Les badges NFC et la sécurité

Thierry Didi,
Avril 2025
Tidiwi

La technologie NFC (Near Field Communication) est aujourd'hui utilisée dans un grand nombre de domaines tels que le transport, le paiement électronique, le contrôle d'accès dans les hôtels, les immeubles de bureaux ou résidentiels … Les badges les plus utilisés sont ceux de la famille MiFare™[(1)], conçus par NXP Semiconductors. Ce document s'intéresse en particulier à ces badges qui sont conformes aux spécifications ISO/IEC 14443 type A.

L'idée d'approfondir ce sujet m'est venue à la lecture d'un post sur LinkedIn. L'auteur décrit comment il a pu observer lors d'un séjour dans un hôtel que les tags Mifare Ultralight utilisés pour gérer l'accès aux chambres étaient déployés en dehors de recommandations de sécurité normales, sans que le gérant de l'hôtel n'en soit conscient. En conséquence, cet hôtel pourrait être victime de vols dans les chambres, de la part d'anciens résidents ou d'employés. En lisant d'autres posts, il m'est apparu que ce type de mauvais déploiements, souvent pour simplifier la configuration et la gestion des badges, était beaucoup plus fréquent que ce à quoi on pourrait s'attendre.

Il existe plusieurs familles de badges Mifare™ (Mifare™ Ultralight™, Mifare™ Classic™, Mifare™ Plus™, Mifare™ Desfire™), et plusieurs générations dans chacune de ces familles (EV1, EV2 ..) et je me suis demandé quels tags étaient utilisés dans quelles applications, et aussi comment ils étaient utilisés. La question sous-jacente était de savoir si ces tags offraient un niveau de sécurité adapté aux ressources qu'ils doivent protéger.

Ce document décrit l'état actuel de mes recherches sur ce sujet. Notamment il décrit :

- Les risques associés à l'utilisation de badges NFC.
- Les attaques contre les badges NFC.
- Les différents scénarios d'attaque.
- Les outils à la disposition des attaquants.
- Les familles de badges Mifare, en se focalisant sur la sécurité.
- Les recommandations.

Ce document ne s'intéresse pas à la sécurisation des lecteurs de badges qui fera l'objet d'une autre étude plus largement dédiée à la sécurisation de l'IOT (« Internet Of Things »).

Il s'adresse aux intégrateurs ou aux utilisateurs de solutions intégrant des badges NFC pour qu'ils puissent facilement s'y retrouver dans les différentes familles de badges Mifare, tout en étant informés des risques associés à leurs choix.

En remarque préliminaire, je tiens à préciser que les descriptions, recommandations, remarques contenues dans ce document n'engagent que moi et n'engagent en aucun cas NXP Semiconductors et que je n'ai aucune affiliation avec cette société. Ce document est donc basé, au-delà de mon expérience personnelle, sur :

[1] Les Marques Mifare™, Mifare™ Classic™, Mifare™ Plus™, Mifare™ Ultralight™, Mifare™ Desfire™ sont des marques déposées par NXP Semiconductors.

- Une recherche sur les différents outils (hardware et/ou Software) utilisés dans les tests d'intrusion (« Penetration tests » ou « Pentests ») et disponibles de manière entièrement légale.
- Une recherche sur les différentes attaques connues concernant la technologie NFC.
- L'analyse des datasheets publics des badges Mifare.
- Des expérimentations avec des outils de « tests de Pénétration » sur des badges que j'ai en ma possession.

Si vous avez des commentaires, remarques ou des questions, n'hésitez pas à me contacter par email pour m'en faire part : thierry.didi [at] tidiwi [dot] com

Introduction

Avant de rentrer dans les détails des technologies utilisées dans le NFC, il est utile de préciser quelques éléments :

- Dans ce document, on ne s'intéresse qu'aux badges NFC passifs, c'est-à-dire aux badges qui ne sont pas équipés d'une batterie.

- Un badge NFC contient en premier lieu un identifiant unique (UID codé sur 7 octets) ou « non unique » (NUID[2] codé sur 4 octets) qui permet d'identifier le badge, et par extension le porteur de ce badge. Cet identifiant est inscrit par le fabricant du badge dans une zone de la mémoire qui ne peut pas être modifiée. On pourrait donc légitimement considérer qu'on a identifié un utilisateur une fois qu'on a lu l'UID/NUID de son badge, et exécuter les actions demandées par l'utilisateur : par exemple, ouvrir une porte d'immeuble. Mais c'est en fait une très mauvaise idée car il existe des badges très faciles à se procurer, de manière entièrement légale, qui permettent de cloner cet UID/NUID.

- Un badge NFC est essentiellement composé d'une mémoire de stockage non volatile (EEPROM) dans laquelle on peut stocker des informations qui sont conservées quand le badge n'est pas alimenté. La taille de cette mémoire est assez petite et varie en fonction des familles de badges entre une centaine d'octets et quelques milliers d'octets pour les badges les plus puissants.

- Au-delà de cette mémoire, un badge contient une antenne et des composants électroniques qui lui permettent de générer une tension d'alimentation quand il se trouve placé dans un champ électromagnétique généré par un « lecteur » et d'exécuter des opérations de cryptographie. Une fois qu'un badge a été « réveillé » par un lecteur, le lecteur et le badge échangent quelques messages pour s'authentifier mutuellement, puis le lecteur peut déclencher des actions sur son environnement (ouvrir une porte …), et/ou modifier les données stockées dans le badge (décrémenter le solde d'un portefeuille électronique, d'un ticket de transport …).

- La technologie NFC est une technologie « Radio Fréquence » qui fonctionne à 13,56MHz. La distance à laquelle un badge peut être lu par un lecteur est « normalement » de l'ordre d'une dizaine de centimètres. Mais un utilisateur malveillant pourrait utiliser un lecteur dont la puissance de sortie serait plus importante que la puissance autorisée, et équipé d'une antenne de très grande taille (dans un sac à dos par exemple) : dans ce cas, ce lecteur pourrait communiquer avec un badge qui se trouverait à plus d'un mètre de distance[3].

- Ce document s'intéresse essentiellement aux badges de NXP, même si la plupart des problématiques identifiées peuvent être facilement transposées à d'autres badges NFC. Ces badges sont utilisés dans un grand nombre d'applications de contrôle d'accès dans les entreprises, dans les immeubles résidentiels, dans les transports, dans les projets de ville

[2] Les identifiants NUIDs codés sur 4 octets peuvent être réutilisés par le fabricant du badge pour identifier plusieurs badges. Cependant, la probabilité d'avoir en sa possession deux badges ayant le même NUID est très faible, même si elle n'est pas nulle.

