and ease of use.
- It should be noted that the virtual tokens should be configured to mandate authentication before this selection can occur. This prevents an attacker being able to electronically profile the whole range of virtual tokens on the mobile phone.

3.2 Deployment of DESFire EV2

This sub-section details the features that must be adopted when specifying and deploying DESFire EV2 tokens on a site.

KEY DIVERSIFICATION

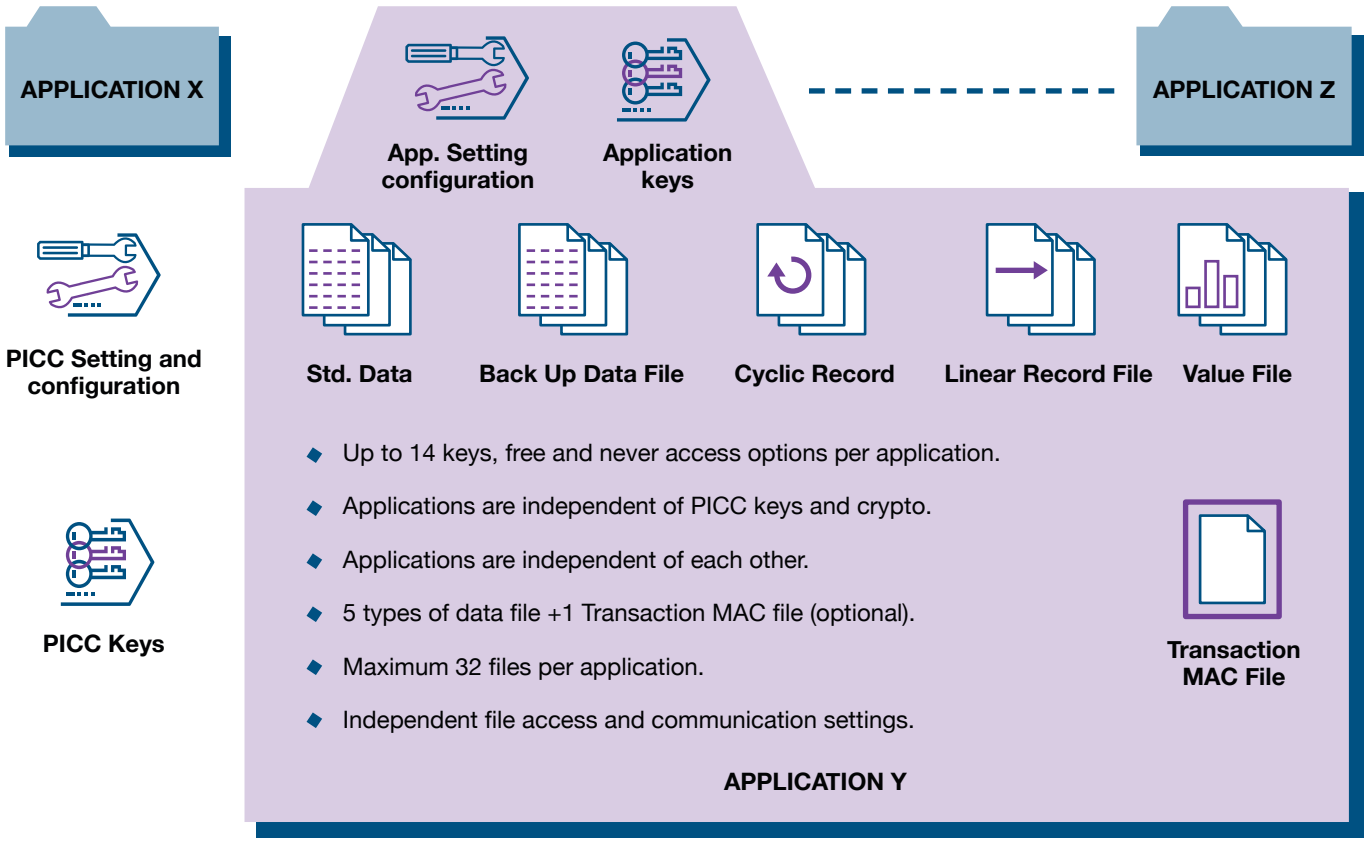
- Under key diversification, every card has its own unique keyset. This means that in the event of a key compromise only the corresponding token/file is exposed.
- The number of diversified keys should be based on the risk assessment of the organisation.
- Master keys must be kept in a strictly controlled environment, and if possible, should be generated by a process involving more than one individual and hidden from all participants.
- In a high security environment, secure access modules (SAMs) should be used to store and generate keys in the diversification system as well as to store keys in the AACS reader itself.

