International Journal of Engineering & Technology, *7 (4.7) (2018) 344-351*

SPC

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paperResearch Paper

A new lightweight authentication protocol in IoT environment for RFID tags

Alireza radan[1], Hoseine samimi[1], Ali moeni [1]

[1] Computer Engineering Faculty, University of Tehran kish international campus
**Corresponding author Email: kiyanradan@chmail.ir*

Abstract

Nowadays, internet of things (IoT) is a prominent technology that provides the field for connection and information transfer through communicational networks like internet and intranet. Since IoT is applied in various functions so that personal information of individuals or objects may be read without consent of them or in absence of a secure protocol, that is a challenge in this part, this study aimed at identifying tags efficiently in order to protect privacy of things against attackers. In this research, a two-factor authentication protocol based on the public key cryptography has been presented in order to protect privacy. It has been tried in this protocol to diminish complicated calculations while the proposed method can resist against some attacks like tampering with tag and tag reader, tag tracking, man-in-the-middle attack, and replay attack. To examine the proposed method, application of paying toll in traffic management was used. Various traffic models have been considered to pay toll in proposed model and implementation of this method was done in simulation environment of MATLAB. In this simulation, two cryptography algorithms with Robin and ECC public keys with keys in different sizes were evaluated and results showed that almost all of passed tags were identified using an anticipation rate with high rate between tag and tag reader in a system implemented with Rabin algorithm. Moreover, this ratio was higher than 90% in a system that was implemented with ECC algorithm. However, application of conventional symmetric cryptographic algorithms such as AES can reduce ratio of read tags to lower than 20%.

Keywords: *- Authentication, RFID, Tag, Tag Reader, Key Cryptography, Privacy*.

1. Introduction

Nowadays, internet of things (IOT) is considered as one of modern technologies that provides the field for connection and information transfer for all of creatures (human, animal, things etc.) through communicational networks like internet and intranet. In other words, it is defined as the relationship of sensors and devices through a network in which, they are connected with each other and their users. This concept can be either as simple as the connection between a smart phone and TV or as complicated as the monitoring urban and traffic infrastructures. From washing machine and refrigerator to our cloths, this network can cover many of devices and appliance around us.

With advent of IOT and development of modern industries in current world, we can suggest new solutions. RFID is one of these solutions that provide the filed for connection between various objects through internet [1]. Superior firms used RFID tags in 2017 in order to reduce their costs and gain competitive advantage.

We can identify and track various things in RFID technology using radio frequencies. RFID function correlates with two Tag and Tag reader devices. In fact, tags are electronic chips with a certain code that transfer information to a code reader. Code reader converts the radio frequencies received from the tag to digital information in order to send and process them in computer [2]. According to experts, the issue of security in RFID field has been a concern in 2017 so that new technologies are developed in a way that are inactive when exposure to any attack or privacy violation so that there will be 0% attack possibility against them.

Numerous studies have been conducted under the title of "lightweight cryptography" to produce new and protocols suitable for facilities with limited resources in small devices like RFID tags. Despite the large number of existing methods, few of them have been evaluated in terms of security and safety. Conducted studied show that some of broadly used lightweight cryptography algorithms have been impressible. The popular algorithm of MiFare Crypto is a sample of these impressible algorithms [3]. Development of lightweight cryptography standards is essential to be used broadly in IOT technologies. Moreover, composition of lightweight cryptography protocols that have been designed for light works should be analyzed as well as ordinary cryptography framework such as public key infrastructure (PKI) for final infrastructures. Key management is an important point that should be considered in this case [4]

2. Research Background

Various solutions have been presented in this field and numerous formal efforts have been conducted to legalize RFID security in some USA states, California, New Mecsico, Yota and Masachoset, Japan and European Union. For instance, a consultation group named working group on data protection published a document entitled "working document on data protection issues related to RFID technology" in European Union [5].

Kill order [6] of this solution was recommended by Auto-ID Center and EPCglobal. In this project, the tag has a specified code- for example 24 bit- that is planned when tag is created. EPC tag will be immediately inactive forever when receives specific Kill order from code reader (with PIN of authentication of tag specification). It seems that Killing tags are proper security methods just for short term since they destroy many of beneficial capabilities of RFID. In addition, they may destroy information details of product such as removal of series number that maintains data of product type; so they cannot be used in long term [6].

Faraday Cage method [7] is another method to protect security of things with RFID tags isolating them against electromagnetic waves. This action can be done using Faraday Cage (FC) that is made of net or metal plate that is resistant against radio signals or desired frequencies. Faraday cages have limited function. The problem is that they not only prevent from scanning RFID tags on the personal product but also helps thieves to pass through electronic article surveillance (EAS). In addition, if RFID tags are attached to large number of personal assets such as cloths, then there will be finite efficacy of faraday cage. Lightweight cryptographic authentication protocols of RFID are usually classified to three types of technologies including Hash Function, Cryptography Algorithms and LPN [9]. Scientific society has emphasized on IoT security in recent years because security issues are essential to guarantee the reliable interaction between devices [4][3][1]. Tewari and Gupta [10] proposed a lightweight mutual authentication protocol for IoT devices that use RFID tags in 2016. Since this protocol applies bit operations, it is highly efficient. The authors carried out an accurate analysis to show that this protocol is secure against various attacks. However, primitive studies have shown vulnerabilities in structure of this protocol. Has function-based authentication protocol uses a tag ID as hidden data the responses authentication and key update by the challenge/response protocol. Sarma, Weis and Engels proposed the Hash-lock Protocol [11] that is a leading work among this type of protocols. Another prominent protocol was presented immediately called Hash chain-based protocol but all of these protocols had still numerous security problems such as vulnerability to disclosure attacks, code reader, etc. in this regard, Lei and Cao suggested some solutions to solve these problems [12]. Osaka, Takagi and Yamazaki presented an authentication protocol based on ownership transfer [13] that is an ultra-lightweight protocol due to applying simple rational operations to achieve mutual authentication and tags. Moreover, it is claimed that protocols are secure in terms of “prevention from attack” and “resistant against counterfeit”.

