2018 IEEE Intelligent Vehicles Symposium (IV)
Changshu, Suzhou, China, June 26-30, 2018

A Security Model for Intelligent Vehicles and Smart Traffic Infrastructure

Sachin Kumar*, Sonu Jha*, Sumit Kumar Pandey†, Anupam Chattopadhyay*
*Nanyang Technological University, Singapore. {sachinkumar, skjha, anupam}@ntu.edu.sg
†Ashoka University, India. emailpandey@gmail.com

***Abstract*—Intelligent vehicles require to communicate with other vehicles as well as with the road-side infrastructure for gathering information such as traffic management, cooperative driving, telematics and road construction. However, vehicle-to-vehicle (V2V) and vehicle to infrastructure (V2I) communications can impose some serious security threats against vehicles' safety and other sensitive information which can lead to catastrophic consequences. Therefore, there is a pressing need to develop appropriate security protocols facilitating V2I and V2V communication. This paper presents a model for smart traffic infrastructure consisting of numerous entities like smart sensors, intelligent vehicles, base stations along with a new user authentication and key-exchange protocol which aids in the establishment of a secure session for communication between the entities. In the proposed protocol, any SUF-CMA (Strong Unforgeable- Chosen Message Attack) secure digital signature algorithm can be used for authenticating the users and any NM-CCA2 (Non-Malleability Chosen Ciphertext Attack) secure algorithm can also be used for the challenge response phase between the users authenticating themselves.**

***Index Terms*—User Authentication Protocol, Key-Agreement, Public-Key Cryptography, Cyber Security, Intelligent Vehicles, Authenticated Key Exchange.**

I. Introduction

For more than a decade, development of intelligent vehicle technology received great interest in academia as well as in industry. Basically, intelligent vehicles are required to communicate with everything such as in-vehicle, vehicle-to-user device, vehicle-to-vehicle communication, vehicle to infrastructure unit etc. For example; Vehicle-to infrastructures (V2I) communications is mandatory for exchanging the data between vehicles and roadway infrastructure such as traffic signals, work zones, and speed-limit.

Multiple corporate houses including Google and Tesla have come up with their own autonomous vehicle (AV) propositions. In an effort towards the standardization of AV, SAE International, identified six categories – ranging from No-Automation (Level 0) to Full Self-Driving Automation (Level 5) [4]. Though predominantly in test-mode, these vehicles are expected to be on road within a few years, which calls for an immedieate attention to their security issues. To that effect, there is a lack of standardized cyber security protocols for communications among the intelligent vehicles, the traffic monitoring systems and the base stations (which analyses the data sent by the traffic monitoring systems). Intelligent vehicles such as Google driver-less car communicates with Automated Traffic Monitoring Systems (ATMS) for directions and guidelines; like speed limit, school zones, road construction and traffic density. An ATMS uses cameras for road traffic monitoring to observe particulars and facilitates necessary traffic information to be available in places like traffic signals and bus information display as depicted in Figure 1. Furthermore, ATMS plays a crucial role in traffic monitoring and is considered one of the major components in smart city and secure Internet-of-Things (IoT) infrastructure [6].

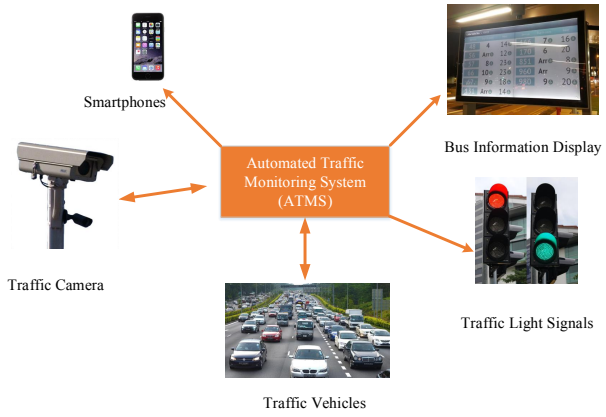

Figure 1: Vehicle-to-Infrastructure (V2I) communication

In the scenarios of smart sensors handling traffic management systems or automotive/intelligent vehicles running on the roads, there is a requirement that those sensors are able to communicate with ATMS/Base Stations or other sensors in real-time in order to make real-time decisions and share data. Similarly autonomous vehicles (AV) should also be able to communicate with other AV, smart devices and ATMS in real time to make correct decisions. Here, a major concern comes is whether these sensitive communications are secure or not. If there are security loop-holes in such systems, any potential hacker or malicious adversary can gain unfair advantage. Experimental studies have been reported in literature where traffic systems have been breached to achieve some unfair advantage over the system [14], and where the concerns on the different security issues in intelligent vehicles have been

