AN12653

End to end system security risk considerations for implementing contactless cards and tags

Rev. 1.1 — 5 May 2021
155011

Application note
COMPANY PUBLIC

Document information

Information	Content
Keywords	Contactless card, contactless tags, end-to-end system security, attack and threat model, countermeasures
Abstract	This document lists possible security attacks and threats for systems that use contactless cards and tags as a medium to carry some kind of value and also provides tips on how to implement appropriate security in the system.

Revision history

Revision history

Rev	Date	Description
1.1	20210505	• AN number changed into AN12653, security status changed into "Company public" • Contactless tags to the title and scope added • Updated to inclusive terminology • Section 3.20 added
1.0	20080601	Initial version

1 Introduction

Contactless cards and tags[1] are being used in a broad range of applications like public transport, fare collection, e-purse, facilities access management, event ticketing, etc. In general, they are used as part of a secure infrastructure that includes a backend system, card readers and card validators, and possibly other equipment such as personalization and controlling equipment.

The data content of the cards generally represents some kind of value that can draw the attention of certain individuals to explore the security features of the card. However, the security of the entire system relies on all components of the infrastructure and must, therefore, not just rely upon the security implementation on the contactless cards. All parts of the system must be designed along with security targets amongst all its mission critical functions. Threats are derived from these security targets along with their potential countermeasures.

Each system deploying contactless smart cards has its own unique combination of system attributes for which only the system integrators and their customers can understand as a whole. It is up to the system integrators and customers to determine and deploy the best balance between the security measures implemented in the different components. The best balance must include consideration of the trade-offs between cost, user interface (ease-of-use), and the required level of security.

This document provides tips for implementing an appropriate level of security in systems using contactless cards. Some of the suggested measures may be hard to implement in practice. This document does not pretend to cover all possible threats specific to the use of contactless cards, nor does it pretend to cover all or the best countermeasures against the listed threats. It also does not pretend these countermeasures have no side effects.

Some of the proposed measures may unintentionally compromise the privacy of the end user (with regard to storing data that is related personal information and/or tracking the location of an individual). In such a case, it will normally be mentioned. An example of this is the recommendation to keep track of the last gates entered by the user.

The proposed countermeasures in general have an impact on the infrastructure of the system (reader and backend system) and can require the storage of some extra information in the contactless card. In general they can be implemented on any contactless card type, unless the storage capacity of the card is too limited to store all extra data.

This document focuses on the security of the communication between the reader and the card. It does not cover attacks on the reader devices, the backend system and the communication between reader and backend, nor does it cover hardware attacks on the card such as probing, chip analysis, chip modification, etc.

[1] In the remainder of the document, cards will be used to denote both cards and tags.

2 Possible threats and attacks

Possible threats are:

- **Spoofing of rights**: This happens when an attacker successfully poses as an authorized user of the system and is able to get access to some services, e.g. unauthorized access to a building, free ride at the expense of the public transport company or at the expense of an innocent user. In this model, the spoofing of rights threat covers several threats having different levels of severity depending on the application. Getting unauthorized access to a nuclear site is obviously more severe than getting a single free ride in public transport.
- **Tampering with card content:** An attacker modifies, adds, deletes or reorders data on a card, e.g. changing the travel product on a card (from a single trip to a valid yearly subscription), changing data to a value that was valid earlier (roll back attack).
- **Information disclosure:** Card content is read by an unauthorized person. Either the attacker gets access to the card content by using a fraudulent reader or the card content is eavesdropped during the communication between a reader and the card without the knowledge of the card owner. This can infringe the card owner’s information privacy, e.g. if the name and/or address on a card can be read by an unauthorized person.
- **Denial of service:** Denial of service occurs when a valid card cannot properly function to provide the expected service, e.g. denylisting a legitimate card, locking a card, etc. The card owner may not only be denied the service, but may lose money and confidence in the system.

