Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 42 (2014) 263 – 270

Procedia Computer Science

International Conference on Robot PRIDE 2013-2014 - Medical and Rehabilitation Robotics and Instrumentation, ConfPRIDE 2013-2014

Cryptographic Key Exchange Protocol with Message Authentication Codes (MAC) using Finite State Machine

Mohd Anuar Mat Isa[a]*, Miza Mumtaz Ahmad[b], Nor Fazlida Mohd Sani[b], Habibah Hashim[a], Ramlan Mahmod[b],

[a]*Faculty of Electrical Engineering, 40450 UiTM Shah Alam, Selangor, Malaysia.*
[c]*Faculty of Computer Science & Information Technology, 43400 UPM Serdang, Selangor, Malaysia.*

Abstract

In this work, we explore the authentication and verification of key exchange protocol using Message Authentication Code (MAC). We propose a new MAC scheme model using input-output automata to protect the integrity of the secret key in the key exchange protocol. Our scheme was devised in reference to the Diffie-Hellman communication protocol model. We divided our MAC protocol into three stages of communication sequences in order to simplify the model and the design of automata machine. In the final result, we combined all stages and represented the protocol as Cryptographic MAC Protocol in the regular language. We have shown that the cryptographic MAC protocol for key exchange protocol can be implemented using finite input-output automata with some small modification of the finite state machine. The proposed protocol would be useful for implementation in a lightweight or a secure smart devices communication in the wireless sensor nodes (WSN) network.

Peer-review under responsibility of the Center for Humanoid Robots and Bio-Sensing (HuRoBs)

Keywords: cryptography; security protocol; message authentication codes; input-output automata; cellular automata; timed automata; Diffie-Hellman, MAC, Security, Trust, Privacy, STP, formal methods, finite state machine, lightweight, asymmetric, symmetric, encryption, hashing;

* Corresponding author.
*E-mail address:*anuarls@hotmail.com

Peer-review under responsibility of the Center for Humanoid Robots and Bio-Sensing (HuRoBs)
doi:10.1016/j.procs.2014.11.061

1. Introduction

Lightweight security applications that are offers high security services such as authentication and confidentiality to contemporary network technologies such as "Internet of Things" and "Cloud Computing" are expected to facilitate and accelerate the growth and popularity of these technological platforms. In these environments, authentication plays a major role in securing the communication for the purpose of ensuring the legitimacy of the transmitted data as well as the communicating parties [1]. In practice, a message authentication codes or MAC is sent together with the message via a secure channel. The MAC is derived from a hash function of the message with the secret key as its variable. Since the sender and the receiver have the same secret key, both parties will be able to recompute the MAC using their shared variables. Automata Theory is one of the common ways to engineer the MAC. There are several types of automaton that are suitable for security purposes such as cellular, input-output, asynchronous product, tree and timed automata. In this work, we focus on modeling our MAC scheme using the input-output variant due to its practicality to our model.

Diffie-Hellman Key Exchange (DHKE) protocol is use to exchange a cryptographic key between two parties that have no prior knowledge of each other and to establish a key exchange between them over insecure communication channels [2]–[4]. The Diffie-Hellman algorithm was developed by Whitfield Diffie and Martin Hellman in 1976 as a major breakthrough in the asymmetric key encryption. It serves the purposes to generate a same private cryptographic key at both end-points (e.g. client and server) so that there is no need to transfer this key from one communication end to another end in physical manner. The primitive mathematical strength of this algorithm that is relies on the Discreet Log Hard Problem (DLHP) wherein to find private exponential number is intractable [4]. This algorithm is proven secure with proper usage and combined with Message Authentication Code (MAC) [3]. A commendable security property from this algorithm is to provide secure key exchange in the TFTP communication. Therefore, the product or key generated from this algorithm can be used for AES128 and AES256 in the data encryption.

In the next section, we present the literature review of previous works done on security protocol using some of the most common automata. Section 3 briefly describes our motivation and objective. We present our model in Section 4 followed by the security analysis in Section 5. The result and discussion are discussed in Section 6 and 7. Section 8 concludes the research work.