978-1-5386-4451-5/18/$31.00 ©2018 IEEE

raised [7] [9]. In Figure 2, we show example of dynamic and static entities (vehicles, drones, cameras, base stations etc.) which requires real time communication. These issues can only be comprehensively tackled with a proper security protocol in place, which considers both the static and dynamic components' participation. This forms the prime motivation of this work.

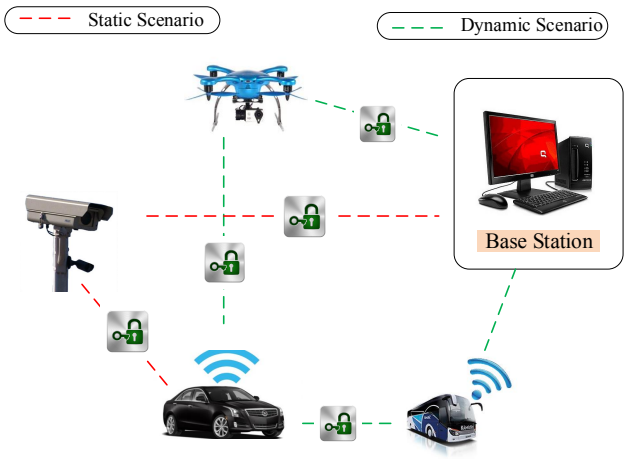

Figure 2: Dynamic and static communications between the entities.

Organization of the Paper

This paper is organized as follows. Section II shows examples of security threats and related works in order to resolve the threats. In Section III, we propose a scheme for user authentication and key agreement which works well in dynamic communication scenarios and is also suitable to work on constrained environment. Finally in Sections IV and V, we discuss on the security aspects of our scheme followed by concluding the paper.

II. Related Works

For communications to be secure, the most important security features to be taken care of are Authentication, Data Confidentiality (secrecy) and Data Integrity (message authentication). The team, led by the University of Michigan computer scientist J. Alex Halderman, said that the networked traffic systems are left vulnerable to three major weaknesses [1], [3] out of which one of them is a poor authentication mechanism which is the use of factory-default usernames and passwords. The use of username-password combination is a common way to authenticate any user. But, it does not provide security during a remote login session or particularly when the communication channel is wireless. In a wireless channel, an attacker can collect all transcripts and then can extract username-password combination easily. For user authentication, the protocols can be static (passwords, PIN, etc.) and dynamic (random nonces, challenge-response, etc.). In [15], authors propose a provably secure password based protocol using the public-key schemes. Similarly there have been proposal of other password based protocols such as [12], [16], [17]. However, an inherent disadvantage of password based authentication is the huge memory requirement when there are several nodes in the system making the situation more dynamic. In password-based authentication approach, an authentication server needs to store passwords for users who want to authenticate. Not only that, the server must have a public-private key pair and each user needs to store the digest of public key of the server. In the dynamic scenario; there will be a rapid authentication requirements within multiple nodes (such as V2V, V2I etc) during communications. All nodes are required to store passwords (usually hash of the passwords) of the other nodes of the system as to authenticate the users. This would be an overhead in such system where the number of nodes keeps increasing. Apart from password based approaches, there are existing authentication schemes based on challenge-response in the literature. In [5], Kerberos Protocol [2] based scheme was developed. However it requires the involvement of a third party (trusted) each time when a communication session needs to be established between two parties. This becomes an overhead in dynamic scenarios discussed in our model where numerous rapid communications needs to be established in parallel between several parties, and in each of those communications, involvement of a trusted third party increases overhead and cost.