Possible attacks are:

- **Eavesdropping communication** between a legitimate card and a legitimate reader.
- **Eavesdropping and disturbing the communication** between a legitimate card and a legitimate reader by disturbing the radio field so that a communication is not complete.
- **Reading or modifying the content of a legitimate card** using a standalone attack device.
- **Replaying the information** eavesdropped between a legitimate card and a legitimate reader. A valid transaction is maliciously repeated between a card emulator and a valid reader or between a valid card and a fraudulent reader.
- **Presenting a cloned UID** of a legitimate card or a faked UID toward a legitimate reader.
- **Cloning a card**: It is copying the content of a legitimate card to a blank card. For a blank card, it might not be possible to set a dedicated UID.
- **Card emulator:** The content of a legitimate card can be copied into a card emulator. A card emulator can emulate the complete legitimate card including the UID and simulates the behaviors of a legitimate card. The attacker has full control on the software running in the emulator and in particular they are able to restore a previous content (memory image) at any time.

The attacks can be mounted considering the following scenarios:

- The attacker attacks their own legitimate card or a stolen but legitimate card. The attacker is not restricted in time and can use each location of their choice.
- The attacker attacks the legitimate card of another user while this user is making a transaction. The attacker can only momentarily record the communication between a card and a reader. For an attack that involves multiple communications, e.g. for successively breaking of the keys, it will not be trivial to communicate to the card again

in a public context. This may be different, though, for access management systems where more predictable usage patterns may prevail.

- The attacker attacks the legitimate card of another user while this user is not making a transaction. For example, a user is sitting in a train or bar and an attacker next to them attacks the user’s card with a device that has the size of a cell phone. An example of read out attempt can be: the attacker may have time to break one key, and then read data, break the next key etc., but not if breaking a key will take a long time.

3 Countermeasures

Table 1 provides the matrix of possible threats and related attacks, described in section Section 2, with countermeasures, which make the attacks more difficult. The countermeasures referenced from 1 to 19 are respectively described in Section 3.1 to Section 3.19. The efficiency of some countermeasures depends on whether the attacker uses a cloned card or a card emulator. Note that a number in a box means that the particular countermeasure can contribute to the counter the attack. In many cases multiple countermeasures will be needed to reduce the risk to an acceptable level.

Table 1. Threats, attacks and countermeasures matrix

	Spoofing rights	Tampering with card content	Information disclosure	Denial of service
Eavesdropping communication	1, 8, 9, 12[1], 14, 15, 16, 17, 20	N.A.	1, 2, 10, 17	N.A.
Eavesdropping and disturbing communication	1, 3[2], 4[2], 6[2], 7[2], 8, 9, 11, 12[1], 14, 15, 16 17, 19, 20	1, 3, 8, 9, 10, 11, 13, 15, 17, 19, 20	1, 2, 10, 17	11, 13
Reading or modifying content of legitimate card using an attack device	N.A.	1, 3, 8, 9, 15, 17, 20	1, 2, 17	1, 17
Replaying the information	12[1]	13	1, 2, 10, 17	
Presenting a cloned UID	1, 4[2], 5[2], 6[2], 7[2], 11, 12[1], 14[2], 15, 16, 20	1, 4[2], 5[2], 6[2], 7[2], 11, 12[1], 14[2], 15, 16, 20	1, 4[2], 5[2], 6[2], 7, 11, 14[2], 17	14[2], 17, 19, 20
Cloning a card	1, 3, 4, 5, 6, 7, 9, 12, 15, 16, 19, 20	N.A.	1, 2, 4, 5, 6, 7, 9, 10, 11, 17, 18, 19	18, 19, 20
Card emulator	1, 9, 12, 16, 18, 20	N.A.	1, 2, 4, 12, 17, 18, 19	12, 18, 19, 20

1. The countermeasure is efficient against card emulator but not against cloned card.
2. The countermeasure is efficient against cloned card but not against card emulator.

3.1 Key diversification

Key diversification means that every card gets its own specific keys. Thus, if one key of one card is discovered, then the other keys of this card and the keys of the other cards remain secret.

Key diversification also prevents cloning of a legitimate card to another card having a different UID. To clone a card, the UID and its corresponding keys must be duplicated.

Remark: Key diversification does not prevent cloning a card with a card emulator since in that case the UID can be copied.

