https://doi.org/10.52326/ic-ecco.2022/CS.06

Comparing two security models for RFID

Rareș-Aurelian Radu [1], ORCID: 0000-0002-3168-6047
[1] Alexandru Ioan Cuza University of Iași, Faculty of Computer Science, supervisor Prof. Dr. Țiplea Ferucio-Laurențiu
radu.rares.a@gmail.com, https://ro.linkedin.com/in/rares-aurelian-radu-18400b223

Abstract — **Radio Frequency Identification Technology became more and more involved in authentication processes over the years and is still rising. Security in this context needs to be strongly ensured, hence security models have a crucial role motivated by the fact that any entity with the right tools can interfere or eavesdrop in the communication process between a tag and a reader. The two most relevant, complete and worth mentioning models at this hour are Serge Vaudenay's model based on the introduced 'blinder' notion and the HPVP model of J. Hermans, R. Peeters and B. Preenel's based on the left-or-right indistinguishability notion. We provide a comparison between these two models that highlights not only the differences and the similarities, but also the elements that make each model unique along with the tag corruption aspects and the different privacy levels achieved by each model regarding both symmetric and asymmetric cryptography.**

I. Introduction

In recent years, Radio Frequency Identification technology has become more and more involved and relevant in authentication processes and it is still rising. Security in this context needs to be strongly ensured. Tags' and readers' communication has to resist attacks such as impersonation attacks or eavesdropping. Several security models have been designed and suggested to tackle the security and privacy issues. Two of these models we consider to be the most relevant, complete and worth mentioning at this time are Vaudenay's model based on the introduced *blinder* notion [1] and the HPVP model of J. Hermans, R. Peeters and B. Preneel's based on the *left-or-right* indistinguishability notion [2]. In this paper we come up with the detailed comparison of these two models, which we believe has not been provided strictly on these two models side by side. We believe this papers' approach highlights the important differences between two highly used security models, points out the stronger notions of privacy that can be achieved in similar contexts and, lastly, presents an invitation to anyone interested in the subject and to anyone willing to contribute on the open issue of achieving higher notions of privacy with symmetric key encryption along with extending the HPVP model and refining the *blinder* of Vaudenay's model.

II. Basic Notations and Notions

The notations and definitions are similar to the ones provided in [2,1,3,4].

A. RFID System

Incorporates a set of tags □, a set of readers $\mathcal{R}$ and a communication protocol between them. Each tag T_i has an identifier ID, a memory (temporary memory and persistent memory) that contains a volatile state □ that may change during the course of life of the tag and that can store the ID. Each tag is a transponder that has finite internal memory and limited computable capacity. Reader's database holds the ID of the tag paired with a secret, $K_{Rj,\ Ti}$ for each tag T_i in □. The role of readers consists in identifying and recovering IDs of authorized tags and, on the other hand, to repudiate all other trials of communication. The authentication process of a tag is successful if the reader's database holds an entry for that specific tag. Such a system needs algorithms and protocols to setup tags, readers and to bind tags that already are or not in a readers' database in an online or offline manner.

B. RFID Scheme

Contains three important items:

1. **SetupReader(1^s)** - algorithm that initializes readers' database and generates the input that is composed of a public-key K_p and a private key K_s
2. **SetupTag$_{Kp}$(ID)** - algorithm that generates the tag's secret K and the initial state of the tag S. If one tag is *authorized*, the pair (ID, K) is kept in readers' database
3. **protocol between a reader and a tag** - that outputs ⊥ if the tag's identification fails, or an ID if the tag's identification succeeds

C. Public-key encryption

A public-key encryption scheme over the triple *(K, P, E)* is a system *S* which includes three algorithms: 1) a PPT algorithm used for generating a pair of keys (*pk*, sk) denoting the public-key and the secret-key, 2) an encryption algorithm that outputs a ciphertext computed from the encryption of a plaintext with the *pk* and 3) a

decryption polynomial-time algorithm which outputs the decrypted message computed from the ciphertext with the *sk* taken as input or ⊥ (denotes failure).

D. Symmetric encryption

A symmetric encryption scheme over the triple *(K, P, E)* is a system *S* which includes three algorithms: 1) a PPT algorithm used for generating a key *k*, 2) an encryption algorithm which outputs a ciphertext computed from the encryption of a plaintext with the *k* and 3) a decryption PT algorithm which outputs the decrypted message computed from the ciphertext with the *k* taken as input or ⊥ (denotes failure).