III. Proposed Model for Secure Smart Traffic Infrastructure and Intelligent Vehicles

This section presents our proposed model for secure smart traffic infrastructure which consists of various entities/nodes. In such kind of infrastructures, the underlying entities need rapid real time communications from other entities as to make important decisions. For instance; a smart vehicle of this system needs rapid communication with other smart vehicle, user devices or even base stations in order to decide various things such as making decisions about the correct paths or whether the correct traffic laws are being followed while roaming around the roads etc. Similarly there are other entities such as smart cameras which observes the traffic and captures videos/images whenever they find some vehicle violating some traffic laws (such as illegal parking, illegal line crossing etc.). In such kind of infrastructure where everything is dependent on the communications between the entities, security becomes a natural and important concern. On top of that, implementation of these security parameters should be lightweight and efficient. Keeping these concerns in mind, we propose some solutions which are intended to provide not only secure communications between the entities present in these model, but also making sure that the solutions provided henceforth are lower in cost and efficient. Our solution consists of a user authentication scheme, which helps making sure that the communication requests made from one party to another

are genuine (ensuring that the party trying to communicate is genuine and not some malicious adversary), a key agreement scheme after which a secure session can be established for the communicating parties to share data. In order to achieve these needs, a protocol for user authentication and key agreement is developed as explained in the following subsections.

A. Preliminaries

Our solution employs a user authentication scheme followed by key agreement which results in the establishment of a secure channel. Once such channel is established, any secure symmetric key authenticated encryption scheme can be used to provide both message authentication and secrecy. This idea is depicted in Figure 3.

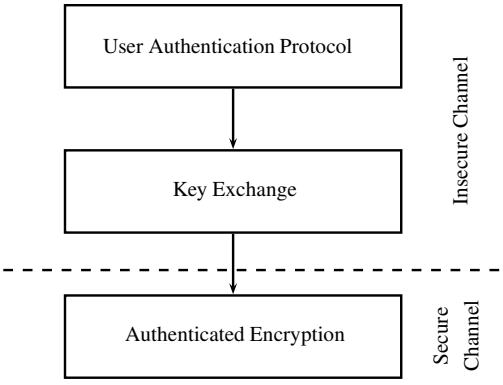

Figure 3: Flow of Security Mechanisms

1) Trusted Vendor(s) - Each vendor will have a pair of public-secret key pair generated from a Signature scheme. The secret key of the vendor will be kept secret whereas public key will be available to each entity. Moreover, a trusted vendor will be responsible for producing an authentication card (as shown in Figure 4) for each entity which contains a pair of public-secret key of the entity, a vendor number (each vendor will be given a number), a signature on the public key generated by the vendor corresponding to its vendor number, and a list of vendor numbers and corresponding public keys. Summary of these are given below:

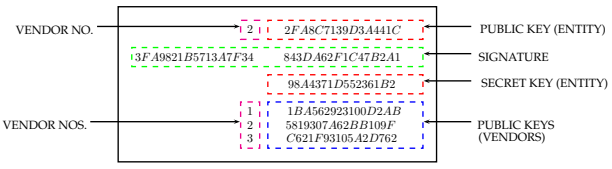

Figure 4: A Sample Authentication Card for Our Proposed Protocol

   a) Vendor Number - If there are t vendors, each vendor will be assigned a unique number from the set $\{1, \cdots, t\}$.
   b) Public Key (Vendor) - Each vendor will have a pair of public-secret key pair generated from a Signature scheme. The secret key will be kept secret by the corresponding vendor whereas public key will be available to each entity through its authentication card. The authentication card keeps a list of all vendor's public keys along with their vendor numbers. These public keys of vendors will be used by each entities during the verification of the public key of communicating entities.
   c) Public Key and Secret Key (Entity) - Each entity will be given a public-secret key pair generated from the Encryption scheme. These public and secret keys will be used during encryption and decryption of the nonces used in the challenge-response method as a part of the user authentication protocol.
   d) Signature - A signature is generated on the public key of each entity through the *SIG* algorithm of the Signature scheme. This algorithm takes the public key of the entity, secret and public key of the vendor as inputs and return signature on the public key of the entity as output.

2) Signature scheme - A Signature scheme consists of three algorithms - (i) `SETUP`, (ii) `SIGN` and (iii) `VERIFY`. Notationally, we write `SIGN` as *SIG* and `VERIFY` as *VER*. A Signature schemes is described as follows:
   a) *SETUP(λ)* - It is a probabilistic polynomial time algorithm that takes λ (a security parameter) as an input and returns public-secret key (PK, SK) pair as output.
   b) *SIG(m, SK, PK)* - It is a probabilistic polynomial time algorithm that takes a message m, secret key SK and public key PK as inputs and returns a signature σ on the message m.
   c) *VER(m, σ, PK)* - It is a deterministic polynomial time algorithm that takes a message m, a signature σ and public key PK as input and returns true if σ is a valid signature on the message m corresponding to public key PK, otherwise outputs false.