[3] https://www.rfidfuture.com/fr/rfid-read-range.html

intelligente … mais ils ne sont pas utilisés (à ma connaissance) dans les cartes bancaires, dans les Passes Navigo parisiens ou dans les passeports électroniques.

- Les acronymes et définitions suivants sont utilisés dans ce document :

    - **AES** : est l'algorithme de chiffrement symétrique qui est actuellement le plus utilisé dans tous les systèmes visant à assurer la confidentialité ou l'intégrité des données. On parle d'AES-128 pour indiquer qu'on utilise cet algorithme avec des clés de chiffrement de 128 bits.
    - **3DES** : est un algorithme de chiffrement symétrique qui est aussi très utilisé, mais moins robuste qu'AES. Il est progressivement remplacé par l'AES.
    - **CMAC** : est un algorithme basé sur AES qui permet de générer la signature d'un message à partir du contenu du message et d'une clé AES. Lors d'une communication, cette signature est calculée par l'émetteur, ajoutée à la fin du message et transmise au récepteur. Le récepteur recalcule la signature du message reçu, et vérifie qu'il obtient la même signature que celle calculée par l'émetteur. Si ce n'est pas le cas, cela indique que le message a été modifié par un acteur malveillant (ou par une erreur de transmission) entre son émission et sa réception. Dans ce cas il ignore le message.
    - **ECDSA** : Elliptic Curve Digital Signature Algorithm. Cet algorithme permet de générer une signature d'un message ou d'un fichier en utilisant une clé privée connue uniquement du signataire. Un destinataire pourra vérifier cette signature à partir d'une clé publique connue de tous.
    - **UID/NUID** : Un identifiant unique (UID) ou non unique (NUID) qui est stocké dans le badge par le fabricant du badge. Cet identifiant peut être codé sur 4 octets (NUID) ou 7 octets (UID) suivant les badges.
    - **Attaquant** : un utilisateur malveillant qui cherche à détourner l'usage nominal du système.
    - **Lecteur NFC** : un terminal qui dialogue avec un badge NFC (par exemple, un terminal de paiement dans un restaurant, ou un lecteur de badge à l'entrée d'un immeuble). Le terme « lecteur » est utilisé abusivement car ce terminal sera souvent aussi capable d'écrire des informations dans la mémoire du badge.
    - **Attaques par canal auxiliaire (« Side Channel Attacks »)** : Ces attaques permettent de découvrir des clés de chiffrement à partir du temps que met un badge pour répondre à une requête (« *Timing Attacks* »), ou en analysant sa consommation de courant pendant une opération de chiffrement (« *Power Analysis Attacks* »). A première vue, ces attaques semblent nécessiter de mettre en œuvre des appareils complexes mais il existe pourtant une attaque de type *Timing Attack* qui permet de retrouver les clés d'un badge MiFare Classic à partir d'un simple PC en analysant le temps que met un tag à répondre à certains messages [4].

[4] https://www.sidechannel.blog/en/mifare-classic-2/

Les risques associés à l'utilisation de badges NFC

Avant de déployer un système basé sur des badges NFC, il est important de connaître les risques auxquels on s'expose, et d'évaluer si ces risques posent un problème pour le système. Ces risques sont les suivants :

- **Usurpation d'identité**
  Un attaquant se fait passer pour un utilisateur légitime pour avoir accès à un bâtiment ou à un service, ou pour voyager en faisant payer un autre utilisateur …

- **Modification du contenu d'un badge**
  Un attaquant modifie par exemple le nombre de voyages qu'il peut faire dans un système de transport.

- **Accès à du contenu privé stocké sur la carte**
  Comme le nom ou l'adresse d'un utilisateur par exemple.

- **Denial Of Service**
  Un utilisateur légitime n'a plus accès à une ressource à laquelle il devrait avoir accès. Ceci peut avoir un effet négatif sur l'opinion que l'utilisateur final a sur la fiabilité du système déployé, jusqu'à l'amener à s'en détourner.

- **Suivi des personnes**
  En enregistrant les dates et heures de présentation de l'UID d'un badge sur différents lecteurs, l'opérateur du système peut suivre les mouvements du porteur du badge.

Les attaques contre les badges NFC

Les attaques les plus courantes contre les badges NFC sont les suivantes :

- **Ecoute passive (« Eavesdropping »)** : Un attaquant peut espionner la communication entre un badge et un lecteur.
  Cette attaque est très simple puisqu'il suffit d'installer un récepteur radio à proximité du tag et du lecteur. On peut aussi utiliser un dispositif composé de deux antennes NFC reliées par un câble (« range extender ») pour dupliquer la communication entre le badge et un lecteur, et espionner la conversation à distance. L'attaquant pourrait alors :

    - Lire l'UID du badge et usurper ainsi l'identité de son propriétaire dans certains systèmes de contrôle d'accès qui ne se basent que sur l'UID du badge. Même si cette pratique est à proscrire, elle est implémentée dans certains systèmes de contrôle d'accès à faible coût.
    - Avoir accès à des données confidentielles contenues dans le badge si les communications ne sont pas chiffrées sur l'interface radio entre le badge et le lecteur. Par exemple, les badges Mifare Classic et Mifare Ultralight ne chiffrent pas les communications sur l'interface radio, alors que les badges Mifare Plus ou Mifare Desfire peuvent le faire.
    - Dans certains cas deviner les clés de chiffrement utilisées pour protéger les données contenues dans le badge.

- **Brouillage** (« jamming »): un attaquant peut brouiller tout ou partie de la communication entre un badge et un lecteur. Cette attaque est aussi très simple à réaliser, même si elle est illégale. Les conséquences peuvent être importantes, notamment si l'utilisateur finit par perdre confiance dans le système installé.

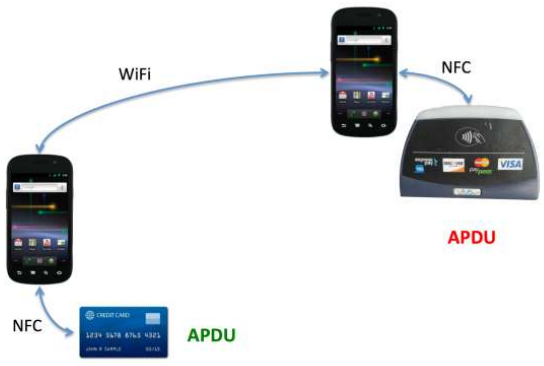
- **Lecture et/ou modification du contenu d'un badge**
  Si le contenu du badge est mal protégé, un attaquant pourrait lire des données confidentielles dans le badge, ou modifier son contenu, par exemple le nombre de voyages qui lui sont autorisés.