E. Pseudo-random function

A pseudo-random function (PRF) is a family of functions that if a function from this family is randomly chosen, then its input and output are indistinguishable from a computational point of view compared to a random function. Considering two polynomials, γ and δ, a set of keys □ and a security parameter $\lambda \in \mathbb{N}$, $□_\lambda = \{K \in □ \setminus |K| = \lambda\}$. A family of functions indexed by □ is denoted by $F = F_{\lambda, K \{K \in □\}}$ where $F_{\lambda, K}$ is a function from $\{0, 1\}^{\gamma(\lambda)}$ to $\{0, 1\}^{\delta(\lambda)}$. If γ and δ are polynomially bounded, if $2^{-\gamma(\lambda)}$ and $2^{-\delta(\lambda)}$ are negligible and if an adversary has negligible advantage in distinguishing an oracle simulating $F_{\lambda, K}$ with K random from an oracle initialized with a random function, we say that $F_{\lambda, K}$ is a PRF. The following security game proves that $F_{\lambda, K}$ is a PRF: a) a challenger chooses a random bit from {0, 1} b) if *b == 1* then the challenger picks K from $□_\lambda$ and sets $f = F_K$, otherwise the challenger picks a random function and sets *f* to that function and gives oracles access to *f* for the adversary c) the adversary outputs a bit *d*. The adversary wins if *d = b*.

F. Physically unclonable function

A physically unclonable function or PUF can be identified as a physical object that, when it is queried with some challenge, generates a response that depends on the object's particular properties and on the challenge. PUFs are assumed to be hard to clone, unpredictable in response given a stray challenge and tamper-evident regarding attacks or unauthorized physical access to them that result in changing the behavior of the challenges and responses relevant to them.

G. Basic notations

1. $t \in_R □$ means that *t* is uniformly chosen from the set □
2. |□| denotes the cardinal of □
3. □ is an *algorithm*, □ is an *oracle*, $□^□$ hints that □ has access to *oracle* □
4. *vtag* denotes a virtual reference for a tag
5. □ refers to an instance of a protocol
6. *m, m'* represent the message that is sent and the answer that is sent back
7. ⊥ represents the failure of an operation
8. λ represents a security parameter denoting the probability of an adversary breaking the cryptographic scheme

III. PHILOSOPHY OF THE MODELS

The HPVP model's philosophy is heavily based on the notion of left or right indistinguishability, whereas Vaudenay's model is based on simulations of interactions with a RFID system by the means of a *blinder*. The HPVP model, by design, does not suffer from the concept of *blinder* introduced in Vaudenay's model, the privacy games played are based on guessing with which tag an adversary has interacted with and, moreover, tackles important aspects such as *tag tampering*, *privacy leakage* or *tag corruption limitations* that in Vaudenay's model are present. In Vaudenay's model, a *blinder* simulates the operations of an adversary and the goal is to arrive at the conclusion, based on the simulations of a real adversary or a *blinded* adversary, regarding the output of the simulations.

IV. ADVERSARIAL MODELS

The definitions of readers and tags remain the same between both models. In regards to HPVP, the first difference occurred is that the set of readers $\mathcal{R}$ and the set of tags □ are initially empty and are being dynamically populated with readers and tags by the adversary. Another difference that appears in HPVP is that, as we will see below when we will talk about the oracle's differences between the two, the adversary is allowed to draw pairs of tags and is allowed to interact with only one of them, the *left tag* or the *right tag*. The goal of the adversary is to guess if the tag that he ended up interacting with is a *left tag* or a *right tag*.

Next, we are going to underline the similarities and differences between the definitions of the oracles given and present in both models. Firstly, we will touch on the oracles' definitions that they have in common:

- **CreateTag** oracle creates a free tag in both models, the differences that appear in HPVP model are that all the tags created are registered in reader's database, hence all the tags are legitimate tags, the oracle does not fail on duplicate IDs and returns a reference to the new tag created; in Vaudenay's model, if the tag is not legitimate, the reader's database is not updated, otherwise it is updated and the oracle does not return anything.
- **Launch** oracle has the same outcome in both (resp. launches a new instance of a reader's protocol) with the difference that in the HPVP

model the reader can be chosen and given as input.