   Our protocol takes a public key of an entity as a message for the Signature scheme. The algorithm *SIG* is run by vendors only at the time of manufacturing of authentication cards whereas the algorithm *VER* is run at entity's end whenever the verification of public key of an entity and its signature is required.
3) Public Key Encryption scheme - A Public Key Encryption scheme consists of three algorithm - (i) `SETUP`, (ii) `ENCRYPT` and (iii) `DECRYPT`. Notationally, we write `ENCRYPT` as *ENC* and `DECRYPT` as *DEC*. A Public Key Encryption scheme is described as follows:
   a) *SETUP(λ)* - It is a probabilistic polynomial time algorithm that takes λ (a security parameter) as an input and returns public-secret key (PK, SK) pair as output.

b) *ENC*(m, PK) - It is a probabilistic polynomial time algorithm that takes a message m and public key PK as inputs and returns a ciphertext c on the message m.

c) *DEC*(c, SK, PK) - It is a deterministic polynomial time algorithm that takes a ciphertext c, secret key SK and public key PK as inputs and returns a message m or an error symbol $\perp$.

For consistency, it is required that whenever $c \leftarrow ENC(m, PK)$, $m \leftarrow DEC(c, SK, PK)$. Our protocol takes nonces as messages for the Encryption scheme. Both algorithms *ENC* and *DEC* are run by entities during authentication protocol.

The following Figure 5 depicts how a trusted vendor generates and distributes the Authentication Cards between its corresponding entities.

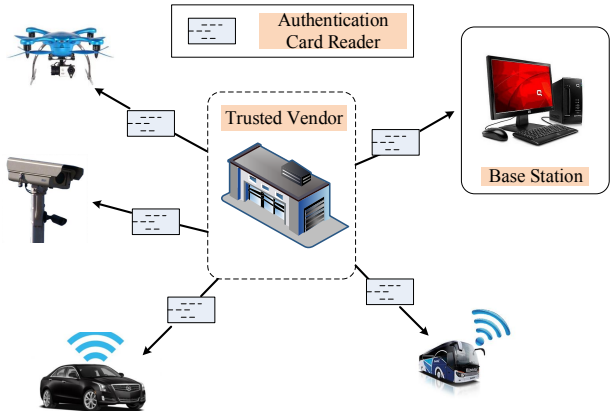

Figure 5: Authentication Cards of different entities generated by the trusted vendor.

B. Exemplary Deployment

Let us consider an example of some autonomous vehicle roaming around a road which gathers data from it's surrounding. The vehicle may be allowed only to collect data (image) and send it to ATMS or base station. It may not have enough computational capabilities to do any further processing on the collected data and therefore required only to transmit it to next level. In this scenario, a proper and secure authentication scheme is extremely important. Our solution provides a mutual authentication protocol as the authentication will be done between the vehicle and the base station. This protocol assumes that the both vehicle (inside its processing unit) and base station have their respective authentication cards. Consider the Figure 6.

Let the vehicle C and the base station B want to authenticate each other. Assume that the vehicle C starts the communication. In such case, C will send the signature of its public key and its public key along with its vendor number (everything stored inside the C's authentication card, please see Figure 4) to the base station B. After receiving these, B runs the verification algorithm at its end taking corresponding public key of the vendor and checks whether that received C's public key was indeed the genuine one or not. If the verification fails, B will not continue the communication. Otherwise, it will go for the next step. Note that, the authentication of public key does not guarantee that B with whom it was interacting was indeed C. Since entity's public key and its signature always flow unencrypted in the insecure channel, any attacker (malicious entity) can trap these and use them in the future for impersonating C. Therefore, an additional step is required for authentication. However, the authentication of the public key ensures that, with almost surety, it was generated by one of the trusted vendors, not by any attacker.

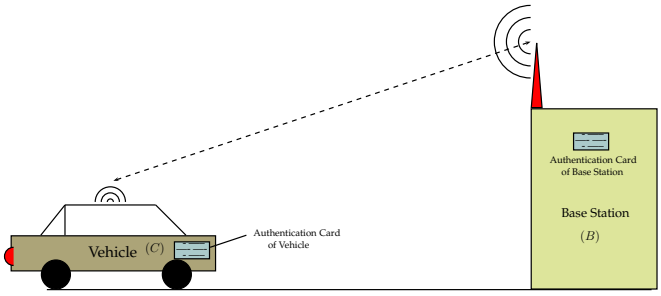

Figure 6: The vehicle and the base station with their respective authentication cards.