2. Related Work

There are several ways to use automata engineering security protocol as well as in checking the protocol model. Wolfram (1986) [5] is the founder of cellular automata-based cryptographic scheme for symmetric encryption protocol. The proposed protocol is implemented in stream cipher using one-dimensional N cells of cellular automata. Timed automaton as proposed by Alur et al. (1994) [6] is a methodology to model a system in real time as well as Kurkowski et al. (2009) [7] for verifying timed security protocols in a computer network among finite users with the presence of an intruder. Cellular automata are also used for the message and image authentication scheme as proposed by Mukherjee, Ganguly et al. (2002) [8]. The scheme is said to be highly secure against current security threats and highly efficient for real-time system implementation as well as VLSI production with high authentication throughput. Needham-Schroeder protocol is described using asynchronous product automata (APA) (Gürgens et. al 2002) [9]. Needham-Schroeder protocol is a key distribution protocol from a key server S to agents A and B, making all S, A and B as elementary automata with 4 states each in APA.

Bao (2004) [10] states that most of the previous cellular automata cryptosystems are vulnerable to cryptanalysis like chosen-plaintext attack. Only hundreds of chosen plaintexts are required to break the cipher text from the cryptosystem as compared to the minimal security benchmark of 2^{80} chosen plaintexts. Furthermore, the attacker does not need to know the cryptosystem design in order to break the code. Corin, et al. (2004) [11] proposed security protocol checking based on timing information of the protocol. To consider the time factor in a security protocol, timed automata is used in engineering the security model. One benefit of using timed automata in the protocol checking is the protocol performance becomes highly translucent with the accurate and thorough protocol specification.

Finite automata is used to analyze complexity of Dolev–Yao [12] model of security protocols using a notation based on multi-set rewriting with existentials (Durgin, et al. 2004) [13]. Blundo, et al. (2004) [14] experimented with input-output automata as a tool to describe and verify the security of cryptographic protocols that run in an asynchronous distributed system. Using this security verification procedure, some security properties of a certified email protocol

can be checked after it has been examined and formalized using IOA model. We choose this automata model due to its feasibility and ability to permit precise description of the code with detailed proofs via its framework. Similar work on verifying and analyzing security protocols have been done on a shared key communication protocol and Diffie-Hellman key distribution protocol as presented in (Lynch 1999) [15].

Rey (2007) [16] presented a secure message communication using finite cellular automata as hash function to prevent same hash digest for different messages or packets. Hash function is used to map memory cellular automata with k-length hash digest as message integrity. Cellular automaton in MAC utilizes transition function to provide a chain of message integrity as hash digest. Reitzig (2008) [17] proposed utilizing finite automata for modelling and proving system security using formalism of system properties. Kurkowski et al. (2009) [7] used timed automata for verifying timed security protocols in a computer network among finite users with the presence of an intruder. Similar work was also done by Koltuksuz et al. (2010) [18] to verify Neuman-Stubblebine Repeated Authentication Protocol by building a finite model of the system and navigating through all of the accessible states. Liu et al. (2009) [19] defined security protocol using approximation-based model and came up with an algorithm design that is analogous to the real implementation. This design is used to calculate fix-point tree automata based on the tool called TA4SP. Furthermore, the authors proposed hierarchy analysis on authentication properties as extension to TA4SP. Comparably, Gennaro et al. (2010) [20] applied tree automata in analysis of cryptographic protocols based on abstract interpretation and regular tree languages. Oliveira et al. (2010) [21] implemented cellular automata in Variable-Length Encryption Method (VLE) for symmetric encryption scheme. The VLE produces shorter cipher text with high level of randomness in the text known as encryption entropy. This is to strengthen the scheme against differential cryptanalysis via transition rules in the automata and spaces as specifications.

In our previous work, we have conducted an experiment using RaspberryPi ARM [22] board to enable secure TFTP communication between client and server. The secure key exchange is derived from cryptographic algorithms such as DHKE [2], [4] Cramer-Shoup [23] and El-Gamal [24] encryption schemes. For an extensive implementation details of our secure protocol with a security proofs, one may refer to our latest works (2014) in the research papers “An Experimental study of Cryptography Capability using Chained Key Exchange Scheme for Embedded Devices” [25], “A Secure TFTP Protocol with Security Proofs” [26] and “Performance Measurement of Secure TFTP Protocol for Smart Embedded Devices” [27].