- **DrawTag** oracle, from the beginning, is quite different from both sides of the model; in HPVP, the oracle takes two real tag (free) references as input, T_i and T_j, generates a *vtag* for one of the tags, depending on *b*'s value and stores the triple *(vtag, T_i, T_j)* in □' (the hidden table from Vaudenay's model); in Vaudenay's model, the oracle takes an algorithm, as input, to draw a number of tags and generates a vector of *vtags* for the tags, generates a vector of bit values (1 for legitimate tags, 0 for the others) for the drawn tags and returns a combined vector of the two ($vtag_1$, b_1, $vtag_2$, b_2, ...).
- **Free** oracle presents the same input and output with the difference that in the HPVP model the *vtag* is no longer accessible because of the deletion of the triple *(vtag, T_i, T_j)* from the □', the tag's temporary memory is erased, but its state is maintained.
- **SendReader** and **SendTag** oracles both send a message; in Vaudenay's model the message is sent to a reader's protocol instance or to a *drawn* tag which is identified by the *vtag* taken as input for the **SendTag** oracle and returns a list of successive protocol messages; in HPVP model the message is sent to a tag according to the *vtag* and *b* value given as input (for **SendTag** oracle), or to the reader identified by its reference R_j given as input (for **SendReader** oracle) and returns a reply message for the tag, if the triple corresponding to □' is found or ⊥ otherwise (for **SendTag** oracle), and from the reader R_j if the reader sends a reply otherwise it does not return anything (no reply given from the reader).
- **Result** oracle in both models takes as input an instance of a session □ and may output a bit value; in Vaudenay's model it returns 1 if the session is complete, or 0 otherwise; in HPVP model a reader R_j is also given as parameter for which the session instance □ is bound to and returns a bit value if the reader acknowledged the session and authenticated a tag successfully and the session is finished, otherwise it returns ⊥.
- **Corrupt** oracle, in Vaudenay's model, takes as input a *vtag* to a tag, the tag is destroyed if the tag is never used again and returns the current state of the tag; in HPVP model, the oracle takes as input a real reference to a tag, no control over the tag is given to the adversary and returns both the temporary and permanent state of the tag.

Secondly, we will touch on the newly-introduced oracles in the HPVP model that in Vaudenay's model do not appear:

- **CreateReader** creates a new reader and returns a reference R_j for the reader created.
- **RegisterTag** takes as input a tag T_i and a reader R_j, then bounds the tag with the reader R_j.
- **CreateInsider** takes as input an ID of a tag, calls **CreateTag** to create a new tag, corrupts it for it to become an *insider tag*, introduces it in a list of *insider tags* and returns the insider tag and its state; the oracle is used for exploiting the privacy of other tags using the state of a corrupted tag.
- **CorruptReader** oracle takes as input a reader R_j, corrupts it by leaking the reader's DB and returns that DB along with all the secrets; when a reader is corrupted, only a tag should authenticate to other readers it is bound to and its identity must be concealed for those readers.

V. SECURITY AND PRIVACY

Both models present the same definitions in regards to security. Both models articulate that for achieving security, *tag authentication* and *reader authentication* have to be provided.

Tag authentication is achieved if, for any **STRONG** adversary, the probability of retrieving an uncorrupted tag ID along with not having a matching conversation with any tag ID in that protocol instance has to be negligible.

Reader authentication is achieved if, for any **STRONG** adversary, the probability of authenticating the reader with an uncorrupted legitimate tag ID along with not having a matching conversation is negligible.

Regarding privacy of the models, the restrictions on the adversary in the HPVP model are the same as in Vaudenay's model. We will now talk about the adversary classes which are, or not, present in both models.

In both Vaudenay and HPVP models the following classes are present are remain the same:

- **STRONG**: no restrictions on any oracle usage.
- **DESTRUCTIVE**: **Corrupt** oracle destroys the tag.
- **FORWARD**: only corruptions allowed after the first **Corrupt** oracle usage.
- **WEAK**: not allowed to call the **Corrupt** oracle.
- **NARROW**: cannot use the **Result** oracle.

The following adversary classes do not appear in Vaudenay's model but have been introduced in the HPVP model:

- **WIDE**: no restrictions on the usage of the **Result** oracle.
- **INSIDER**: allowed to call the **CreateInsider** oracle.

- **FORWARD-INSIDER**: allowed to call **CreateInsider** oracle, but only allowed corruptions after the first **Corrupt** call.
- **WEAK-INSIDER**: allowed to call **CreateInsider** oracle, but not allowed to call the **Corrupt** oracle.

In Vaudenay's model, privacy of a given class is achieved if, with the help of the *blinder* notion, the probability of receiving different output given the communication between a reader and a tag from both a real adversary and a *blinded adversary* simulation is negligible, then privacy is achieved.