- **Attaque de Replay**
  Un attaquant espionne une conversation entre un badge et un lecteur, et « rejoue » cette conversation à partir d'un émulateur. Il pourrait ainsi espionner la communication entre le badge d'un employé et un lecteur de badges à l'entrée d'un immeuble tertiaire, puis rejouer la même séquence en face du lecteur pour accéder à l'immeuble.

- **Clonage de badge**
  Le clonage d'un badge permet d'usurper l'identité d'un utilisateur légitime.

- **Utilisation d'un émulateur**
  Si l'attaquant dispose d'un émulateur (très facile à se procurer dans le commerce, de manière tout à fait légale), il peut faire passer cet émulateur pour un badge authentique, pourvu qu'il ait réussi à lire le contenu du badge authentique.

- **Les attaques par relais** (« Relay Attacks »)
  Cette attaque vise à faire communiquer un badge avec un lecteur qui est très éloigné du badge (par exemple dans un autre pays). Les communications entre le badge et le lecteur sont relayées via un canal de communication à haut débit.



source : https://salmg.net/2018/12/01/intro-to-nfc-payment-relay-attacks/relay

Les données peuvent éventuellement être modifiées pendant le transport. Ce type d'attaque permettrait à un attaquant d'avoir accès à un bâtiment en relayant les données d'un utilisateur légitime qui se trouve dans un café ou dans le train par exemple, ou bien d'exécuter des paiements sans contact en utilisant le badge d'un autre utilisateur. Il existe

une application open source (nfcgate[5]) développée pour étudier la sécurité des applications mobiles qui permet d'analyser, modifier ou relayer les données NFC entre un badge et un serveur distant. Cette application a servi de base à des utilisateurs malveillants pour développer des applications capables d'exécuter des attaques par relais.

- **La retenue du message de fin de transaction**
  Si l'attaquant utilise un relais entre le terminal et son badge, il peut décider de capturer le message de fin de transaction émis par le terminal, et de ne pas le transmettre au badge. Dans ce cas, les données du badge ne seront pas mises à jour car le badge « pensera » que la transaction a été interrompue par le terminal avant d'être définitivement validée. Le terminal pourrait ne pas détecter la fraude puisque, de son point de vue, la transaction a été terminée avec succès. Il pourrait par exemple ouvrir un portique dans un système de transport, alors que le badge de l'utilisateur n'aurait pas été mis à jour.

Les différents scenarios d'attaque

Même si on imagine qu'un attaquant dispose d'outils très sophistiqués et de grandes compétences en informatique pour compromettre un système, ce n'est pas toujours le cas. Les scénarios à envisager sont les suivants.

- Un attaquant attaque son propre badge. Il n'a donc pas de limite de temps, et peut utiliser des outils sophistiqués pour attaquer le badge. Par exemple, il peut essayer de modifier le nombre de titres de transport présents sur son badge, où prolonger ses droits d'accès à une chambre d'hôtel, ou obtenir l'accès à des chambres qui ne sont pas la sienne …

- Un attaquant attaque le badge d'un utilisateur légitime pendant qu'il est assis à côté de lui dans le train ou dans un café par exemple. Le temps n'est pas illimité dans ce cas. Ce scénario correspond par exemple à une attaque de type « Relay Attack », ou au vol de données personnelles. Il peut aussi lire l'UID du badge (ou d'autres données contenues dans le badge) qui lui permettront d'usurper l'identité du propriétaire légitime.

- Un attaquant essaie de compromettre le badge d'un autre utilisateur. La démarche est donc plus compliquée, et passe par l'enregistrement des conversations entre le badge et un lecteur. Plusieurs conversations devront souvent être espionnées pour que l'attaque réussisse, ce qui rend la démarche plus compliquée.

Les outils à la disposition des attaquants

Pour se protéger des attaques contre les badges, il est utile de connaître les outils que pourraient utiliser les attaquants. Cette section s'intéresse aux outils très simples qu'on peut se procurer très facilement et de manière entièrement légale. Au-delà de ces outils, des organisations criminelles ou étatiques pourraient disposer d'outils beaucoup plus élaborés.

[5] https://github.com/nfcgate/nfcgate

- Des badges « Magic » où il est possible de configurer un grand nombre de paramètres qui sont normalement inscrits par le fabricant (UID, Type de badge Mifare Classic 1K, ou Mifare Classic 4K ou Mifare Ultralight …).

- Des logiciels qui permettent de manipuler les données de badges Mifare Classic, de cloner des badges, ou de retrouver les clés de chiffrement utilisées pour protéger les données de ces badges (par exemple l'application Android MCT).

- Des appareils électroniques qui permettent d'émuler, de lire, écrire et attaquer des badges Mifare Classic, Ultralight, Desfire (par exemple Flipper Zero, Chamelon Ultra, Proxmark 3).

- Des prolongateurs d'antenne NFC (« Range Extender ») qui permettent par exemple d'espionner les échanges entre un lecteur et un badge. Ils sont composés de deux antennes NFC reliées par un câble RF. Ces prolongateurs d'antenne peuvent aussi être intégrés discrètement dans un lecteur NFC compromis, afin que le badge présenté sur le lecteur communique avec un lecteur déporté et caché à l'insu de son propriétaire.

- Des logiciels dérivés de logiciels utilisés pour la recherche en cybersécurité qui permettent de relayer les signaux radio entre un lecteur local et un lecteur déporté, qui pourrait même se trouver dans un autre pays.

Les familles de badges Mifare

Au fil du temps, NXP a introduit plusieurs familles de badges compatibles avec (ou basés sur) les standards de la famille ISO/IEC 14443 Type A [6]. Chaque famille répond aux besoins d'un certain niveau de sécurité. Dans chaque famille de tags (Mifare Classic, Mifare Ultralight, Mifare Plus, Mifare Desfire), les spécifications ont évolué et il est recommandé d'utiliser les tags les plus récents dans les nouveaux designs. Cependant, beaucoup de systèmes déployés sont basés sur les versions les plus anciennes, et donc les moins sécurisées, de ces tags. Il est donc utile de connaître les caractéristiques de ces anciennes versions si on s'intéresse à la sécurité des systèmes déployés.