In a similar manner, the authentication of B's public key will be done by C also. Simultaneously, B will authenticate C by giving a challenge, say nonce_B, to C which will be encrypted using C's public key. C responds the challenge after sending the encrypted value of the same nonce added one (i.e. $\text{nonce}_B + 1$) to B. The response from C will be encrypted using B's public key. After receiving the response from C, B decrypts it and checks whether the sent nonce (nonce_B) and the received one ($\text{nonce}_B + 1$) differs by one or not? If so, B is confirmed that with whom it was interacting indeed has the secret key which has sent the public key in the first step. If not, B stops the communication. But, C has not verified yet that it is interacting with B only. For the authentication of B, C sends a challenge, nonce_C, along with its response to B. The challenge to B, nonce_C, will be encrypted using B's public key. After receiving the challenge, B decrypts it using its secret key and responds back to C by sending the encrypted value of nonce_C added one (nonce_C + 1) using C's public key. After receiving the response from B, C decrypts it using its secret key and checks whether the decrypted value is nonce_C added one ($\text{nonce}_C + 1$) or not? If not, C stops the communication otherwise authentication is completed from both sides.

C. Protocol

This section presents formal description of our proposed user authentication protocol. Let C and B participate in the

communication. The notation $C \rightarrow B : \{PK_C, \sigma_C, VN_C\}$ means C sends $\{PK_C, \sigma_C, VN_C\}$ to B. Here, PK_C denotes the public key of C, σ_C denotes the signature on PK_C generated using the secret key of the vendor number VN_C. For example, in the Figure 4, $VN_C = 2$ and therefore the verification of the public key-signature pair (PK_C, σ_C) will be done by the verification algorithm $VER(PK_C, \sigma_C, PK_{VN_C})$ using the public key of the second vendor. $ENC(N_B, PK_C)$ denotes the encryption of a randomly generated nonce N_B using the public key of C which is PK_C. Similarly, $DEC(C_4, SK_C, PK_C)$ denotes the decryption of C_4 using the secret key of C which is SK_C. In step 3, $\{C_1, C_2, C_3, C_4\} = \{PK_B, \sigma_B, VN_B, ENC(N_B, PK_C)\}$ denotes $C_1 = PK_B, C_2 = \sigma_B, C_3 = VN_B$ and $C_4 = ENC(N_B, PK_C)$.

Note that, in this protocol, two different public key schemes are used - one is Signature and another is Encryption. The *SIG* algorithm of signature scheme is used by vendor while generating the signature on public keys of vehicles and base stations (and other nodes/entities) whereas *VER* algorithm is run by the processing units of vehicles and base stations. Furthermore, *ENC* and *DEC* algorithms are run by vehicles and base stations only. Following is the stepwise explanation of the protocol proposed.

1) $C \rightarrow B : \{PK_C, \sigma_C, VN_C\}$
2) $B : VER(PK_C, \sigma_C, PK_{VN_C}) \stackrel{?}{=}$ true
   If the step above does not satisfy, B stops the communication, else goes to the next step.
3) $B \rightarrow C : \{C_1, C_2, C_3, C_4\} = \{PK_B, \sigma_B, VN_B, ENC(N_B, PK_C)\}$
4) $C : VER(PK_B, \sigma_B, PK_{VN_B}) \stackrel{?}{=}$ true
   If the step above does not satisfy, C stops the communication, else goes to the next step.
5) $C \rightarrow B : \{E_1, E_2\} = \{ENC(DEC(C_4, SK_C, PK_C) + 1, PK_B), ENC(N_C, PK_B)\}$
6) $B : DEC(E_1, SK_B, PK_B) \stackrel{?}{=} N_B + 1$
   If the step above does not satisfy, B stops the communication, else goes to the next step.
7) $B \rightarrow C : \{E_3\} = \{ENC(DEC(E_2, SK_B, PK_B) + 1, PK_C)\}$
8) $C : DEC(E_3, SK_C, PK_C) \stackrel{?}{=} N_C + 1$
   If the step above does not satisfy, C stops the communication, else authentication protocol over.