3. Proposed Method

The issue of security in radio ID systems can be discussed within two viewpoints. The first viewpoint associates with basic specifications and system function. Security protocols should be deigned in a way to be resistant against different attacks. Second viewpoint relates to limited resources existing in system. It is important to examine if a security protocol can be implemented on these systems. System application should be considered when designing security protocols. Various uses require their specific considerations. Implementation method has been explained herein.

3.1- Structure of anti-collision protocol

An improved version of anti-collision algorithm of Q-Protocol was used in this research. Structure of this protocol is based on DFSA with ALOHA type of protocols. In summary. A reading cycle initiates in this protocol by sending Query command from tag reader and all of tags that receive this command should participate in this cycle. Query command consists of one or several frames that each frame is composed of several slots. Tag reader shows value of Q parameter by sending this command then tags choose a slot with value between 0-12Q based on the Q value in order to send their generated RN16 to reserve channel for tag reader. Since the process of selecting slots’ values is a randomized process, there may be one of following situations for each slot:

- No tag has selected the slot.
- Two or more tags have selected the slot.
- Only one tag has selected the slot.

In first case, slot is free. There is collision in second case and these tags participated again in next frame within identification process. In third case, the slot is assigned to the tag in order to send its data for tag reader via the slot. There will be higher efficacy in this method if number of slots is almost equal to number of tags existing in the environment.

3.2- Structure of proposed authentication protocol

The proposed authentication protocol in this paper is based on the asymmetric cryptography in which, tag reader receives coded data by the tag then sends it to central server and this server decrypts this message and authenticates the tag. At next step, tag reader releases those data that should be written in tag, sends it to tag reader and tag reader sends it to tag. In fact, this is a mutual authentication protocol in which, tag and central server authenticate each other and tag reader is the only communicational factor between them. Structure of this protocol has been shown in figure 1.

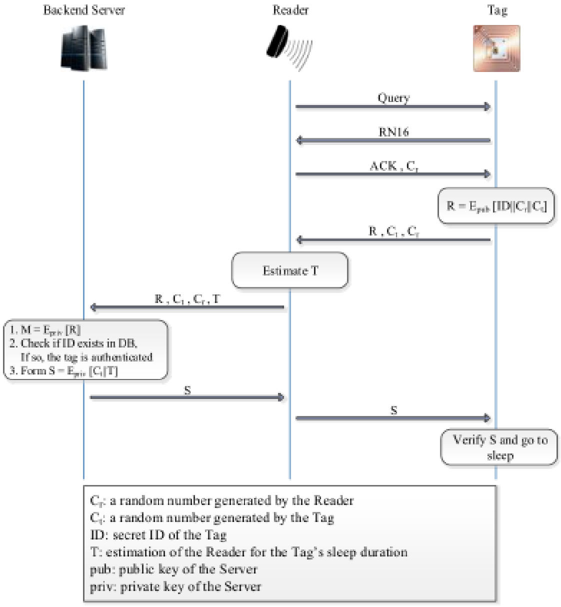
Backend Server
Reader
Tag
Query
RN16
ACK , C_r
R = E_{pub} [ID||C_r||C_t]
R , C_t , C_r
Estimate T
R , C_t , C_r , T
1. M = E_{priv} [R]
2. Check if ID exists in DB,
If so, the tag is authenticated
3. Form S = E_{priv} [C_t||T]
S
S
Verify S and go to sleep
C_r: a random number generated by the Reader
C_t: a random number generated by the Tag
ID: secret ID of the Tag
T: estimation of the Reader for the Tag’s sleep duration
pub: public key of the Server
priv: private key of the Server

Fig.1. Structure of proposed mutual authentication protocol based on public key cryptography

Tag reader sends a query command to tags the tag that gains the slot sends its generated RN16 to tag reader for channel reservation. Tag reader generates a stochastic value of C_r after receiving RN16 then sends it with ACK to tag. Then, the tag generates a stochastic value of C_t then encrypts it with public key of server along with its ID and C_r and then sends the generated R message with C_r to tag reader. The tag reader estimates sleep time of tag (T) then sends it along with R and two values of C_r and C_t to central server. Central server encrypts the received R using its private key. Now, the ID related to tag is found through a simple search in database having value of ID||C_r||C_t so the tag will be authenticated then the server encrypts C_t along with T using its private key then sends it to tag reader in order to send it for tag. The tag will open this message using public key after receiving it then sleeps for T duration. Following hypotheses have been considered for the mentioned protocol:

1. Communicational channel between tag reader and central server is secure. Therefore, this protocol only deals with threats related to the connection between tag and tag reader.
2. Tags will go to sleep after the end of mutual authentication process in order not to involve in next cycles.
3. Tag reader estimates duration in which, tag goes to sleep after receiving encrypted message from tag then sends it to central server along with message R and values of C_r

and C_t. this estimation can be done considering traffic model or the approximated distance between car and next tag reader.

3.3- Security analysis of authentication protocol

The main goal of proposed authentication protocol is to reduce complexity of calculations in central server. It means that central server does not require decrypting the received message for all of IDs existing in database since the message has been encrypted by public key of server; so, tag ID can be extracted by single decryption to authenticate the tag within a simple search through database. Because there is limited energy IoT and the objective is to reduce computational overhead and energy consumption. However, such reduction in overhead occurs if security is maintained. This protocol is assessed against potential attacks in urban traffic application in next part.