SECURE ACCESS MODULE (SAM)

- As noted in the previous section, in high security environment applications (banking or government), SAMs should be used to securely store and generate cryptographic keys.
- A SAM is a hardware security access module which is either produced physically as a microchip wafer or in the format of a mobile phone SIM card.
- This SIM card can then be installed into a card reader or other hardware units on the secure side of an access controlled door.
- The purpose of this unit is to store static keys or information in a card reader. When a DESFire EV2 token is presented to this, it aids in the mutual authentication process. This is to determine if the token is valid. Once validated, data on the token can be accessed or read.
- DESFire EV2 offers an enhanced security level with a Common Criteria certification of EAL5+. This is better than previous versions of DESFire certified to EAL4.

3.3 Card Architecture

This section details the card architecture for a DESFire EV2 token. The image below highlights the DESFire EV2 architecture showing an example of a particular application. A feature of this architecture is that it can support multiple and unlimited applications. The applications are only limited in terms of card memory. Each application can be used by a client to address various requirements, these may include cashless vending, office printing, etc. Applications can also be offered to third parties to program without having to share the master key.



3.4 Token Security Additional Enhancements

In addition to defining a good and robust token architecture, considerations should be made as to the communication method between the token and the RFID reader. This is to ensure the data exchange cannot be compromised. In this respect

1. **Secure Messaging** – ensures the data exchanged between the token and RFID reader is transmitted accurately and successfully. These transactions need to be enciphered so they are not simply 'plain text', using as a minimum AES128. Use authentication keys for all transactions.

2. **Proximity Checking** – this feature avoids compromises in the security of the AACS by preventing 'relay attacks'. In this type of attack an adversary relays communication between a reader and a card that is not present at the reader. Proximity checking is done by timing a specific sequence of card commands. DESFire EV2 cards support proximity checking.

4 Deployment of Encryption Keys

An important aspect of configuring an AACS is the secure deployment of the encryption keys around the system. Encryption keys are necessary to decipher encoded messages that hide plain text.

This aspect should be planned from the start of any AACS deployment as the requirements can be extensive and complex, otherwise, the efficacy of the AACS system might be compromised. There are several methods to deliver encryption keys:

MANUAL KEY ENTRY

- Encryption keys are manually entered into the access control system or door controllers. Keys are typically distributed around the AACS via unencrypted means.
- Not encouraged as the security of how keys are generated, handled, deployed and then stored cannot be guaranteed easily.

ENCIPHERED CONFIGURATION TOKEN

- Use to hold and deploy the AACS encryption keys.
- A public transport key is deployed into the AACS by the vendor which then allows the AACS token encryption keys to be electronically deployed into the AACS and changing the public transport key to a ‘private’ transport key.
- This method converts the AACS keys into a coded form.
- Enciphered keys are then stored onto an RFID token, preventing the encryption keys from being read as plain text.

USE OF PUBLIC KEY INFRASTRUCTURE (PKI) TYPE METHODS

- Key deployment over wi-fi, Bluetooth or via an Open Supervised Device Protocol (OSDP).
- Relatively convenient and simple way of delivering the token encryption keys.
- However, they typically use enciphered key packets to transmit the data in the first place and require a notional transport key infrastructure.
- Suitable if deployed correctly, assessment is required to ensure any encryption keys are not exposed and compromised.

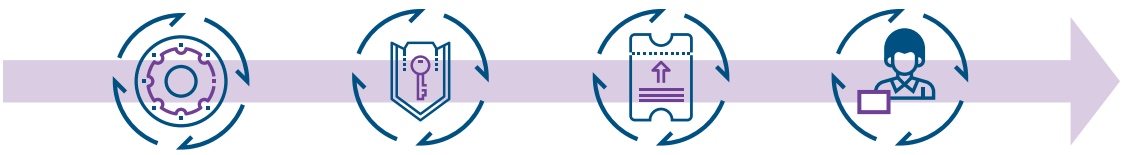
USE OF A HARDWARE SECURITY MODULE (HSM)

- Pre-configured to store the enciphered keys electronically. This device can be removable (like a SIM card) or permanently fitted to the AACS system.
- Each variant requires a complex process of pre- personalising to the AACS.
- Very secure way of key deployment and storage.
- The key package is enciphered from ‘production’ through to ‘transport’ and final HSM module.
- If the HSM is removable, this is a good method of key removal at the end of service life or key rotation.