3.2 Encryption of data

The data on the card can be encrypted by the reader with diversified keys that are independent from the keys on the card. In that case, a strong encryption scheme such as 3DES or AES can be implemented that should also diversify keys per card and per application.

This measure improves the confidentiality of the data exchanged between the reader and the card in three ways:

1. If the key that protects the communication between the reader and the card is broken, the transferred data is still confidential.
2. It allows encrypting transferred data when the encryption of contactless card cannot be enabled. This also increases the protection of the privacy of the data. For instance, older MIFARE DESFire D40 and EV1 protocols could not encrypt AND sign a message. Thus, the reader can encrypt the data with an application key and the signature of the message is performed by the MIFARE DESFire D40 or EV1 protocol. Another example is the use of the encryption by the reader for MIFARE Ultralight which provides no encryption in its protocol.
3. This countermeasure makes it more difficult to mount an attack on the keys by collecting combination of plaintext and cipher text (read Section 3.10). It becomes more difficult to harvest known clear text together with its cipher text.

3.3 Cryptographically bind data with card UID

The reader computes one or more strong cryptographic signatures on a combination of all data stored on the card and the UID. These signatures are stored on the card. Thus, all card information is bound to the card via the UID. The signatures can be a keyed hash value.

When using multiple signatures (e.g. because not all data is always read), care must be taken that there is sufficient overlap between the data being signed by the different signatures so that the overall integrity can be maintained.

The reader checks the signatures to verify the information integrity and the binding with the UID. It prevents the cloning of a card to another card with a different UID. It does not block the cloning to a card emulator emulating the correct UID.

It also makes it harder to maliciously modify the card content such as the travel balance. Restoring a previous content onto the same card with the valid signature cannot be detected by this measure.

Remark: Cryptographically bind data with card UID does not prevent cloning a card with a card emulator because then the UID can be copied.

3.4 Allow listing

The infrastructure keeps a list of legitimate cards and only communicates with the cards that belong to the infrastructure allow-list. The UID can, for instance, be used to identify the card.

A fraudulent card must emulate a valid UID to communicate with the infrastructure.

Remark: This measure does not prevent cloning a card to a card emulator since then the UID can be copied.

3.5 Deny listing

The infrastructure maintains a list of likely, tampered cards. A fraudulent card is rejected by the infrastructure after checking it against the denylist accessible to the readers. The update of the denylist is easier to implement with online readers than with offline readers for which the denylist cannot be updated frequently. Yet the technique is still feasible.

A reader detecting a fraudulent card could try to write an indication with a signature onto that card. The next reader reading the card would read the indication, verify the signature and then update its denylist.

For example, a fraudulent card can be detected with the countermeasures measures presented in Section 3.3, Section 3.8, Section 3.11, Section 3.12, Section 3.14 and Section 3.18.

Remark: This measure does not prevent cloning a card with a card emulator since then the UID can be copied.

3.6 Hot listing

When a reader identifies a hot listed card, the infrastructure enables an alarm at the gate alerting the security guard for instance.

Remark: This measure does not prevent cloning a card with a card emulator since then the UID can be copied.

3.7 Card revocation

A card is revoked by a reader when it is identified as fraudulent. For example, the reader could write data to the card indicating it has been revoked. The card revocation does not require the propagation of a denylist to offline readers.

For example, a MIFARE Classic product-based card can be revoked by setting the application's sector(s) access condition bits to all 1s, forever disabling the reading or writing of data to that application sector(s).

Remark: This measure is efficient for a cloned card but not for an emulated card. The information indicating the revocation can be removed from the card emulator.

3.8 Include a transaction number

A transaction number can be added to the card and traced by the backend system. An authenticated transaction number is implemented in the card and is decremented before doing any operation with the card content. The transaction number together with the card UID and time stamps are communicated to the backend. For each UID, the backend buffers the latest transaction numbers and timestamps. When all readers have communicated their transactions to the backend, the cache is cleaned up and only the latest transaction number is kept by the backend.

This measure makes it harder to use cloned cards and rollback attacks but may raise privacy concerns by tracking the UID and time of the transaction.