Ces familles sont les suivantes :

- **Mifare Classic** : c'est sans doute le badge le plus utilisé, et qui a fait l'objet d'un grand nombre d'études et d'attaques de sécurité. Il utilise un algorithme de chiffrement propriétaire nommé Crypto1. Mais cet algorithme a été complètement cassé, et il existe aujourd'hui de nombreux logiciels qui permettent de pirater ces badges. Malgré ces défauts, ces badges sont encore largement utilisés pour sécuriser l'accès à des chambres d'hôtel, à l'entrée des immeubles résidentiels ou tertiaires, ou pour autoriser l'accès à des installation publiques (piscines, déchèteries …), ou pour gérer l'accès à des événements. A l'origine, ils étaient aussi largement utilisés dans les transports.
  La spécification des tags Mifare Classic a évolué au fil du temps :
  - **Mifare Classic 1K / Mifare Classic 4K** : les plus anciens
  - **Mifare Classic EV1** : la recommandation actuelle de NXP pour les badges Mifare Classic.

- **Mifare Plus** : les badges Mifare Plus ont été développés comme alternative aux badges Mifare Classic, en offrant une sécurité basée sur l'algorithme AES plutôt que Crypto1. Ils offrent donc une bien meilleure sécurité mais je ne suis pas sûr qu'ils soient largement déployés aujourd'hui. Les évolutions des badges Mifare Plus sont les suivantes :
  - **Mifare Plus SE**
  - **Mifare Plus EV2**

- **Mifare Ultralight** : Ces badges sont souvent utilisés pour sécuriser les accès aux chambres d'hôtel par exemple, ou pour gérer des titres de transport à usage unique ou limité, ou encore pour contrôler l'accès à des événements (concerts …). Il en existe à ce jour 4 versions :

  - **Mifare Ultralight :** Cette version est maintenant obsolète mais largement déployée et sans aucune fonctionnalité de sécurité.
  - **Mifare Ultralight EV1 :** Cette version offre une sécurité « ultra minimaliste » basée sur un mot de passe de 32 bits.
  - **Mifare Ultralight C :** Cette version offre une meilleure sécurité basée sur le chiffrement 3DES.
  - **Mifare Ultralight AES :** Cette version est destinée à remplacer Mifare Ultralight C. Elle utilise l'algorithme de chiffrement AES et intègre de nouvelles fonctionnalités de sécurité.

[6] Tous les tags MIFARE sont compatibles avec ISO/IEC 14443 Part 2 et ISO/IEC 14443 Part 3- Les tags MIFARE Desfire et MIFARE Plus sont compatibles avec ISO/IEC 1443 Part 4 – les tags MIFARE Classic et MIFARE Ultra light sont compatibles avec le protocole MIFARE, qui est basé sur ISO/IEC 14443 Part 3.

- **Mifare Desfire** : ces badges offrent le plus haut niveau de sécurité, et des tailles de mémoire plus importantes que les autres badges. Ils sont destinés aux applications qui requièrent un haut niveau de sécurité, et du fait de leur grande mémoire, ils peuvent être utilisés pour gérer les accès à plusieurs services différents avec un badge unique. Par ailleurs, les systèmes de contrôle d'accès aux bâtiments tertiaires qui utilisaient Mifare Classic migrent progressivement vers Mifare Desfire qui offre un niveau de sécurité beaucoup plus élevé.
    - **Mifare Desfire EV3**

Chacune des familles mentionnées ci-dessus a évolué dans le temps, le plus souvent pour répondre aux failles de sécurité identifiées. Souvent, ces évolutions sont traduites dans le nom commercial du badge en y ajoutant le numéro d'évolution, sous la forme EVx. Par exemple, les tags Mifare Desfire ont existé sous la forme Desfire EV1, puis Desfire EV2, et aujourd'hui Desfire EV3.

Les badges Mifare Classic (MF1S50yyX)

Les badges Mifare Classic sont les plus anciens badges NFC développés par NXP, à l'origine pour les solutions de transport. Ils sont aujourd'hui obsolètes.

Ces badges sont pourtant très largement utilisés dans un grand nombre d'applications de contrôle d'accès (par exemple ces badges sont utilisés en France dans le système VIGIK qui gère l'accès à un grand nombre d'immeubles résidentiels [7]), de transport, mais aussi dans un grand nombre d'application sans aucun lien avec la sécurité. Le niveau de sécurité offert par ces badges est quasiment nul.

- **L'UID/NUID**
  Chaque badge dispose d'un identifiant unique et non modifiable (UID) composé de 7 ou non unique et non modifiable (NUID) composé de 4 octets. Cet identifiant est stocké dans le secteur 0 de la mémoire du badge qui peut être lu par un lecteur NFC sans nécessiter d'authentification préalable.

- **La mémoire**
  Ces badges peuvent avoir une mémoire de 1k octets (Mifare Classic 1K) ou 4k octets (Mifare Classic 4K). La mémoire des badges Mifare Classic 1K est organisée en 16 secteurs de 64 octets chacun. Dans la mesure ou chaque secteur dispose de ses propres clés de contrôle d'accès (voir ci-dessous), on peut imaginer dédier chaque secteur à une application spécifique (par exemple contrôle d'accès, billetterie…).

- **La protection des données**
  Chaque secteur est associé à une paire de clés (KeyA/KeyB) qui servent à protéger les accès en lecture et/ou écriture. Les données de chaque secteur peuvent être protégées par ces clés et grâce à un algorithme de chiffrement symétrique nommé Crypto1. Puisque cet algorithme est un algorithme symétrique, les clés de chiffrement d'un secteur d'un badge doivent être connues par le badge ET par le lecteur pour que ce dernier puisse accéder à ce secteur.
  Il est à noter que les badges sont livrés avec des clés par défaut, connues publiquement. Il est donc essentiel de personnaliser ces clés si on doit stocker des informations confidentielles dans ces secteurs (par exemple, le nombre de passages autorisés dans un portique de métro).

- **Remarques**

  - Il faut utiliser des jeux de clés différents pour chaque badge, pour éviter qu'un badge compromis ne compromette tout le système. Une bonne pratique recommandée par NXP consiste à dériver un jeu de clés pour chaque badge à partir d'une « Master Key » et d'informations fixes telles que l'UID du badge. Cette Master Key ne sera connue que du lecteur et ne sera pas stockée dans le badge.