5 Commissioning

When a token is procured, consideration should be given to who will be responsible for configuring the tokens. The process of configuring a token is called ‘personalisation’. ‘Pre-personalisation’ is the electronic programming of the token’s architecture, and final ‘personalisation’ is typically the application of printing personal details onto the token. The latter is normally conducted by the respective pass issuing office/department.

The process of procuring a token from token programming to printing is illustrated below.



Pre-Personalisation

Conducted by the vendor or by the issuing office.

Structure of the token, applications and files personalised.

Transportation

Encrypted transport keys to AES128 at a minimum.

Key Personalisation

Central secure repository of keys.

Enciphered keys.

Method of programming the encryption keys into the tokens.

Static vs Diverse keys.

Graphic Overlay

Printing Personal Details onto Token.

The pre-personalisation process can either be conducted on premise (programmed locally by the pass issuing department) or undertaken off-site by a vendor. The table below outlines the considerations for each option.

Pre-personalisation	***On premise***	***By a vendor***
Advantages	Allows the user to ensure that the whole process is conducted in a secure and robust manner – from key generation through to token encoding. This process generally advocates good 'key management' and information handling techniques.	Procurement is relatively simple and cost effective. Little tooling is necessary by the end user. The upfront 'pre-personalisation' is completed by an external vendor for a known cost.
Disadvantages	The process is complex, costly and requires expensive tooling and token design to administer.	The tokens security posture are determined by the manufacturer and the assurance processes of the supply chain. Assurances would need to be continually made as to the ongoing efficacy of how the key material is handled, stored and processed, as failure in this aspect could significantly compromise a whole token estate.

6 Through and End of life

Throughout the life of your token, keys may be compromised or lost. It is worth considering having ‘spare’ keys built into the token when it is produced. This is know as key rollover.

At any time, only 1 key is valid. However, if a key is suspected to have been compromised (likely to be via a programming card being lost) the key in the readers can be updated easily (they are on site and this is a relatively easy task) and the readers can be configured to read ‘key 2’ on the tokens. This allows security to be restored without having to change all the tokens.

At the end of life of the AACS, considerations should be made for the secure removal of the token keys.

- **Keys Location** - Typically, they can be held in either the RFID reader or within the AACS system door controllers. An assessment should be conducted as to how to remove or securely destroy them when the equipment is redundant.

- **Maintenance and Repairs** - An assessment should be made to determine if any ‘live’ encryption keys are contained within the faulty equipment, and if there is any assurance around their safe destruction or removal before they leave the premises. Occasionally, due to the nature of the fault within the equipment, it is not possible to confirm the successful removal of the token encryption keys and the equipment should be destroyed in a secure manner.

7 Summary

This guidance document has provided an overview of the features and the deploying methodology of a MIFARE DESFire EV2 token, as part of an Automatic Access Control Systems. The content can be summarised as follows:

Key Points

1. Operational requirements should guide the selection and specification of an AACS and corresponding token technology

2. The security level is subject to the appropriate configuration of the key architecture and token features supported by a specific token technology

3. Key deployment should be planned from the start of any AACS deployment as the requirements can be extensive and complex

4. Token pre-personalisation can either be conducted by the vendor or in house depending on the enterprise's requirements and priorities

5. Consideration should be given to the removal of encryption keys during AACS maintenance, repairs or end of life of the AACS equipment

Further Information

Specialist advice and supporting guidance documents are available through the relevant NPSA sector adviser, or from the NPSA website **www.npsa.gov.uk.** Further reading is available on the following related subjects:

- NPSA Guide to Producing Operational Requirements for Security Measures
- NPSA Catalogue of Security Equipment
- NPSA Automatic Access Control Token – Visual Design Guide
- NPSA Assured Automatic Access Control Systems
- NPSA Guidance document AACS vulnerabilities In the MIFARE Classic Smartcard
- NPSA Token and Reader Procurement Guide

8 Appendix A: High Security Token Checklist

Sl Nº	DESCRIPTION	MUST HAVES	OPTIONAL
1	Organisational study of security and other token application requirements	✓	
2	Avoiding use of token Card Serial Number	✓	
3	Secure transportation of token keys	✓	

Sl Nº	SETTING UP TOKEN	MUST HAVES	OPTIONAL
4	Diversified keys	✓	
5	Key roll-over		✓
6	Provision for key roll-over		✓
7	Site ownership/ Storage of keys	✓