3. Motivation

Our main motivation in proposing the Message Authentication Code (MAC) scheme using input-output automata is to protect the integrity of the key in the key exchange protocol. To explore the possible constraints in the theoretical and formal designs, we have decided to use hashing algorithm such as SHA1 that provides 160-bit of integrity value. We consider the communication between two parties with the presence of an eavesdropper as the adversary. The main purpose of the above scenario is to show that automata can be used to model MAC as well as to verify its security using outsider adversary threat model. We assume that the adversary tries to forge the MAC and the secret key in the communication channel, but it will fail. The objective of this paper is to explore Message Authentication Code (MAC) scheme using finite input-output automata for message integrity verification in the unsecure communication channel.

4. Proposed MAC Automaton

4.1. MAC Protocol & Key Exchange Protocol

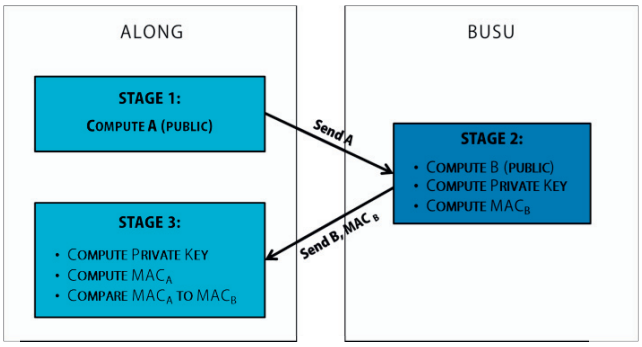

Fig. 1: General protocol for MAC in the Key Exchange Protocol

4.2. Definition and Formalism

- $a, b, g, p, A, B, K \in \{0,1\}^*$: *size for every elements is* 2048-bit (1)
- H : *Hash algorithm* (2)
- T : *Transmission over network* (3)
- "^" : *Exponential symbol* (4)
- "=" : *Compare symbol* (5)
- "M" : *Modular symbol* (6)
- $R, S \in \{0,1\}^*$, $|R| =$ 160-bit &$|S| =$ 160-bit (7)
- *Language*, $\boldsymbol{L(MAC)} = (\boldsymbol{Q}, \boldsymbol{\Sigma}, \boldsymbol{\delta}, \boldsymbol{q_o}, \boldsymbol{F})$: (8)
  - Q: *All finite states in the stage* 1 *to* 3
  - Σ : *All alphabets from* (1) *to* (7)
  - δ : $Q \times \Sigma$, δ: *transition function*
  - $q_0 \in Q$, q_0: *initial state in the stage* 1 *to* 3
  - $F \subseteq Q$, F: *set of acccepting states in the stage* 1 *to* 3

4.3. Cryptographic MAC Protocol in Key Exchange Protocol.

Stage 1:

$$A \equiv g^a \ (mod\ p) \quad (9)$$

Stage 2:

$$B \equiv g^b \ (mod\ p) \quad (10)$$
$$Private\ Key,\ K \equiv A^B \ (mod\ p) \quad (11)$$
$$MAC_B \equiv Hash(K \mid A \mid B) \quad (12)$$

Stage 3:

$$Private\ Key,\ K \equiv B^A \ (mod\ p) \quad (13)$$
$$MAC_A \equiv Hash(K \mid A \mid B) \quad (14)$$
$$Compare\ MAC_A = MAC_B \quad (15)$$