[7] https://www.vigik.com/presentation-de-vigik

- On pourrait être tenté d'utiliser l'UID d'un badge pour identifier l'utilisateur du badge, ce qui est d'ailleurs fait par de nombreuses applications. Cependant, cette méthode n'est pas du tout sécurisée car il est possible d'acheter des badges (« Magic Cards ») dont le secteur 0 est modifiable : il est donc tout à fait possible et très simple de fabriquer un nouveau badge qui aurait le même UID que le badge d'origine. Cette opération est encore plus simple si on utilise un émulateur de badge.

- Il est à noter qu'il existe aussi des logiciels open source qui peuvent exécuter une attaque par dictionnaire (« dictionary attack ») sur l'algorithme Crypto1 pour retrouver la clé d'un secteur d'un badge Mifare Classic. Ces logiciels testent en premier lieu les clés par défaut, ou bien des clés qui ont été publiées par des individus et qui sont en libre accès sur le web (par exemple dans les fichiers source de l'application MifareClassicTool).

- Il existe aussi d'autres types d'attaques qui se basent aussi sur des failles de l'algorithme Crypto1, notamment l'attaque MFKEY32 (« mfkey32 Attack ») pour retrouver les clés Key A et Key B d'un secteur à partir de la capture et de l'analyse de quelques échanges d'authentifications entre le lecteur et le badge.

- Enfin, on pourrait être tenté de personnaliser uniquement les clés des secteurs que l'on utilise, et de laisser les clés des autres secteurs à leurs valeurs par défaut. Cependant, il existe une attaque (« Nested Attack ») liée à des failles de l'algorithme Crypto 1 qui permet de récupérer les clés de tous les secteurs d'un badge Mifare Classic si on connaît au moins la clé d'un secteur.

Mifare Classic EV1 (MF1S50YYX_V1)

Ces badges sont la version la plus récente des badges Mifare Classic. Leurs fonctionnalités sont essentiellement les mêmes que celles des badges Mifare Classic. La différence la plus notable est la suivante :

- **Preuve d'origine**
  Les tags Mifare Classic EV1 supportent une fonction de vérification de leur origine, qui permet à un lecteur de vérifier qu'il dialogue avec un tag authentique. Cette vérification est basée sur une signature ECDSA générée par NXP à partir d'une clé privée connue uniquement de NXP, et stockée dans le badge. Un lecteur qui dispose de la clé publique correspondante pourra vérifier cette signature.

- **Remarques**
  - Ces badges souffrent donc globalement des mêmes faiblesses que les badges Mifare Classic.
  - La fonction de vérification de l'origine apporte un excellent niveau de protection contre la création de faux badges avec un nouvel UID puisqu'un attaquant ne sera pas capable de générer une signature authentique de ce nouveau badge.
  - Cette fonction n'est évidemment utilisable que si le lecteur la supporte.
  - Elle protège aussi contre le clonage d'un badge sur un badge vierge.
  - En revanche, elle ne protège pas contre le clonage d'un badge existant au moyen d'un émulateur. Cet émulateur pourra espionner la communication entre un lecteur

et badge original au moment de la vérification de sa signature, et ensuite reproduire les réponses du badge authentique quand il se trouvera face au lecteur.

Mifare Plus SE (MF1SEP(H)10x1)

Les badges Mifare Plus SE ont été développés pour corriger les failles de sécurité des badges Mifare Classic et pour permettre de migrer d'une infrastructure Mifare Classic vers une infrastructure plus sécurisée. Notamment, ils utilisent l'algorithme AES-128 pour le chiffrement en remplacement de l'algorithme Crypto1. Ces badges ont été conçus pour être compatibles avec une infrastructure déployée pour des badges Mifare Classic. Ils contiennent donc potentiellement deux jeux de clés : un jeu de clés Crypto 1 et un jeu de clés AES-128. Dès lors ils doivent aussi implémenter l'algorithme Crypto1, et sont donc sujet aux attaques mentionnées plus haut quand ils sont utilisés avec une infrastructure ancienne.

- **L'UID/NUID**
  Chaque badge dispose d'un identifiant unique et non modifiable (UID) composé de 7 ou d'un identifiant Non Unique et non modifiable (NUID) composé de 4 octets.

- **La mémoire**
  L'organisation de la mémoire est la même que l'organisation mémoire des badges Mifare Classic.

- **La protection des données**
  - Lors de l'authentification mutuelle entre le badge et le lecteur, deux clés AES (clés de session) sont générées. Ces clés sont donc différentes à chaque nouvelle session, ce qui protège contre les attaques de Replay. Ces clés seront utilisées pour chiffrer les données sur l'interface Radio, et pour assurer l'intégrité de ces messages.

  - Les secteurs peuvent être protégés par l'algorithme Crypto1 et une paire de clés Key A/Key B comme dans le cas des badges Mifare Classic. Ils peuvent aussi être protégés par l'algorithme AES-128 et une paire de clés AES-128. Les clés AES peuvent être mises à jour.

- **Preuve d'origine**
  Pour se protéger contre le clonage, ces badges offrent une fonction qui permet de prouver que ce sont bien des badges originaux. Cette fonction est basée sur une clé AES spéciale, dédiée à cette fonctionnalité. Le lecteur doit donc connaître cette clé AES pour vérifier l'origine du badge.

- **Remarques**
  Ces badges offrent un excellent niveau de sécurité contre la plupart des attaques, sauf bien sûr s'ils sont utilisés sur une infrastructure ancienne avec des données protégées par l'algorithme Crypto 1.
  Ils sont protégés contre les attaques par Replay.
  Ils restent sujet aux attaques par relais, et au suivi des personnes au moyen de l'UID/NUID du badge.

Mifare Plus EV2 (MF1P(H)x2)

Les badges Mifare Plus EV2 offrent un niveau de sécurité compatible avec les besoins des applications bancaires, ou avec les passeports électroniques.
Les badges Mifare Plus EV2 sont eux aussi compatibles avec les badges Mifare Plus SE et avec les badges Mifare Classic EV1. Par ailleurs, ils disposent de fonctionnalités supplémentaires. Notamment, une fonction de détection de proximité entre la carte et le lecteur, pour se protéger contre les attaques par relais. Enfin les messages échangés entre la carte et le lecteur peuvent être chiffrés au authentifiés grâce à deux clés AES. Ces clés sont générées au moment de l'authentification du badge par le lecteur et sont différentes à chaque session. Cela permet d'être protégé contre les attaques de type replay.