3.3.1- impersonation

An unallowable tag reader cannot access to important information of tags since these data are encrypted by public key of server so server can just decrypt the message using its private key. In this regard, no unallowable tag reader cannot introduce itself as an allowable component in system with assistance of an unallowable tag reader since IDs of tags are only under the authority of backend server and tags. When server encrypts the encoded message of tag and finds tag ID in database, it is ensured that this tag is allowable in system since the specified ID of tag is owned by itself and backend server.

3.3.2- Unallowable tag reading

An unallowable tag reader cannot access to data that are sent to tag reader by the tag. There is not any information freely published via media in this protocol except for stochastic challenges generated by tag and tag reader. Confidential tag ID is encrypted by public key o server and only backend server can open the message. In addition, attacker is not capable of imposing any kind of attack having randomized challenges.

3.3.3- changing the tag content

None of unallowable tag readers can change the information of tag by sending signal to it since the information received by tag from allowable tag readers has been signed by database. When tag opens message using public key of database, it ensures that the message has been sent by backend server without any change in it.

3.3.4- Tag tracking

These attacks are the most debatable challenges to application of radio frequency technology within various applications and large scales. None of two messages sent by tag is dependent to each other in this protocol. Every message encrypted by tag consists of a tag ID and two stochastic values that are different in each message; therefore, there is no way for unallowable tag reader to relate messages sent by a tag and to track the tag. However, such specification of backend server in O(1) time can authenticate the tag. In fact, strength point of this protocol is resistant against such attacks besides stable computational complexity in backend server, because conventional cryptography methods based on symmetric key cannot have these two specifications simultaneously.

3.3.5- Re-send attacks

Existence of challenge-response phase in this protocol prevents from resend attacks. For instance, we assume that one unallowable tag hears the message sent by an allowable tag then tends to send it to tag reader in another time. In challenge and response phase, tag reader generates a stochastic value (C_r) then sends it as challenge to tag reader. If the tag sends the heard message to tag reader, then tag reader will send it to backend server along with C_r. the server finds mismatch in the message after opening it; therefore, unallowable tag will not be authenticated.

3.3.6- Replay attacks

In replay attacks, an unallowable tag tries to disconnect an unallowable tag from a tag reader and exchange their messages through itself introducing itself as an allowable component in system. Figure 2 demonstrates such attacks in which, unallowable tag of T_2 receives challenge from tag reader then sends it to allowable tag of T_1 and assume that the tag out of the scope of tag reader is connecting to tag reader. At next step, T_1 encrypt its ID and sends it to T_2 then T_2 sends it to tag reader. Since this information has been encrypted by an allowable tag in system, the tag will be authenticated when tag reader transfers them into central server; in fact, unallowable tag has accessed to system.

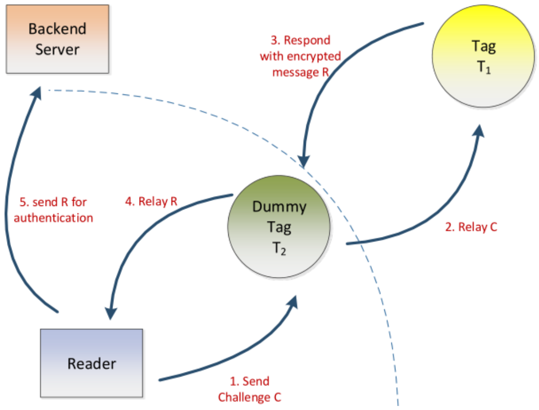

Fig. 2. A sample of man-in-the-middle attack

To cope with replay attack, delay in sending and receiving of packages can be used. In this regard, when tag reader sends its challenge value to tag then activate its timer and if the tag message reaches to it later than the time of threshold, then that message will be discarded and disconnected. Threshold level for package receipt depends on the bandwidth of wireless communicational lane and cryptography algorithm in protocol. Since specific values have not been determined for mentioned cases and used values have been chosen just for comparison in simulations, determination of threshold was not considered in this study.

3.4- Selection of cryptography algorithm

Asymmetric cryptography algorithms are robust tools to protect security of systems. Nevertheless, it is a challenge to use these algorithms in Radio Frequency Systems due to their limited resources. These algorithms have time-consuming and heavy calculations making their use in radio ID problematic but they have been considered in authentication process in radio ID due to their asymmetric keys and operations. Hence, lighter algorithms have been designed to implement in these systems.

Two versions of ECC [14] and Rabin [15] algorithms have been applied in this paper for simulation. ECC algorithm has been designed based on the suede structure of elliptical curves on finite fields; the strength point of this algorithm is small key length leading to optimal action in sending keys and messages but its shortcoming is slow cryptography operations. Like popular RSA algorithm, Rabin algorithm is associates with prime factorization. The advantage of this algorithm is high-speed cryptography process. One of disadvantages of this algorithm is slow process of digital signature that is almost similar to RSA in this case but since digital signature is done just by backend server in proposed protocol, it is not concerned as a disadvantage in this system. Another disadvantage of this algorithm is four responses for encryption process; it means that it is not clear that which one of responses is the main encrypted text; but since backend server has the main text of message requiring only one comparison for tag authentication, this weakness does not make any problem.

Table 1 compares key lengths of RSA, Rabin and ECC algorithms besides asymmetric encryption algorithm of AES based on computational attempt for their decryption. As it can be seen, ECC algorithm that has smaller key length compared to Rabin and RSA provides the same security.

We used a version of Rabin algorithm with key lengths of 512 and 1024 bit besides a version of ECC algorithm with key length of 233 bit for simulation process. Table 2 indicates speed of encryption and digital signature operations with 16MHz processor for this algorithm. As it is seen, encryption operation is done more rapidly in Rabin algorithm compared to ECC algorithm, while the result is vice versa in terms of digital signature so that ECC algorithm indicates better performance. Since encryption and authentication operations are done by tag and encryption and signature are implemented by server in our protocol, Rabin algorithm seems more appropriate for such application.