In HPVP model, privacy is denoted by the probability of an adversary to guess correctly with which tag he interacted, either a *left tag* or a *right tag*, based on the guess bit outputted. If this probability is negligible then privacy is achieved.

VI. TAG CORRUPTION ASPECTS

In regards to Vaudenay's model, if an adversary corrupts a tag, then both the *temporary* and *persistent* part of the tags are revealed, hence no notion of privacy is possible to be achieved. However, if the tag cleans its *temporary* part each time the adversary loses the tag from its range, then the corruption problem vanishes. On the other hand, if the tag is in the adversary's range, corruption can be made before the cleaning of the *temporary* part of the tag, hence *reader authentication* is not possible. With *temporary state disclosure* only **WEAK** and **NARROW-WEAK** privacy might be achieved [5]. Without *temporary state disclosure*, only the *persistent* part is revealed, but the possibility of learning the *temporary state of the tag* can be exploited during the protocol execution, hence an adversary can distinguish between a *blinded adversary* or a *real adversary* simulation in regards to *reader authentication* [5]. On the other hand, by not being able to interact with the *temporary state of the tag*, an adversary cannot make any verification between the responses of the simulations of a *blinded adversary* and a *real adversary*, hence **NARROW-FORWARD** privacy is not reachable. On the other hand, with PUFs added to the PRF based scheme which ensures **WEAK** privacy in [1], the problem of achieving **DESTRUCTIVE** privacy is resolved if the call on the **Corrupt** oracle on a tag destroys the PUF and the **Corrupt** oracle provides the state of the tag between protocol steps. *Mutual authentication* is also ensured with the addition of a seed to a PUF for extending the domain of the PRF function.

In regards to the HPVP model, some restrictions on tag corruption are imposed. **Corrupt** oracle reveals both the *persistent* and *temporary* parts of a tag, hence privacy notions that are stronger can be achieved. Corruptions can occur only if physical access is possible. Another restriction is that an adversary is forbidden to corrupt tags that are in the course of being drawn for him to learn if that tag is an active tag because, otherwise, it would contradict the physical access assumption and also, he would be able identify that tag. Hence, corruptions by any adversary are allowed to be made only on inactive tags, those drawn in the *left* or *right* privacy games.[2]

VII. WEAK PRIVACY IN HPVP

As another common element between the two models, the RFID scheme based on PRF, found in [1], achieves **WEAK** privacy in both models. We saw that in section 4.1 in [1], **WEAK** privacy has been proved for Vaudenay's model, based on **Lemma 8** [1], by proving **NARROW-WEAK** privacy. The usage of the *blinder*, which simulates the privacy game without knowing the secrets of the tag or the reader and simulates the **Launch**, **SendReader**, **SendTag** and **Result** oracles, is similar with the indistinguishability game. If there is no way to distinguish between the output of a *blinded adversary* and the output of the *real adversary* then privacy is achieved. If a **NARROW-WEAK** adversary has no significant advantage over a *blinded adversary*, meaning that if the output of the protocol is not different between the two, with negligible probability, then in the privacy game played by both, when the reader never picks duplicate *a*'s, the tag never picks duplicate *b*'s and i does not present an advantage to any because of the PRF properties, then a **NARROW-WEAK** adversary does not win more than a *blinded adversary* and vice-versa.

Given the HPVP model, by using the same PRF-based RFID scheme, by design **WEAK** privacy is achieved in this context. As mentioned above, a *blinder* for an adversary simulates the operations of that adversary. If there is no way to distinguish between a *blinded adversary* and a *real adversary* then privacy is achieved. In the privacy game in the context of PRF, we first play with the PRF function and then play with a random function which outputs only random elements - this represents the way of distinguishing the PRF function from the random function. HPVP model is based on the *left-or-right* indistinguishability which fits the PRF game, hence fits to achieve **WEAK** privacy. The blinder simulates a random game which is complementary at its core with the random game simulated in the PRF game.