4.4. Automata Machine

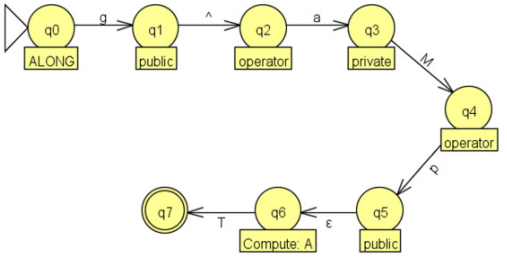

Fig. 2. Nondeterministic Finite Automaton (NFA) in **Stage 1**

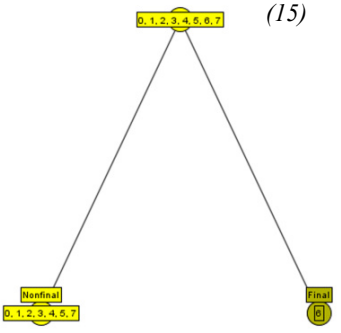

Fig. 3. Summary states in the **Stage 1**

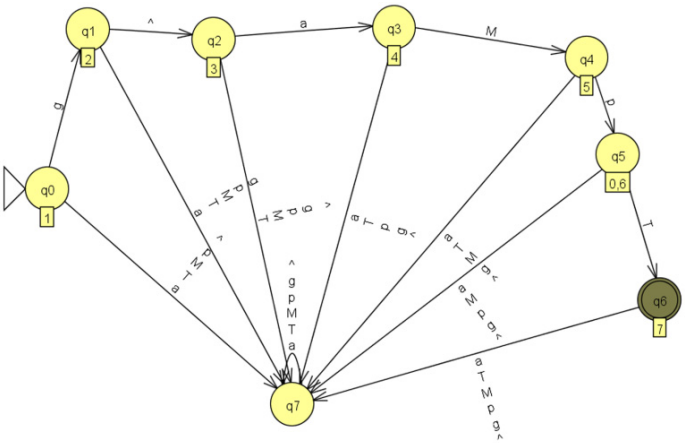

Fig. 4. Deterministic Finite Automaton (DFA) in the **Stage 1**

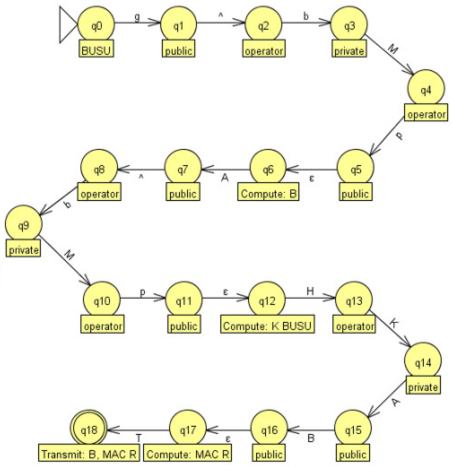

Fig. 5. Nondeterministic Finite Automaton (NFA) in **Stage 2**

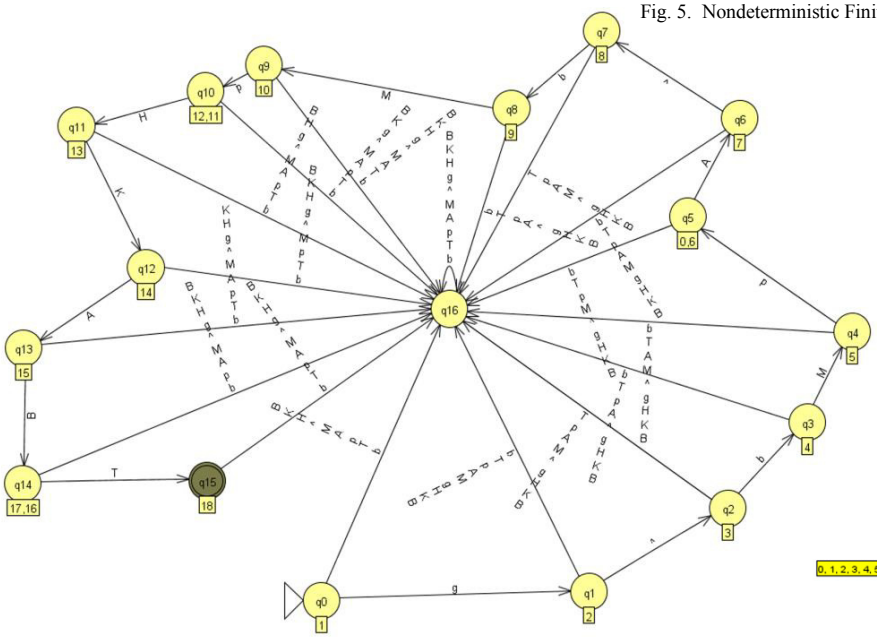

Fig. 6. Deterministic Finite Automaton (DFA) in the **Stage 2**

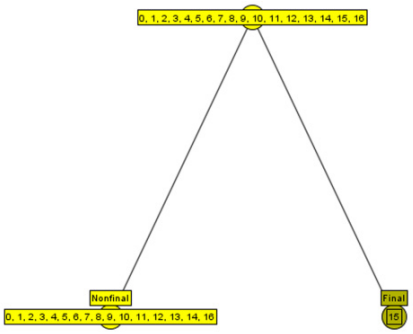

Fig. 7. Summary states in the **Stage 2**

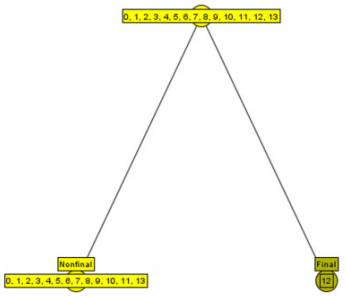