Table 1. Comparison of key lengths of encryption algorithms based on bit [16]

AES	ECC	Rabin	RSA
۵۶	۱۱۲-۱۵۹	۵۱۲	۵۱۲
۸۰	۱۶۰-۲۲۳	۱۰۲۴	۱۰۲۴
۱۱۲	۲۲۴-۲۵۵	۲۰۴۸	۲۰۴۸
۱۲۸	۲۵۶-۳۸۳	۳۰۷۲	۳۰۷۲
۱۹۲	۳۸۴-۵۱۱	۷۶۸۰	۷۶۸۰
۲۵۶	۵۱۲+	۱۵۳۶۰	۱۵۳۶۰

Table 2. Speed of cryptography and digital signature operations based on ms with 16MHz processor

	Rabib-512	Rabin-1024	ECC-233
Cryptography/authentication	0.09	0.34	11.73
Decryption/signature	34	162	6.86

3.5- Sleep algorithm for tags and tag reader

At the end of authentication process of a tag, if the tag is still in the area of tag reader after receiving query for tag reader, it will reserve the channel to create connection with tag reader. This issue represents itself in moving things with low displacement such as heavy traffics that tags are present longer in the area of tag reader so this case reduces opportunity for channel reservation and connection with ta reader for other tags that have not been authenticated. Hence, when a tag is authenticated by central server receiving the last message from tag reader, it will go to sleep in order not to participate in next cycles. On the other hand, we know that radio ID tags are active types containing an internal battery for calculations and making connection with tag reader. Battery life is limited between 3 and 8 years. The method of sleeping tags not only improves efficacy of system but also increases battery life of tags; on the other hand, it is beneficial for things that have finite energy in IoT expanding lifespan of such things. Putting tag reader to sleep under specific circumstances increases battery life. In general, tag reader should go to sleep when there is not any tag in area of tag reader (when the function of tag reader is useless) in order to save energy. Hence, tag reader should be awaken from sleep within constant time intervals and search in environment for tags then go to sleep in case of non-identification of tag.

4. Proposed method

To examine the proposed protocol in this research, a simulation scenario was designed to see the effect of different algorithms on system's performance. For this purpose, it is considered that vehicles pass through a wide road with several lanes assuming that a tag reader with height of 5m from the land level exits at the middle of the road in order to read tags of passing vehicles and identify them.

Since vehicles are passing with different speeds, limitations of these tags make tag reader unable to identify many of cars that are passing with low speed. In addition to speed of vehicles that makes them be less present in area of tag reader, use of security methods imposes high overhead to the system. So, conducted simulations showed that inactive tags do not have appropriate performance in this application. Therefore, all of simulations in this study have been done by concerning active tags only.

In order to have appropriate estimation of real conditions of traffic in roads, three models of light, medium and heavy traffic were used for simulation, because as we know density of vehicles and their speed in different hours of day is different in urban and suburban roads. Therefore, we have tried to model different traffic conditions in day and night hours by using above-mentioned movement models. Heavy traffic model is when there is a high density of passing vehicles with low speed. In This model, there is large number of tags in area of tag reader in each moment and each tag spends more time in area of tag reader. In medium traffic model, density of vehicles is lower while their speed is higher. In this traffic model, vehicles are moving lightly. Speed of vehicles has been considered equal to 40-50Km in simulations of this model indicating an ordinary traffic. Light traffic model is the worse situation since density of vehicles is lower while their speed is higher than other models. In this model, tags are in area of tag reader for short time so their opportunity for being read will be reduced. Therefore, it is anticipated that more percent of tags in heavy traffics will be identified by tag readers. Properties of each model have been indicated in table 3. Model was implemented through MATLAB 2017 Software.

Table 3. Properties of three assumed traffic models

Property Movement model	Speed (m/s)	Distance between vehicles (s)	Number of lanes	Density of cars (vehicle/mlane)
Light	22-42	0.10	5	0.10
Medium	11.14	0.25-0.5	5	0.16
Heavy	1.5	3	6	0.22
• Width of each lane is 3				

Density of vehicle is obtained using following formula:

1) $\rho = \frac{N}{\pi \times r^2} \times w_{lane}$

Where, ρ is density of vehicle in each meter of lane, N is number of vehicles in coverage area, r is radian of coverage area, w_{lane} is the width of each lane of parameter N that is average of vehicles existing in environment at each moment that has been measured for each model. For instance, this parameter obtained to 23 in heavy traffic model.

Physical parameters applied in this implementation are described in table 4.

Table 4. Physical parameters in research simulation

Type Properties	Tag	Tag reader
Consuming power	40 mW	10 W
Sent output power	-10 dBm	0 dBm
Receiving sensitivity	-70 dBm	-82 dBm
Type of antenna	Omni directional	
Height of antenna	1 m	5 m
Frequency	UHF (900 MHZ)	
Diffusion model	Small-scale Shadowing	

	n=4, σ=2
Radian of coverage area	10 m

4.1- Results of use of Rabin-512 encryption algorithm

This part of paper examines results obtained from simulation of authentication protocol that has been implemented by Rabin encryption algorithm with 512-bit key length. It should be noted that Rabin encryption algorithm with such key length does not provide a relatively high security for system. This study aimed at comparing simulation results with different parameters and algorithms in order to obtain a general viewpoint about conditions to make decision about security issues of system to select the best algorithm and parameter based on the type of application and considerations.

4.2- Fraction of tags read

Figure 3 illustrates the tags that have been identified by tag reader in heavy traffic model. In this diagram, horizontal axis indicates the delay of close commutation between tag reader and backend server based on m/s and vertical axis indicates those tags that have been authenticated successfully. Diagrams have been shown for 4-bit rate modes (from 128kbit/s to 1megabit/s) between tag and tag reader. As it is seen, the more delay in commutation between tag reader and backend server, the lower percent of tags can be identified by system. Such result is expected because when channel is reserved for longer period of time then other tags have less opportunity for identification.