VIII. FORWARD PRIVACY USING SKC

In Vaudenay's model, the impossibility result of **NARROW-FORWARD** privacy in [1] denotes the fact that with temporary state disclosure we cannot both achieve **NARROW-FORWARD** privacy and *mutual authentication*. Thus, **NARROW-FORWARD** privacy can be brought up only if the corruption does not disclose the temporary state of the tag or the authentication is not

mutual. In [6] the author clarifies that **FORWARD** privacy cannot be achieved in Vaudenay's model. Desynchronization is heavily involved in the proof of the last affirmation. Desynchronization implies that one tag can be desynchronized with the reader if, when the communication between the tag and the reader starts, the tag secret is updated but the reader's database fails to update its database due to a protocol shutdown. Unbounded desynchronization means that there are no boundaries given the number of times one is allowed to desynchronize a tag and a reader. Bounded desynchronization means that after a certain predefined number of steps the tag and the reader synchronize back. The proof of the impossibility of achieving **FORWARD** privacy in Vaudenay's model is by contradiction. It is assumed that there exists such a scheme and there is an adversary that creates two authorized tags, draws one tag, launches an instance of a protocol, receives the reader's message, the tag sends back a reply message, the adversary frees the tag drawn and draws another one. Then, the adversary queries the tag on the first message but in another protocol step, the tag answers with a different message then the first reply, the adversary corrupts the messages, retrieves the table which links temporary tags with real tags. Using a SKC, the adversary which now has the link table and knows what the database states, can check if the reader's and tags' answer and reply are valid. By using the *blinder* notion, it is shown that a blinder can give a valid answer to a *vtag* with 1/2 probability, hence, an adversary can distinguish between a real or blinded privacy game and no **FORWARD** privacy can be achieved.

In [7], as briefly talked about also in [6], four classes of RFID schemes are introduced when talking about symmetric key protocols:

- Type 0: no tag state updates in the reader's database
- Type 1: tag state is updated at each protocol execution
- Type 2a: tag state is updated after a reader authentication
- Type 2b: tag state is updated before a reader authentication

In [7], Type 0 was demonstrated to not provide **FORWARD** privacy due to the fact that corruption discloses the key tag and provides only **WEAK** privacy given the PRF used. Type 1 was demonstrated to not provide any notion of privacy that is not **NARROW** due to the fact that Type 1 protocols are Type 0 protocols with key update and synchronization. Due to desynchronization only **NARROW-FORWARD** can be achieved. Type 2a is proved to be reduced to Type 0 protocols, hence **WEAK** privacy is achieved and Type 2b can be reduced to Type 2a or Type1.

Now, having given the information above, regarding HPVP, can we assume that the higher notion on privacy achieved is **WEAK** privacy?
If there is a Type 0 SKC protocol for the HPVP model, it is safe to assume that at least **WEAK** privacy is achieved. The authors of HPVP refer to [7] and assume that the examples of protocols based on SKC given are expected to achieve the same privacy notions since no protocols given as examples achieves **WIDE-FORWARD** privacy. The question remains open still.

IX. FORWARD PKC-BASED SCHEME IN VAUDENAY'S MODEL IS STRONG IN HPVP MODEL

In this section we highlight the differences between the proposed HPVP model and Vaudenay's model in regards to the Vaudenay's PKC-based protocol which is **NARROW-STRONG** private given the encryption is IND-CPA and **FORWARD** private given the encryption is IND-CCA which in HPVP model the protocol achieves higher notions of privacy.

We reiterate below the experiment which follows Vaudenay's model on the PKC-based protocol proposed.

1. reader picks a random $a \in \{0, 1\}^{\alpha}$
2. sends a to the tag
3. the tag computes a challenge c by encrypting the concatenation of the tag ID, the shared secret K between the tag and the reader and the received a from the reader with the K_p of the reader and sends the challenge c to the reader
4. reader decrypts the challenge c with its K_s, retrieves the shared secret K, identifies the tag ID and the given a from the tag
5. if the value of a sent by the tag is equal with the a sent by the reader to the tag in the first step and if the shared secret K is located in the reader's database, then the system outputs the tag ID, otherwise outputs $\perp$, meaning failure

By relying on the IND-CPA encryption, the protocol achieves **STRONG**} privacy given **NARROW** adversaries in Vaudenay's model and **FORAWRD** privacy by relying on the IND-CCA encryption.