- **UID/NUID**
  - Chaque badge dispose d'un identifiant unique et non modifiable (UID) composé de 7 ou d'un identifiant Non Unique et non modifiable (NUID) composé de 4 octets.
  - Ce badge peut aussi être configuré pour générer un UID « pseudo aléatoire » à chaque connexion. Cette fonctionnalité permet de protéger l'anonymat du porteur : le gestionnaire du système ne pourra pas retracer les déplacements de l'utilisateur en suivant les apparitions de son badge sur les différents lecteurs de son installation.

- **Mémoire**
  La mémoire de ces badges est composée de 2048 ou 4096 octets.

- **La protection des données**
  - Les données du badge peuvent être protégées contre les accès en lecture et/ou écriture. Cette protection peut être réalisée par l'algorithme Crypto1 et une paire de clés KeyA/KeyB si le badge est utilisé face à une infrastructure déployée pour des badges Mifare Classic (donc avec un faible niveau de sécurité), ou par l'algorithme AES-128 avec des clés AES dans le cas d'une infrastructure récente.
  - Protection de l'interface Radio
    Les communications Radio entre le lecteur et le badge peuvent être chiffrées avec l'algorithme AES-128, et leur intégrité peut être vérifiée grâce à l'ajout d'une signature CMAC à la fin de chaque message.
- **Proximity Check**
  Ce badge implémente une fonctionnalité de vérification de proximité, basée sur la mesure précise du temps de transit des messages entre le lecteur et le badge. Cette fonctionnalité doit être initiée par le lecteur (donc il faut que le lecteur la supporte).

- **Preuve d'origine**
  Ces badges offrent deux méthodes pour vérifier qu'ils sont originaux :
  - Une signature ECDSA calculée à l'aide d'une clé privée connue uniquement de NXP. Le lecteur doit connaître la clé publique associée à cette clé privée pour effectuer la vérification de la signature.
  - Une clé AES128 (qui doit donc être connue du badge et du lecteur).

- **Remarques**
  Ces badges permettent de se protéger contre toutes les attaques connues.
  La fonctionnalité « Proximity Check » protège sans doute correctement contre les attaques par relais quand le temps de transit est « important » (quelques millisecondes) entre le

lecteur et le badge. C'est le cas si le lecteur et le badge sont localisés dans des endroits différents, et que le signal est transporté via le réseau WIFI, ou 4G … En revanche, cette fonctionnalité n'offre sans doute pas de protection si le lecteur et le badge sont séparés par à un « *range extender* » car le temps de transit du signal sur ce type de dispositif est très court.

Mifare Ultralight (MF0ICU1)

Ce badge n'est plus référencé sur le site de NXP, mais on peut toujours l'acheter sur internet et il est toujours utilisé dans des applications.

- **UID**
  L'UID de ce badge est composé de 7 octets non modifiables.
- **Mémoire**
  - La mémoire de ce badge contient 48 octets (384 bits) utilisables pour stocker des données. Contrairement aux badges mifare Classic, cette mémoire n'est pas partitionnée en secteurs disjoints.
- **Protection**
  - Aucune authentification n'est requise pour accéder aux données et ce badge ne dispose d'aucune fonctionnalité de chiffrement.
  - Chaque page peut être verrouillée pour devenir « Read Only » de manière irréversible.
- **Remarques**
  - Ce badge ne devrait jamais être utilisé dans des applications de sécurité puisqu'il est très simple de le cloner avec un émulateur, même si j'ai pu vérifier qu'il est actuellement utilisé dans des systèmes d'alarme de maison.

Mifare UltraLight EV1 (MF0ULX1)

- **UID**
  L'UID de ce badge est composé de 7 octets non modifiables.
- **Mémoire**
  La mémoire de ce badge contient 48 octets (384 bits) ou bien 128 octets (1024 bits) utilisables pour stocker les données de l'utilisateur.
- **Protection**
  - Chaque page peut être verrouillée pour devenir « Read Only » de manière irréversible.
  - L'accès à la mémoire (en lecture et/ou écriture) peut être protégé par un mot de passe de 32 bits
  - Il est possible de programmer un nombre maximum d'échecs d'authentification pour se protéger contre les attaques par force brute.
- **Preuve d'origine**
  Ce badge fournit une fonction permettant de s'assurer qu'il s'agit bien d'un badge original et pas d'une copie. Le badge contient une signature ECDSA (Elliptic Curve Digitla Signature Algorithm) générée par NXP à partir d'une clé privée connue seulement de NXP. Cette

signature peut être vérifiée par un lecteur disposant de la clé publique associée à cette clé privée. L'intérêt d'utiliser ce type de signature est que la clé privée utilisée pour générer la signature n'est connue ni du badge ni du lecteur. Cette assurance protège contre la duplication d'un badge original sur un badge vierge, mais pas contre la duplication par un émulateur qui pourrait aussi reproduire la signature du badge original.

- **Autres fonctionnalités**
  Ce badge Implémente 3 « one way counters », c'est-à-dire que ces compteurs ne peuvent qu'être incrémentés par un lecteur, et jamais décrémentés. Ces compteurs peuvent être utilisés pour limiter le nombre de passages dans les moyens de transport par exemple.

- **Remarques**
  Pour accéder aux données du badge, le lecteur commence par envoyer « en clair » le mot de passe du badge pour le déverrouiller … Donc, un attaquant qui écouterait la communication pourrait facilement connaître le mot de passe du badge. La sécurité offerte par ce badge est donc ultra-minimaliste. Certaines implémentations utilisent le même mot de passe pour tous les badges, ce qui s'avère être une hérésie [8]. Quand ce type de badge est utilisé, il faut absolument utiliser des mots de passes différents pour tous les badges, et configurer un nombre maximum d'échecs d'authentification pour se protéger contre les attaques par force brute.

Mifare UltraLight C (MF0ICU2)

Ce badge est similaire au badge Mifare Ultralight EV1, mais il implémente une meilleure protection de la mémoire. Cependant, NXP recommande aujourd'hui d'utiliser plutôt le badge Mifare UltraLight AES (voir ci-dessous) qui implémente un algorithme de chiffrement plus robuste et d'autres fonctionnalités permettant d'améliorer la sécurité.