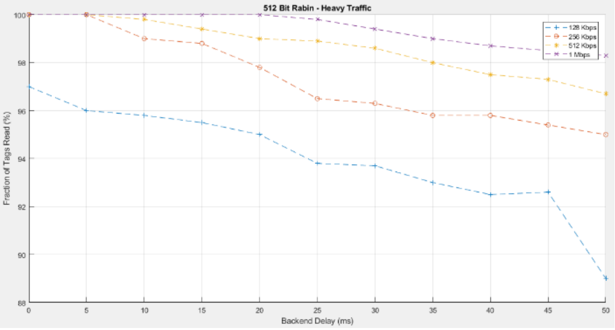

Fig. 3: Fraction of read tags in heavy traffic model with Rabin-512 algorithm

In addition, it is seen in this figure that the more the bandwidth of wireless channel between tag and tag reader, the higher percent of tags will be identified when passing through tag reader. The reason is that higher bandwidth reduces delay in tag reading; therefore, other tags that exist in the area have more opportunity to connect with tag reader. As it can be seen in 1 megabit/s rate even for longer delays, high percent of tags are authenticated successfully.

Fraction of tags read for medium and light traffic models have been shown in figures 4 and 5, respectively. Figure 4 shows that there will be an increase in fraction of missed tags rapidly with increasing delay in connection between tag reader and backend server and this issue is more intensified for light traffic model. In medium and light traffic models, tags pass through tag reader more rapidly; therefore, there will be a higher percent of tags exiting from the area of tag reader without being identified compared to heavy traffic model. Such difference is higher in longer delays. Moreover, there is a relatively good performance in terms different bit rates in medium traffic model with 256kb/s bandwidth and higher for shorter delays. In light traffic model, it is possible to identify 95% of tags only with 512kb/s rates and above at maximum level of 5m/s delay.

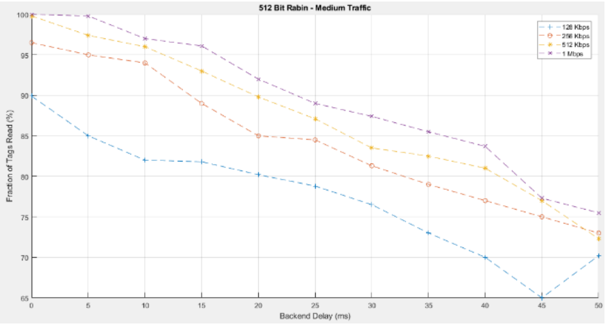

Fig.4: Fraction of read tags in medium traffic model with Rabin-512 algorithm

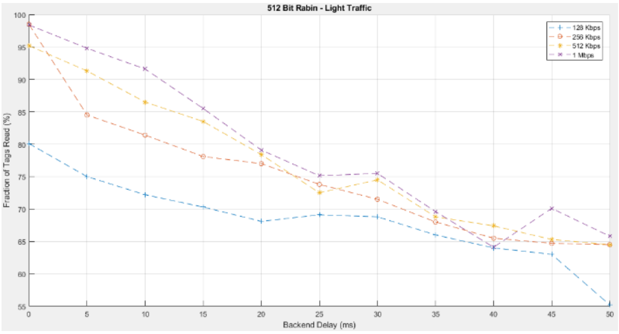

Fig. 5: Fraction of read tags in light traffic model with Rabin-512 algorithm

4.3- Energy consumption of tags

We used the idea of sleeping tags after successful authentication in system simulation. In this case, when a tag sleeps after the end of authentication then it not only saves the energy but also increases the opportunity for other tags to be read in the area. Table 5 demonstrates activity percent of a tag during the time it is in the area covered by tag reader for different delays considering 1mb/s bandwidth. As it is seen, tags are active only about 20-26% of their presence in the area of tag reader so most of the time they are slept.

Table 5. Effect of tags sleep in different traffic models assuming 1mb/s bandwidth

Parameter Traffic model	**Effect of tag read (ms)**	**Presence duration in coverage area (s)**	**Fraction of awake time to total tine of presence in coverage area for different delays**		
			0ms	**25ms**	**50ms**
Light	1.7	0.718	20.27%	21.72%	25.16%
Medium	1.7	1.612	20.18%	21.03%	20.92%
Heavy	1.7	13.333	25.67%	25.96%	25.78%

Effect of successful read for a tag is measured to 1.7ms concerning 1mb/s bandwidth using Rabin-512 cryptography algorithm without considering delay in commutation between tag reader and server. In fact, this delay is equal to time distance between receiving a request from tag reader to receiving the last message from tag reader at the same cycle and end of the mutual authentication. In addition, in simulations, average time of presence of tag in area of tag reader reported to 13.334, 1.612 and 0.718 seconds for heavy, medium and light traffic models, respectively. According to table 5, in light traffic model, increase in delay between tag reader and backend server with subsequent increasing delay in authentication operations leads to increase in activity percent of tag in the area. This case is not seen in medium and heavy traffic models, because fraction of delay in reading a tag to presence time of tag in the area of tag reader is minor in these traffic models. Therefore, there is not a significant difference in awake duration of tag in longer delays between tag reader and server.

4.4- Energy consumption of tag reader

All of presented simulation results are related to the situation in which, tag reader is always active. We put tag reader to sleep when tag reader is inactive and there is no tag to be read in environment in order to save the consuming energy of tag reader. Since there may be different traffic models in practice, an algorithm should be used to identify the model type and sleep duration for tag reader in the model. This parameter may be different values in each traffic model so the model can be identified based on this value. Table 6 indicates values of this parameter in different traffic models. If it is assumed that all of tags in environment are read, it means that rate of tags entrance to the environment is almost equal to the rate of tags reading by tag reader. Entrance rate of vehicles to the environment is calculated by following formula:

2) $\lambda = N_{lane} \times \rho \times V$

Where, λ is entrance rate of vehicles (tags) to the environment, N_{lane} is number of lanes in coverage area, ρ is density of vehicles per meter in each lane, V is average speed of vehicle.