In HPVP model [8], the same protocol achieves **NARROW-STRONG** privacy if the encryption is IND-CPA and **WIDE-STRONG** if the encryption is IND-CCA. For the proof of **NARROW-STRONG**, assume an adversary that wins the privacy game with high probability and construct an adversary that wins the IND-CPA game with high probability. The adversary that is constructed communicates to the adversary that wins the privacy game by simulating the system where the reader's K_p is the public key for the IND-CPA game and the **SendTag** oracle retrieves two references of two distinct tags using their *vtag*, generates the corresponding c from

step three of the above experiment for each tag with the encryption oracle of the IND-CPA game which returns only one of the two challenges and then the newly constructed adversary outputs the guess of the given adversary. If the given adversary can distinguish between the left or the right world, then the constructed adversary conquers the IND-CPA game. Based on the security of the models (which is common between the two as presented in the section 6.1.3) and the correctness of the scheme, **Lemma 8** in [1] helps in defining the **WIDE-FORWARD** privacy. To achieve **WIDE-STRONG** privacy, IND-CCA encryption must be used. Following the proof for **NARROW-STRONG** privacy, when getting the output of the **Result** oracle, the adversary compares *c* with a list of encrypted ciphertexts provided by the encryption oracle of the IND-CPA game when **SendTag** oracle calls were made. If any matches then *true* is returned. If there are no matches, then the **Result** oracle hands the ciphertext to the IND-CCA decryption oracle and receives the plaintext which is later verified. This game has the exact outcome as the IND-CPA game.

X. CONCLUSION

Which model is better? There is no simple answer to this question. Both models provide different levels of privacy. If we need strong privacy with a reasonable demonstration, we can choose the HPVP model. However, this strong privacy is just forward privacy in Vaudenay's model. We can switch our approach, pick Vaudenay's model and try to achieve destructive privacy with special elements like PUFs. What if we need more than destructive privacy? We may reach a complicated situation due to the fact that Vaudenay's model cannot achieve strong privacy. If the blinder's definitions and restrictions can be tweaked, we may reach strong privacy, but that means we are no longer in Vaudenay's mode, hence the highest privacy class achieved is destructive, which essentially is forward privacy enriched with vast corruption capabilities. Can we approach a blinder-less scenario? What has been underlined is that the demonstrations based on the blinder are hard from a theoretical point of view. It would be useful if there was a simplified privacy-based model with the same levels of privacy as Vaudenay's model. Can the HPVP model be a better alternative for demonstrations? We believe that it is a relevant approach due to the fact that it inherits the properties and definitions of the indistinguishability-based cryptosystem models. However, is it natural for RFID schemes to distinguish between two privacy games? Certainly, the use of the blinder for proving privacy seems more natural and firmer than distinguishing between two privacy games. We open the discussion for extending the HPVP model and also, for refining Vaudenay's model by refining the blinder.

REFERENCES

[1] S. Vaudenay, "On privacy models for RFID", in *Proceedings of the Advances in Cryptology 13th International Conference on Theory and Application of Cryptology and Inoformation Security*, ser. ASIACRYPT'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 68-67

[2] J. Hermans, R. Peeters, and B. Preenel, "Proper rfid privacy: Model and Protocols", *IEEE Transactions on Mobile Computing*, vol. 13, no. 12, pp. 2888-2902, 2014

[3] F.L. Tiplea and C. Hristea, "Privacy and reader-first authentication in Vaudenay's rfid model with temporary state disclosure", Cryptology ePrint Archive, Paper 2019/113, 2019, https://eprint.iacr.org/2019/113. [Online]. Available: https://eprint.iacr.org/2019/113

[4] C. Hristea and F.L. Tiplea, "Destructive privacy and mutual authentication in Vaudenay's rfid model", *IACR Cryptol. EPrint Arch.*, vol. 2019, p.73, 2019.

[5] F. Armknecht, A.-R. Sadeghi, I. Visconti, and C. Wachsmann, "On rfid privacy with mutual authentication and tag corruption", in *Applied Cryptography and Network Security*, J. Zhou and M. Yung, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 493-510.

[6] F.L. Tiplea, "Narrow privacy and desynchronization in Vaudenay's rfid model", *International Journal of Information Security*, vol. 21, no. 3, pp. 566-575, Jun 2022. [Online]. Available: https://doi.org/10.1007/s10207-021-00569-0

[7] C. Y. Ng, W. Susilo, Y. Mu, and R. Safavi-Naini, "New privacy results on synchronized rfid authentication protocols against tag tracing", in *Computer Security – ESORICS 2009*, M. Backes and P. Ning, Eds. Berlin Heidelberg: Springer Berlin Heidelberg, 2009, pp. 321-336

[8] J. Hermans, A. Pashalidis, F. Vercauteren, and B. Preneel, "A new rfid privacy model", in *Computer Security – ESORICS 2011*, V. Atluri and C. Diaz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp.568-587.