- **UID**
  L'UID de ce badge est composé de 7 octets non modifiables.
- **Mémoire**
  La mémoire de ce badge contient 144 octets (1152 bits) utilisables pour stocker les données utilisateur.
- **Protection**
  - La mémoire peut être verrouillée (par blocs) pour devenir « Read Only » de manière irréversible.
  - L'accès à la mémoire peut être protégé par une authentification mutuelle : le badge et le lecteur doivent se prouver l'un à l'autre qu'ils connaissent une même clé de chiffrement 3DES. Cet échange est basé sur l'algorithme 3DES en mode CBC (Cipher Block Chaining). Ce mécanisme est plus efficace qu'une protection par mot de passe car la clé de chiffrement qui sera utilisée pour accéder à la mémoire est une clé dérivée de cette clé partagée, qui n'est elle-même jamais transmise en clair sur l'interface NFC.
- **Autres fonctionnalités**

[8] https://www.linkedin.com/posts/stephanelemee_vuln%C3%A9rabilit%C3%A9-s%C3%A9curit%C3%A9-d%C3%A9poussi%C3%A9rage-activity-7127927457076109312-o7m8/?originalSubdomain=fr

  - Ce badge implémente un compteur « one way » qui peut être incrémenté mais jamais décrémenté.
- **Remarques**
  Même si le niveau de sécurité est plus élevé que pour les badges Mifare Ultralight EV1, ce badge n'offre tout de même pas une sécurité absolue. L'algorithme 3DES est considéré comme assez faible, et son implémentation dans les badges, notamment le mode CBC avec un IV (Initialization Vector) nul le rend peu robuste contre des attaques cryptographiques. Mais ce type d'attaque ne semble pas être encore implémenté dans les outils de tests de pénétration.
  Dans tous les cas, il est fortement recommandé d'utiliser des clés de chiffrement uniques pour chaque badge. Dans la mesure où le lecteur doit connaître les clés de tous les badges, il n'est pas envisageable de générer des clés aléatoirement pour chaque badge. On utilisera plutôt une « fonction de dérivation » qui permet de générer une clé unique à partir de l'UID du badge et d'une « Master Key ». Cette Master Key sera connue par l'usine qui personnalise le badge et par le lecteur, mais elle ne sera pas stockée dans le badge.

Mifare UltraLight AES (MF0AES(H)20)

Ce badge est une évolution du badge Mifare Ultralight C. La cryptographie est basée sur l'algorithme AES-128 qui est plus robuste que 3DES. Il ajoute aussi l'authentification des messages sur l'interface NFC pour protéger le système contre les modifications des messages entre le lecteur et le badge (attaques « Man In The Middle »).

- **UID**
  - L'UID de ce badge est composé de 7 octets non modifiables.
  - Ce badge peut aussi être configuré pour générer un UID « pseudo aléatoire » à chaque connexion. Cette fonctionnalité permet de protéger l'anonymat du porteur : le gestionnaire du système ne pourra pas retracer les déplacements de l'utilisateur en suivant les apparitions de son badge sur les différents lecteurs de son installation.
- **Mémoire**
  La mémoire de ce badge contient 144 octets (1152 bits) utilisables pour stocker les données de l'utilisateur.
- **Protection**
  - La mémoire peut être verrouillée (par blocs) pour devenir « Read Only » de manière irréversible.
  - L'accès à la mémoire peut être protégé par une authentification mutuelle : le badge et le lecteur doivent se prouver l'un à l'autre qu'ils connaissent une même clé de chiffrement AES-128. Cet échange est basé sur l'algorithme AES en mode CBC (Cipher Block Chaining).
  - Les messages peuvent-être authentifiés sur l'interface radio par l'ajout d'une signature calculée avec l'algorithme CMAC.
  - Les messages ne sont pas chiffrés sur l'interface Radio : ils sont transmis en clair.
  - Il est possible de limiter le nombre d'échecs d'authentification successifs. Si le nombre d'échecs d'authentification est dépassé, le badge est rendu définitivement inutilisable.
- **Preuve d'origine**

Ce badge fournit une fonction permettant de s'assurer qu'il s'agit bien d'un badge original et pas d'une copie. Cette assurance protège contre la duplication d'un badge original sur un badge vierge, mais pas contre la duplication par un émulateur qui pourrait aussi reproduire la signature du badge original.

- **Autres fonctionnalités**

  Ce badge Implémente 3 « one way counters », c'est-à-dire qu'ils ne peuvent qu'être incrémentés par un lecteur, et jamais décrémentés. Ces compteurs peuvent être utilisés pour limiter le nombre de passages dans les moyens de transport par exemple.

- **Remarques**
  - Même si l'algorithme AES est plus robuste que l'algorithme 3DES, ce badge utilise aussi le mode CBC avec un IV (Initialization Vector) fixe, ce qui le rend vulnérable contre des attaques cryptographiques. Mais ce type d'attaque ne semble pas être encore implémenté dans les outils de tests de pénétration.
  - Comme pour le badge Mifare Ultralight C, il est fortement recommandé d'utiliser des clés de chiffrement uniques pour chaque badge (Cf tag Mifare Ultralight C).

Mifare Desfire EV3 (MF3D(H)x3)

Ce badge est la version la plus récente de la famille Mifare Desfire, après Mifare Desfire EV1 et Mifare Desfire EV2. C'est le badge offrant le plus haut niveau de sécurité de tous les badges Mifare, qui est le même niveau que celui recommandé pour les cartes de paiement ou pour les passeports électroniques. NXP préconise son utilisation dans des applications de ville intelligente, où un même badge pourrait être utilisé pour accéder à des services différents (transport, partage de véhicules, micro-paiement…). Chaque application peut y stocker ses propres fichiers, dans la limite de la taille mémoire disponible. La taille mémoire varie de 2kOctets à 16 kOctets suivant les modèles.
Ces badges fournissent une fonction de vérification de l'originalité du badge, et implémentent des fonctionnalités de détection de proximité pour se protéger des attaques par relais.