The backend system can detect a fraudulent card in the 2 following cases:

- For a UID, when the transaction number is the same or higher than the transaction number that has been registered by the backend system
- For a UID, when a transaction is made at a time that is not sequential in time (outside the clock tolerance window) and transaction number with another transaction for the same UID.

When fraudulent cards are detected, the backend can decide to add the UID to the deny-list/hot-list and/or remove it from the allow-list.

3.9 Increment double transaction counter

The double transaction counter decremented via different ways: two transaction numbers are devoted to track the transactions:

- Counter 1 is written as a value field and after initialization only operated via decrement operations. This counter is protected with the same key as the data (for example, the fare balance). The counter is decremented before the new fare balance is written, but after the authentication, so that if an attacker tries to eliminate the decrementing of the counter, the writing of the new fare balance will fail as well.
- Counter 2 is written as a data field.

This measure makes replay attacks more difficult. For example, if an attacker would eavesdrop a session upgrading of the fare balance and replay it to the card while keeping the card random number the same (using sophisticated equipment) then the first transaction counter will be decremented again while the second transaction counter will keep the same value.

If a reader detects that the transaction counters do not match, it can refuse the card.

This needs to be refined in order to cope with incomplete communications. E.g., first write Counter 2, then decrement Counter 1. If Counter 2 is lower than Counter 1, the previous transaction was not fully executed. The reader could accept the card and fix the inconsistency. If Counter 2 is however higher than Counter 1, then it suggests that a replay attack was done and the reader can reject the card.

It is recommended to combine this measure with storing a keyed hash on the value of Counter 2. The incompleteness of the transaction could also be detected by first writing the keyed hash and then the counter. If the keyed hash belongs to a value of one less than the counter, it could be acceptable. In case of other differences, the card is rejected.

The counter is decremented rather than incremented, because access conditions can be set such that only decrement is allowed on a value field.

3.10 Avoid encrypting known or guessable texts

For a cryptographic attack on keys, an attacker needs a collection of cipher text/plaintext combinations. Thus, the reader should only encrypt data which are unknown or cannot be guessed. This measure restricts harvesting of cipher text/plaintext combinations and makes the key discovery more difficult.

3.11 Detect authentication failures

The reader implements a system detecting fraudulent cards when authentication fails for a number of times. In this case, the reader can decide to block the card by adding the card to deny/hot list or by revoking the card.

3.12 Check physical form factor

The physical form factor is verified. For instance, controllers in a train or bus or guards at the access control gates check the physical form factor to trap emulator devices.

Remark: This measure does not prevent fraudulent card since they have the same physical form as a legitimate card.

3.13 Read back data after write

The reader reads out and verifies data after it has been written on the card.

This countermeasure can detect an attack where the attacker intercepts the writing command and manipulates an ACKNOWLEDGE/NOT-ACKNOWLEDGE response with a replay attack. Reading back the new content of the card confirms the modification of the card content.

To avoid the manipulation of the read back operation, this countermeasure should be combined with other measures. For instance, the 2 following measures can be proposed:

- When the card content is read back, the command and the modified card content are signed by the card.
- During the writing, the command and its operand are encrypted by the contactless card transaction protocol. Thus, the attacker is not able to build the correct read command response.

3.14 Check the UID of the card

The reader verifies that the UID used during the anti-collision sequence is identical to the UID stored in the card memory. The card is assumed to be tampered if there is a discrepancy.

Remark 1: This measure is only applicable to the cards that do not use a random UID during the anti-collision sequence.

Remark 2: This measure does not prevent card emulators because the UID can be copied.

3.15 Maintain in/out state on the card

For cards used in a system that deploys an in/out state (public transport, access management) an in or out status is recorded on the card. The card user cannot get into an infrastructure before getting out of it.

This measure makes it harder to record multiple transaction traces. For instance, an attacker would have to enter the system, and get out before being able to get a new trace of a transaction to enter the system.

This countermeasure can be much enhanced by physically separating the readers for entering and exiting the system. Entrance and exit readers should not be in reach of each other and the exit reader should be located within the protected area of the infrastructure.