The difference between numbers of read tags per second in different traffic models is due to idea of putting tags to sleep, because there would not be any difference between values of this parameter in different movement models if tags were not put to sleep after being read.

According to table 6, although there is large number of vehicles in environment in heavy traffic model, but entrance rate to the reading area in very low and since number of read tags per second is determined based on the entrance rate, value of this model is low in heavy traffic model. The case is completely different in light traffic model since there is few numbers of vehicles in environment per second while entrance and exit rates are high. Hence, number of read tags per second is more in this traffic model. In this part, simulation results are examined when we can put tag reader to sleep.

Table 6. Number read tags per second in different traffic models

Traffic model **Specification**	**Light**	**Medium**	**Heavy**
Number of read tags per second	18-22	11-14	1-3

In figure 6, read tags in heavy traffic models have been shown in different bitrates. Data on the diagram have indicated percent of sleep time of tag reader for each delay and bit rate. For instance, tag reader has been slept 80.06% of its time with1mb/s rate considering 0ms delay; however, it has identified more than 97% of tags. As it is obvious, with increase in connection delay between tag reader and backend server the tag reader has less to time for sleep. Moreover, the lower the bit rate, the more active the tag reader will be with lower sleep time.

Diagrams indicate that in high bit rates even for relatively longer delays tag reader has more time for sleep compared to lower bit rates; in addition, more tag readers are identified in this case. For instance, tag reader is slept 48.76% of time and identifies more than 93% of tags in case of 1mb/s bandwidth and 40ms delay while in case of 128kb/s bandwidth without considering close commutation delay between tag and tag reader (0ms), tag reader is slept only about 44.31% of time.

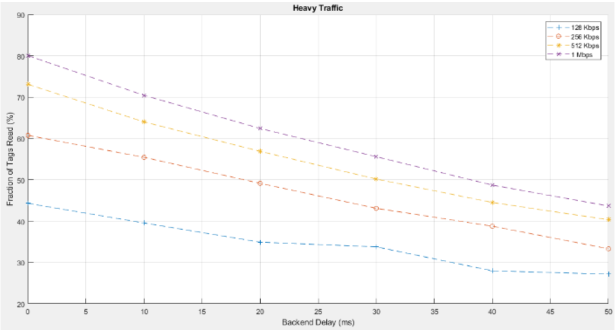

Fig. 6: Fraction of tag reader sleep in heavy traffic model in different bit rates

Figure 7 shows effect of tag reader sleep in different traffic models. 1mb/s rate has been considered for comparison. System’s behavior is similar in other rates and the only difference is in values. As it can be seen in heavy traffic model, increase in delay does not lead to reduction in read tags percent. While, such reduction is more and rapid in medium and light traffics. These observations are same when tag reader does not apply sleep algorithm and as it was mentioned, the reason is that tags are in the area of tag reader for longer tile in heavy traffics so tag reader has enough time to identify them. Curves indicated in figure 7 also demonstrate that tag reader spends longer time for sleep in heavy traffic model; this fraction is lower in medium and light traffics because there heavier the traffic, the lower movement in environment and speed of vehicles entrance and exit. In this case, tag reader has more time to read tags and put to sleep longer. In contrary, movement speed is high in light traffics and if tag reader is putting to sleep for more time, then some tags pass through it without being successfully read by ta reader.

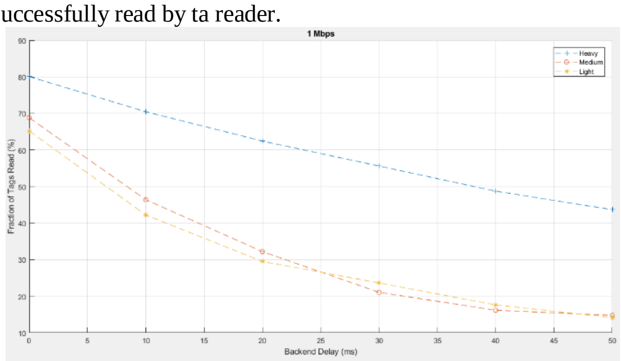

Fig. 7: Fraction of tag reader sleep in heavy traffic model in different bit rates

The considerable point here is that although it is expected in heavy traffics that more percent of tags be identified compared to light traffic, but all of tags are successfully authenticated in medium traffic model for no delay while 97% of tags have been read in heavy traffic models. Nevertheless, in heavy traffics model with increasing delay, there is not any considerable reduction in read tags while there is considerable reduction in other models. Larger numbers of tags have been authenticated in medium traffic model with no delay since tag reader has been active for longer period. In other words, in heavy traffic model, tag reader has been sleeping 80.06% of time so a percent of tags has passed through the area without being authenticated.

Considering the figures and explained issues, this question is raised that what is the relationship between fraction of tag reader activity and fraction of tags read. In fact, it can be stated that there is an agreement between read tag and fraction of tag reader sleep. This relationship has been shown in this study for different traffic models considering 1mb/s rate and different delays. For this purpose, parameters related to sleep algorithm of tag reader have been changed in a way to find the relationship between these two fractions.

As it is seen in figure 8 for heavy traffic model, tag reader can sleep longer in low delays between tag reader and backend server but identify high percent of tags. For instance, tag reader could identify about 100% of tags in delays smaller than 10ms while it has been slept more than 60% of time. The longer the delay between tag reader and server, the lesser the time of tag reader for sleep.