D. Key Agreement

Once the user authentication protocol is done successfully, a key agreement is required for the next step after which the vehicle and the base station start communicating with each other securely.

A key agreement can be done in many ways but this paper presents two different ways - (a) Any nonce which were generated during user authentication protocol may be used as a common key or (b) Physically Unclonable Function (PUF) may be used to generate common keys. For nonce based key exchange, although any nonce (either $N_B, N_B + 1, N_C, N_C + 1$) may serve the purpose of common key, we propose to use N_C as a common key.

E. Discussion

If the secret key of the entities get compromised, forward secrecy ensures the confidentiality of the data inside the secure session. Therefore, forward secrecy plays an important role in a secure communication and it is used in the key exchange phase. Although we don't discuss forward secrecy separately in this article, but one can obtain forward secrecy with replacing the nonces as powers of some primitive root modulo p group as given in the Diffie-Hellman Key-Exchange protocol [11].

After the key agreement, a secure session is established in which any authenticated encryption scheme can be used for further communication till this session ends. An authenticated encryption scheme provides both secrecy and message authentication at the same time. Message authentication ensures the communication entities that a tampered data can be easily detected. Authenticated Encryption (AE) consists of encryption and tag generation (hash) circuits. AE takes Plain-text (say images captured by a camera) and secret key as inputs and generates encrypted data with authentication tag (hash). The authentication tag used for checking the validity of received data. For example the ongoing worldwide competition called CAESAR aims to select best AE schemes which are being proposed by the researchers from all corners of the globe [13]. One can also use the candidates of that competition according to their requirements. For our test case, a lightweight CAESAR third round of candidate named ACORN cipher has been employed in this protocol for encryption and authentication. In ACORN , the difference is injected into the state during authentication for better performance. This type of authentication-encryption cipher is designed with the help of bit-based sequential stream cipher for the first time. High authentication security is achieved by employing six concatenated linear feedback shift registers. In addition, ACORN also allows parallel computation which benefits high-speed hardware and software implementation.

1) Assumptions on the Security of Our Proposed Protocol: We provide here some of the assumptions made in our model, which are

1) An attacker can collect any data (either encrypted or unencrypted) passing through any two entities while they communicate with each other.
2) An attacker can inject any data of his/her/its choice inside the data passing through any two entities while they communicate with each other.
3) An attacker may try to impersonate any entity a reasonable number of times.
4) An attacker may use previous data or his/her/its own data or a combination of both while impersonating any entity.

Our model declares an attacker as a winner when he/she/it finally succeeds in impersonating any genuine entity. If such

case happens, the authentication protocol is said to be insecure (otherwise secure) under our security model. There are limitations also in our security model which are

1) An attacker can not compromise the chip, circuit or any kind of data stored inside the entity.
2) An attacker can not do side-channel cryptanalysis.

Under the proposed security model, our user authentication protocol achieves the desired level of security when (a) the underlying public key signature is strongly unforgeable under chosen plaintext attack (SUF-CMA) and (b) the underlying public key encryption scheme is non-malleable under adaptive chosen ciphertext attack (NM-CCA2).

IV. Security Analysis

A detailed security analysis is avoided due to page limits, however, the real life attacks, such as Replay and Man-in-the-middle Attacks can successfully be safeguarded by our security model. Moreover, we have done theoretical analysis on adversarial models which can simulate such kind of real life attacks, and we also proved that under the given adversarial model, the proposed authentication and key agreement protocol is secure.

The modules suggested to depict the security model described in this paper are Physical Unclonable Function (PUF), our authentication protocol (consisting-trusted vendor, Signature Scheme and Public Key Encryption Scheme) and AEAD module for encryption and tag generation. **PUFs** could be a good option for generation of unique and random nonces (to ensure prevention against replay attacks), moreover it can also be used in randomization/key-generation for signature and encryption schemes. We suggest that **RSA-PSS** could be used to implement the signature scheme. Although it is not yet proved in literature if this scheme is SUF-CMA secure, but it still gives the desired level of security in authetication with signature. **RSA-OAEP** is proved to be NM-CCA2 secure, and we recommend that it can be used to implement the encryption scheme. In the secure session, a lightweight AEAD module, **ACORN-128**, [10] is developed which found to be more hardware efficient than TRIVIUM [8] and AES-GCM. The proposed model can be efficiently implemented with the help of modules discussed above in a cost effective manner without compromising the security.