3.16 Maintain in/out state in the infrastructure

For cards used in a system that deploys an in/out state (public transport, access management) the in or out status is recorded in the infrastructure. The card user cannot get into an infrastructure again before getting out of it. For this countermeasure, all the readers of the system must always be online.

The added value compared to measure Section 3.15 is the solution with the infrastructure is more robust should an attacker tamper with the in/out state in the card.

Remark: This countermeasure may create a privacy issue.

3.17 Implement rolling keys

The keys embedded in a card will be updated after a certain number of transactions. Changing the keys can be done periodically as a security policy or after each transaction, for example, to prevent replay attacks. Updating the key at the end of a transaction protects against a replay attack since the replayed session is using the old session key. It does not prevent the cloning of a card with the consistency of data and keys.

With some contactless cards, the rolling key mechanism can only be implemented in a controlled environment where the card can never be taken out of the reader's radio field before the completion of the key update. Powering off the card during the writing of keys, can result in an undetermined value being stored, rendering that keyed memory inaccessible. A good example of a controlled environment to update the keys is a reader that physically fixes the card while writing and releases it when the update is done.

3.18 Detect a genuine card in the backend office

A signature (for example, signed hash of the UID plus possible padding) is stored on the card. The signature is used to verify whether a card is genuine or a clone. This verification can be done by the backend system or a similar controlled environment.

The read access key to this signature is diversified per card and the master key used for the key diversification is only available by the backend system or in other controlled environments of the infrastructure. The signature is never accessed from a public reader.

This countermeasure makes it harder to fully clone a card by eavesdropping the communication between a reader and a card in a public environment.

3.19 Put authentication data in the first section

Some data identifying the card is added to the first sector or file that is accessed by the reader. For instance, this could be an encrypted version of the UID.

A reader can then detect a cloned card by the discrepancy between the UID and this extra authentication information. The discrepancy occurs when the authentication data is copied on a cloned card having a different UID than the original genuine card. As soon as a reader detects a malicious card, it stops reading data to avoid exposing cipher text/known or guessable plaintext combinations.

3.20 Fraud detection

Various potential ways for fraud detection have already been hinted in the previous sections. There are however many more ways, which cannot all be described in this document in detail. For example, if the card stores a value representing an account balance, the backend can keep track of this value for each card (e.g. linked to the UID) and detect anomalies. Another option could be to keep track of the times and places a card is presented, and detect anomalies which are physically not possible.

These methods can be applied in an online or offline way, and then result in immediate or deferred denylisting or hot listing of fraudulent cards, as suggested in Section 3.5 or Section 3.6.

4 Abbreviations

Table 2.

Acronym	Description
AES	Advanced Encryption Standard
DES	Data Encryption Standard
3DES	Triple Data Encryption Standard performing three serial DES operations
MAC	Message Authentication Code
UID	Unique Identification number

5 Glossary

Table 3.

Term	Description
Contactless card	Card that is able to communicate with a reader via Radio Frequency waves. There is no need to make physical contact.
Decryption	Convert encrypted (ciphered) data into plain text using a secret key.
Encryption	Convert plain text into encrypted (ciphered) data using a secret key.
Reader	Device that can read and write to contactless cards.

6 Legal information

6.1 Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

6.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors. In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory. Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification. Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products. NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Limiting values — Stress above one or more limiting values (as defined in the Absolute Maximum Ratings System of IEC 60134) will cause permanent damage to the device. Limiting values are stress ratings only and (proper) operation of the device at these or any other conditions above those given in the Recommended operating conditions section (if present) or the Characteristics sections of this document is not warranted. Constant or repeated exposure to limiting values will permanently and irreversibly affect the quality and reliability of the device.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

No offer to sell or license — Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Translations — A non-English (translated) version of a document is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

6.3 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

MIFARE — is a trademark of NXP B.V.

DESFire — is a trademark of NXP B.V.

MIFARE Ultralight — is a trademark of NXP B.V.

MIFARE Classic — is a trademark of NXP B.V.

NXP — wordmark and logo are trademarks of NXP B.V.

Tables

Contents

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 5 May 2021
Document identifier: AN12653
Document number: 155011