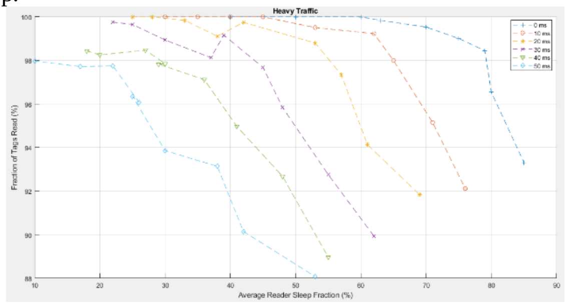

Fig.8: Agreement between fractions of read tags and tag reader sleep in heavy traffic models with 1mb/s bandwidth

Figure 9 indicates agreement between read tags and fraction of tag reader sleep in medium traffic model with 1mb/s rate for different delays. In medium traffic model in low delays also, tag reader can sleep more time while can still a high number of tags but is long delays, sleep time is reduced also number of authenticated tags is decreased due to long delay. It can be seen in figure 10 for light traffic model that in low delays, tag reader authenticates large number of tags while it has been slept longer. However, fraction of reading and sleeping in this traffic model is lower than medium and heavy traffics, in particular for longer delays.

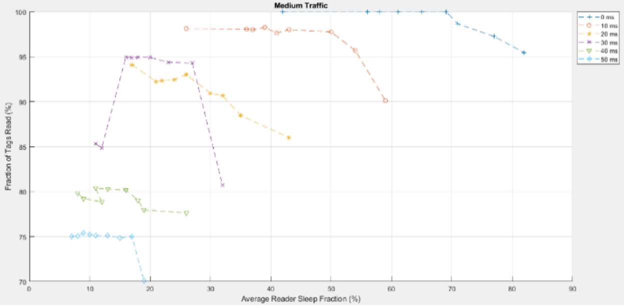

Fig.9: Agreement between fractions of read tags and tag reader sleep in medium traffic models with 1mb/s bandwidth

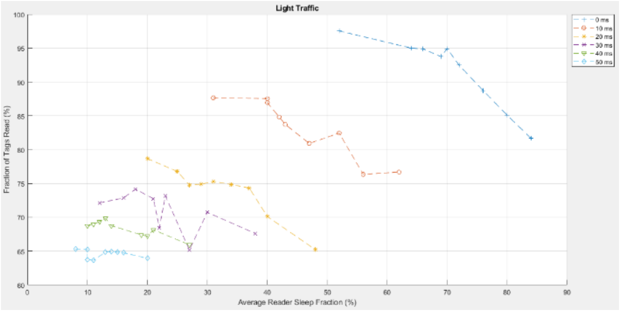

Fig. 10. Agreement between fractions of read tags and tag reader sleep in light traffic models with 1mb/s bandwidth

However, the most significant goal of any radio frequency system is to read all of tags existing in the environment. Therefore, sleep time of tag reader in different traffic models should be calculated in a way that maximum identifiable tags will be authenticated successfully.

4.5- Comparison of results obtained from symmetric and asymmetric algorithms

Symmetric cryptography algorithms like AES are more rapid than asymmetric cryptography algorithms in terms of speed of encryption and decryption operations. Table 6 compares speed of encryption and decryption operations of popular powerful AES-128 algorithm with three applied algorithms in this research. It should be noted that according to table 1, AES-128 provides a security almost equal to Rabin with key length of 3071 bit and ECC with key length of 256-383 bit in terms of computational power for decoding. Therefore, both encryption and decryption operations are more rapid in AES-128 compared to asymmetric algorithms.

Table 6: Speed of operations in symmetric and asymmetric cryptography algorithms based on ms

Algorithm **Model**	**Rabib-512**	**Rabin-1024**	**ECC-233**	**AES-128**
Cryptography/authentication	0.09	0.34	11.73	0.18
Decryption/signature	34	162	6.86	0.18

As it was explained, we used asymmetric cryptography algorithms to reduce computation burden of backend server. In other words, we aimed at reducing computational load in server by increasing computational and communicational load in tag.

To see the effect of reduced computational load of server on system performance, we implemented a protocol similar to proposed protocol but using AES-a28 cryptography. Length of stochastic challenges and tag ID were considered equal to 128bit for this protocol. Therefore, the tag requires three times of cryptography operation to encrypt these values. In addition, processing power of backend server considered to 10GHTz assuming that data related to 10 million tags exist in database. Since the tag encrypts information with its relevant common key, decryption should be dome for all of IDs and keys existing in database in order to identify the considered tag. Since the tag data may be at the beginning or end of database table, server should perform n/2 decryption operations to identify the tag. In our system, decryption action should be done 15 million times by the server (3 decryptions for each ID existing in table 3). Since generation of each 128-bit block in AES-128 algorithm almost requires 1000-cycle hour [17], each decryption operation requires a time of 100ns. Therefore, there will be 1.5 seconds delay averagely in backend server to identify each tag in backend server. Conducted simulations indicated that almost none of tags are successfully authenticated by the server in light and medium traffic models with 1mb/s bandwidth between tag and tag reader because presence time of tag in each coverage area of tag reader for these two traffic models is averagely equal to 0.718 and 1.612 seconds. Therefore, the tag will exist from the coverage area before receiving signed message by the server.