V. Conclusion

This paper highlights the necessity of security when communication between the entities of a smart traffic infrastructures (such as intelligent vehicles, smart sensors/cameras, base stations etc.) plays an important role. A model for such secure infrastructure is proposed which consists of a user authentication and key-exchange protocol which can efficiently be applied in static as well as dynamic scenario. The proposed protocol is validated by the RSA-PSS algorithm for signature scheme and RSA-OAEP algorithm encryption scheme. A lightweight ACORN-128 AEAD cipher has been used in the symmetric phase providing confidentiality as well as authenticity of the data simultaneously in the secure session. Future works on this field is the performance evaluation of the proposed schemes on both software and hardware platform.

Acknowledgment

This research project is funded by the National Research Foundation Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) program.

References

[1] "Hacking traffic lights amazingly," https://thehackernews.com/2014/08/hacking-traffic-lights-is-amazingly_20.html.
[2] "Kerberos papers and documentation," http://web.mit.edu/kerberos/papers.html.
[3] "Researchers hack into michigan's traffic lights," https://www.technologyreview.com/s/530216/researchers-hack-into-michigans-traffic-lights.
[4] "U.s. department of transportation releases policy on automated vehicle development," http://www.nhtsa.gov/About+NHTSA/Press+Releases/U.S.+Department+of+Transportation+Releases+Policy+on+Automated+Vehicle+Development., April 2016.
[5] M. Bilal and S.-G. Kang, "An authentication protocol for future sensor networks," *Sensors*, vol. 17, no. 5, 2017. [Online]. Available: http://www.mdpi.com/1424-8220/17/5/979
[6] A. Burg, A. Chattopadhyay, and K. Y. Lam, "Wireless communication and security issues for cyber-physical systems and the internet-of-things," *Proceedings of the IEEE*, vol. 106, no. 1, pp. 38–60, Jan 2018.
[7] T. Bécsi, S. Aradi, and P. Gáspár, "Security issues and vulnerabilities in connected car systems," in *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, June 2015, pp. 477–482.
[8] C. D. Canniere and B. Preneel, "Trivium: In new stream cipher designs the estream finalists," in *Springer verlag*. Springer, 2008.
[9] A. Chattopadhyay and K. Y. Lam, "Security of autonomous vehicle as a cyber-physical system," in *2017 7th International Symposium on Embedded Computing and System Design (ISED)*, Dec 2017, pp. 1–6.
[10] A. Chattopadhyay, V. Pudi, A. Baksi, and T. Srikanthan, "Fpga based cyber security protocol for automated traffic monitoring systems: Proposal and implementation," in *2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, July 2016, pp. 18–23.
[11] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, Nov 1976.
[12] R. Fan, L.-D. Ping, J.-Q. Fu, and X.-Z. Pan, "A secure and efficient user authentication protocol for two-tiered wireless sensor networks," in *2010 Second Pacific-Asia Conference on Circuits, Communications and System*, vol. 1, Aug 2010, pp. 425–428.
[13] George Mason University, "ATHENa: Automated Tools for Hardware EvaluatioN," https://cryptography.gmu.edu/athena/, 2017.
[14] B. Ghena, W. Beyer, A. Hillaker, J. Pevarnek, and J. A. Halderman, "Green lights forever: Analyzing the security of traffic infrastructure," in *8th USENIX Workshop on Offensive Technologies (WOOT 14)*. San Diego, CA: USENIX Association, 2014. [Online]. Available: https://www.usenix.org/conference/woot14/workshop-program/presentation/ghena
[15] S. Halevi and H. Krawczyk, "Public-key cryptography and password protocols," *ACM Trans. Inf. Syst. Secur.*, vol. 2, no. 3, pp. 230–268, Aug. 1999. [Online]. Available: http://doi.acm.org/10.1145/322510.322514
[16] I. Park, S. Park, and B. Oh, "User authentication protocol based on human memorable password and using rsa," in *Computational Science and Its Applications – ICCSA 2004*, A. Laganá, M. L. Gavrilova, V. Kumar, Y. Mun, C. J. K. Tan, and O. Gervasi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 527–536.
[17] Z. Quan, T. Chunming, Z. Xianghan, and R. Chunming, "A secure user authentication protocol for sensor network in data capturing," *Journal of Cloud Computing*, vol. 4, no. 1, p. 6, Apr 2015. [Online]. Available: https://doi.org/10.1186/s13677-015-0030-z