Fig. 8. Summary states in the **Stage 3**

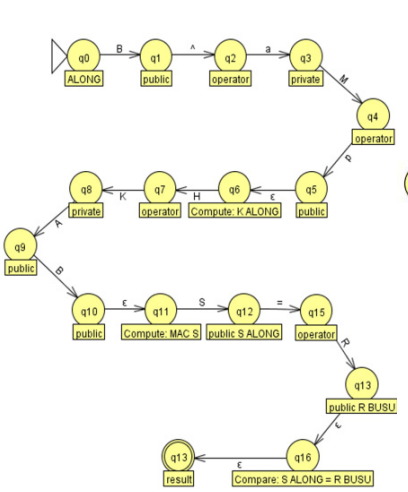

Fig. 9. Nondeterministic Finite Automaton (NFA) in **Stage 3**

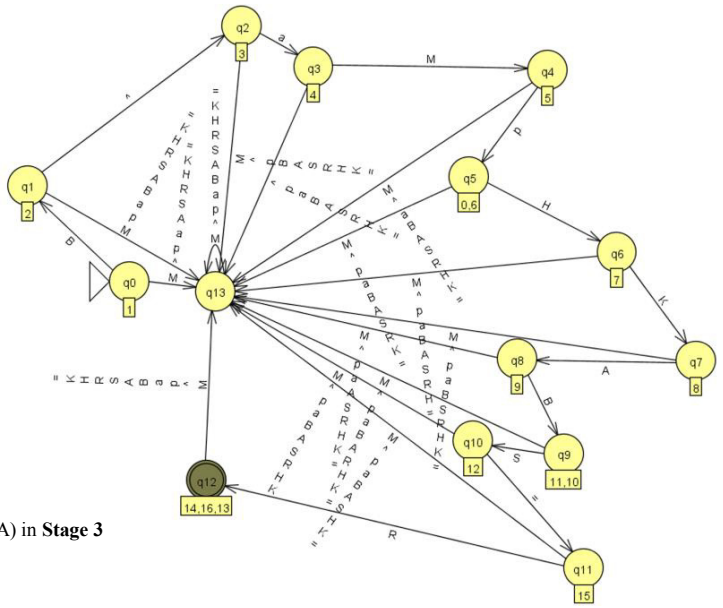

Fig. 10. Deterministic Finite Automaton (DFA) in the **Stage 3**

5. Analysis of Automaton

5.1. Language and Regular Expression for Stage 1

$L(Stage1) = \{ x \in \Sigma^* : \textit{Accept only strings } x \textit{ with } \boldsymbol{g\text{^}aMpT} \text{ pattern} \}$

$(\boldsymbol{g\text{^}aMpT})^*$

5.2. Language and Regular Expression for Stage 2

$L(Stage2) = \{ x \in \Sigma^* : \textit{Accept only strings } x \textit{ with } \boldsymbol{g\text{^}bMpA\text{^}bMpHKABT} \text{ pattern} \}$