5. Conclusion

This research aimed at presenting a mutual authentication protocol based on the public key cryptography in radio frequency systems for traffic handling. The main idea of this protocol was reducing computational complexity in backend server. In applications of traffic management, there may be numerous requests for authentication from different tag readers to backend server at each moment leading to a challenge for these systems. On the one hand, limited sources in these systems make problem in application of powerful algorithms for public key cryptography. On the other hand, movement in system and need for rapid identification of tags adds time limit as another challenge to the system. Examinations indicated that it is not possible to use conventional public key algorithms like ElGamal, RSA and ECC with existing conditions in these systems directly. Therefore, we used an improved version of ECC algorithm and Rabin cryptography for authentication protocol. In order to evaluate performance of system through implementing this protocol using cryptography algorithms, we used simulations scenarios of studies [14] and [18]. There are some solutions and recommendations to implement a radio frequency system for high-motion applications concerning conducted examinations and obtained results since the proposed authentication protocol in this research can cope with some of common attacks in radio frequency systems. since information are encrypted in tag by public key of server and tag IDs are under the supervision of backend server, no existence in this system can forge the identity of another existence. In addition, since the information that should be written in tag should have been encrypted just by private key of server, no unallowable tag reader in system can change the content of tags. Security against attacks to tag tracking is one of the most prominent advantage of this protocol because the protocol has been designed to protect privacy of tags and reduce computations in backend server form O(n) to O(1). However, application of symmetric key cryptography methods to protect privacy of tags would lead to heavy computations with O(n) order in server. Moreover, resend attack will not occur due to presence of challenge-response phase in this protocol. It is possible in this protocol to prevent from replay or man-at-the-middle attacks.

According to the results obtained from simulations, we concluded that the higher the security level of cryptography algorithm, the heavier the cost for suitable system performance we should accept. Using Rabin-512 cryptography with low bit rates, we could identify high percent of tags in environment so that with a 128kb/s rate for 10ms delays between tag reader and server we could identify more than 90% of tags. However, application of Rabin-1024 and ECC-233 reduced reading rate for delays above 10ms to 90% in heavy traffic model. The higher the bit rate is used, the better the system performance in tag identification in environment will be. This case is more effective in Rabin-1024 algorithm due to the main disadvantage of this algorithm that is big block of data. Hence, this problem can be solved to some extent by increasing bit rate. ECC cryptography algorithm uses smaller data blocks compared to Rain so bandwidth problem is less when using this algorithm. In summary, the problem in use of Rabin algorithm with large key length is related to limited bandwidth of wireless system between tag and tag reader while problematic application of ECC algorithm is limited computational power of tags. Therefore, it is essential to use an algorithm with better performance considering system properties.

6. Further studies

There are other cryptography public key-based algorithms named NTUR that have small data blocks and provide rapid cryptography operations. Cryptography operations are done more rapidly and lighter in NTRU compared to Rabin algorithm. It is possible to implement this algorithm for inactive tags but it is not recommended to use these tags in applications with high movements.

References

[1] Atzori, L., Iera A., Morabito G., (2010). The internet of things: a survey. Comput Netw 54(15):2787–2805

[2] Chen D, Chang G, Sun D, Li J, Jia J, Wang X (2011) TRM-IoT: a trust management model based on fuzzy reputation for internet of things. Comput Sci Inf Syst 8(4):1207–1228

[3] He D, Zeadally S (2015). Ananalysis of RFID authentication schemes for internet of things in health care environment using elliptic curve cryptography. IEEE Internet Things J 2(1):72–83

[4] Nguyen KT, Laurent M, Oualha N (2015) Survey on secure communication protocols for the internet of things. Ad Hoc Netw 32:17–31

[5] Peter Schaar, Working document on data protection issues related to RFID technology, Working Document Article 29 -10107/05/EN, European Union Data Protection Working Party, January 2013

[6] Auto-ID Center. 900 MHz class 0 radio frequency (RF) identification tag specification. Draft, March 2003

[7] M. Ohkubo, K. Suzuki, and S. Kinoshita. Cryptographic approach to \privacy- friendly" tags. In RFID Privacy Workshop, 2015

[8] D. Molnar and D. Wagner. Privacy and security in library RFID: Issues, practices,chiu c.tan jie wu security rfid networks and autentication2013

[9] Tewari A, Gupta BB (2016) Cryptanalysis of a novel ultra-lightweight mutual authentication protocol for IoT devices using RFID tags. J Supercomput

[10] Chiu C. Tan, Jie Wu. Security in RFID Networks and Communications. Chapter-10 from book Wireless Network Security, Springer; 2013. p. 247-267.

[11] M.O. Lehtonen, F.Michahelles, E.Fleisch. Trust and Security in RFID-Based Product Authentication Systems.IEEE Systems Journal; 2007, vol. 1, No. 2, p. 129 – 144

[12] Gaochao Li, Xiaolin Xu, Qingshan Li. LADP: A lightweight authentication and delegation protocol for RFID tags. Ubiquitous and Future Networks (ICUFN), 7th International Conference, IEEE; 2015. p. 860 – 865.

[13] J. Fu, C. Wu, X. Chen, R. Fan, L. Ping. Scalable pseudo random RFID private mutual authentication. 2nd IEEE International Conference on Computer Engineering and Technology (ICCET). V. 7; 2010. p. 497-500.

[14] K. C. Loi and S.-B. Ko, "High performance scalable elliptic curve cryptosystem processor for Koblitz curves," Microprocessors and Microsystems, vol. 37, pp. 394-406, 2013.

[15] G. Gaubatz, J.-P. Kaps, E. Ozturk, and B. Sunar, "State of the art in ultra-low power public key cryptography for wireless sensor networks," in Pervasive Computing and Communications Workshops, 2005. PerCom 2005 Workshops. Third IEEE International Conference on, 2005, pp. 146-150.

[16] L. Zhu, K. Lauter, and K. Jaganathan, "Elliptic Curve Cryptography (ECC) Support for Public Key Cryptography for Initial Authentication in Kerberos (PKINIT)," 2008.

[17] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, "Strong authentication for RFID systems using the AES algorithm," in Cryptographic Hardware and Embedded Systems-CHES 2004, ed: Springer, 2004, pp. 357370.

[18] Ema Jome, E. (2013). Evaluation of effect of security methods on efficiency parameters of RFID system for application of violations record and urban traffic management, MSc thesis in Information Technology Engineering (Computer Network Orientation), Science and Industry University of Iran, Computer Engineering School.