- **UID**
  - L'UID de ce badge est composé de 7 octets non modifiables.
  - Ce badge peut aussi être configuré pour générer un UID « pseudo aléatoire » à chaque connexion. Cette fonctionnalité permet de protéger l'anonymat du porteur : le gestionnaire du système ne pourra pas retracer les déplacements de l'utilisateur en suivant les apparitions de son badge sur les différents lecteurs de son installation.
- **Mémoire**
  La mémoire de ce badge contient 2kOctets, 4 kOctets, 8 kOctets ou 16kOctets utilisables pour stocker les données de l'utilisateur. Cette mémoire gérée par un File System, où les applications peuvent créer des fichiers. Cette organisation est flexible pour que le badge puisse abriter plusieurs applications ayant des besoins de stockage différents.
- **Protection**
  - La mémoire est partitionnée pour qu'une application ne puisse pas accéder aux données d'une autre application, sauf si elle y est explicitement autorisée.
  - Au niveau du badge, un certain nombre de clés 3DES et/ou AES permettent de protéger l'accès au badge. Le propriétaire de ces clés (c'est-à-dire le gestionnaire du système de badges) n'a pas accès aux données des applications.
  - Chaque application peut définir un certain nombre de clés de chiffrement (3DES ou AES) pour protéger en lecture et/ou écriture les fichiers qui lui appartiennent.
  - Les données des applications peuvent être chiffrées et authentifiées sur l'interface radio grâce à des clés qui sont propres à chaque application.
  - Les clés de chiffrement peuvent être mises à jour.

- **Preuve d'origine**
  - Ce badge fournit une fonction de preuve d'origine basée sur une signature ECDSA générée par NXP.

- **Proximity Check**
  - Ce badge implémente une fonction de vérification de proximité pour se protéger contre les attaques par relais. Cette fonctionnalité est basée sur une mesure très précise du temps de transit des messages sur l'interface radio entre le lecteur et le badge. Cette fonctionnalité doit être initiée par le lecteur (donc il faut que le lecteur la supporte).
  - Cette fonctionnalité connaît sans doute la même faiblesse que pour les badges Mifare Plus si le lecteur et le badge sont reliés via un « *range extender* », c'est-à-dire une paire d'antennes NFC reliées par un câble RF.

Recommandations

Cette section liste les recommandations les plus importantes qu'il faut suivre lorsqu'on déploie un système basé sur des badges NFC. Comme discuté en introduction, ces recommandations ne s'appliquent qu'aux badges et pas aux lecteurs. Dans la mesure où toutes les familles de badges utilisent des algorithmes de chiffrement symétriques (Crypto1, 3DES ou AES) pour protéger les données, les lecteurs connaissent ou savent dériver les clés de tous les badges. La compromission d'un lecteur est donc un élément absolument critique. La sécurisation des lecteurs fera l'objet d'une étude distincte dans le cadre de la sécurisation de l'IOT.

- Tout d'abord, comme il est recommandé dans les pratiques « Secure By Design », il faut faire une étude de risques. L'objectif de cette étude est bien sûr d'identifier les risques auxquels on s'expose, mais surtout de définir si ces risques sont acceptables ou pas pour le système qu'on déploie. Cette analyse de risque permettra de sélectionner la famille de badge compatible avec ces risques. Toutes les applications n'ont pas besoin de badges Mifare Desfire EV3 …

- Limiter les effets d'une attaque
  Même si les concepteurs de badges introduisent de plus en plus de fonctionnalités liées à la sécurité, il est possible, voire probable, que des attaquants arrivent un jour à les contourner. Il faut donc implémenter au niveau du système des mécanismes de détection de comportements anormaux, par exemple l'utilisation d'un même badge à deux endroits différents à des instants très proches, ou l'évolution anormale du nombre de passages autorisés dans un badge de transport …

- Ne jamais utiliser l'UID/UUID d'un badge pour exécuter une action car cet UID/UUID est facilement clonable.

- Diversification des clés
  L'utilisation d'un jeu de clés différents pour chaque badge assure que si un badge est compromis, un attaquant ne pourra modifier que les données d'un badge avec le même UID : c'est-à-dire ce même badge, ou un clone de ce badge. Cette diversification pourrait être effectuée en dérivant le jeu de clés d'un badge à partir de son UID et d'une Master Key qui elle-même n'est pas stockée dans le badge.

- Chiffrer les données stockées dans les badges, en utilisant un algorithme robuste (AES par exemple) et des clés de chiffrement connues seulement du lecteur, c'est à dire des clés différentes des clés permettant d'accéder aux secteurs du badge.

- Implémenter des Black Lists et/ou des White Lists dans les terminaux. L'utilisation de Black Lists permet de compléter le mécanisme de diversification des clés en bannissant un UID qui aurait été compromis. L'utilisation de White Lists est plus sécurisée, mais elle est complexe à implémenter si le système est composé d'un grand nombre de badges.

- Implémenter un MAC sur le contenu du badge
  A chaque fois qu'un terminal écrit des données dans un badge, il calcule un code MAC à partir de ces données et de l'UID du badge. Ce MAC est calculé à partir d'une clé qui n'est connue que du lecteur, et qui peut elle-même être dérivée d'une Master Key. Mais cela n'empêche pas un attaquant de restaurer la carte à un état précédent dont le MAC est connu. Donc ce mécanisme est inefficace dans une application de transports public qui compterait le nombre de passage d'un utilisateur par exemple.

- Les badges Mifare Plus et Mifare Desfire permettent de mettre à jour les clés d'un badge. Si la Master Key est compromise, on peut déployer une nouvelle Master Key dans tous les terminaux, et programmer certains terminaux qui se trouvent dans des zones sécurisées pour qu'ils reprogramment les jeux de clés des badges à partir de cette nouvelle Master Key.

- Sécuriser le process de personnalisation des clés
  Puisque la sécurité repose sur des algorithmes de chiffrement symétriques, ces clés sont programmées dans les badges au moment de la personnalisation des badges. Il faut donc s'assurer que les locaux et le process utilisés pour cette personnalisation sont correctement sécurisés.

Références

- AN10969 : System level security measures for MIFARE installations
- AN12653 : End to end system security risk considerations for implementing contactless cards and tags
- MF0ULX1 - MIFARE Ultralight EV1 - Contactless ticket IC
- MF0ICU2 - MIFARE Ultralight C - Contactless ticket IC
- MF0AES(H)20 - MIFARE Ultralight AES contactless limited-use IC
- MF1S50YYX_V1 - MIFARE Classic EV1 1K - Mainstream contactless smart card IC for fast and easy solution development
- MF1SEP(H)10x1 - MIFARE PLUS SE - Secure contactless smart card IC for seamless migration
- MF1P(H)x2 - MIFARE Plus EV2
- MF3D(H)x3 - MIFARE DESFire EV3 contactless multi-application IC