$(\boldsymbol{g\text{^}bMpA\text{^}bMpHKABT})^*$

5.3. Language and Regular Expression for Stage 3

$L(Stage2) = \{ x \in \Sigma^* : \textit{Accept only strings } x \textit{ with } \boldsymbol{B\text{^}aMpHKABS = R} \text{ pattern} \}$

$(\boldsymbol{B\text{^}aMpHKABS = R})^*$

6. Results

6.1. Language in MAC Protocol for Key Exchange Protocol

$\boldsymbol{L(MAC)}$
$= \{ \boldsymbol{x} \in \boldsymbol{\Sigma}^* : \textit{Accept only strings } x \textit{ with } (\boldsymbol{g\text{^}aMpT})^* (\boldsymbol{g\text{^}bMpA\text{^}bMpHKABT})^* (\boldsymbol{B\text{^}aMpHKABS = R})^* \text{ pattern} \}$

$$((g\hat{}aMpT)^* \ (g\hat{}bMpA\hat{}bMpHKABT)^* \ (B\hat{}aMpHKABS = R)^*)^*$$

7. Discussion

In section 4, we have shown that the Cryptographic MAC Protocol for Key Exchange Protocol can be implemented using finite input-output automata with slight modification. We used JFLAP Thin 7.0 to model and test our automaton machine. As illustrated in Figure 1 to Figure 10, we have divided our MAC protocol into three stages of communication sequences in order to simplify the model and the design of automata machine. We latter combined all stages as shown in section 6 as the final product of cryptographic MAC protocol in the regular language.

The idea to utilize cellular automata in security protocol was proposed in 1986 by Wolfram [28]. To date, this type of security protocol is only practical for implementation in symmetric encryption scheme. The scheme based on cellular automata proposed by Mukherjee et al [8] is suitable for message and image authentication. In 2007, MAC system that used cellular automata for its hashing algorithm is proposed Rey [16]. After that, Oliveira [21] implemented cellular automata in Variable-Length Encryption Method for bulk data encryption such as Secure File Transfer Protocol.

The aforementioned schemes seem to be incompatible with our idea. Since our proposed protocol does not require the property of randomness and hashing algorithm derived from cellular automata. Nevertheless, the chosen-plaintext attack on cellular automata protocol discussed by Bao [10] provides some guidelines for our to design the key exchange protocol and MAC models. The work done by Oliveira [21] is the most comparable to our proposed model. Though there is a discussion on client-server security protocol that relates to key exchange. However, the Oliveira's model was designed for security policy only, instead of cryptographic key exchange protocol and MAC. We compare our implementation to Lynch [15] that is uses automaton to model, analyze and verify key exchange protocol such as Diffie-Hellman Key Exchange (DHKE) protocol. Our implementation also support outsider adversary threat model for security analysis and security proving. This research work is the complement of our previous work [25]–[27], [29] in providing secure I/O communication in the embedded systems such as sensor nodes security.

8. Conclusion

Since 1986, finite automata have been used widely for modeling and designing security model as well as constructing pseudo random functions. In later decades, automata are used for security implementation in network security protocol (e.g. client-server system). The use of finite automata in modeling security protocol enables formal proving on its security. We used Lynch's model [15] as reference model for designing and modeling our very own MAC scheme based on adversary model, Diffie-Hellman communication protocol model and environment models. We have modeled our new MAC protocol for key exchange protocol using input-output automata due to its practicality. For future work, we plan to check our model using timed automata and formal methods to verify the proposed MAC protocol in term of several properties that are including liveness, safety, deadlock-free and starvation-free.

Acknowledgements

The authors would like to acknowledge the Ministry of Education (MOE) Malaysia for providing the grant 600-RMI/NRGS 5/3 (5/2013), and Universiti Teknologi MARA (UiTM) for supporting this research work.

References

[1] S. Hankerson, Darrel, Menezes, Alfred J., Vanstone, Guide to Elliptic Curve Cryptography. 2004.
[2] E. Rescorla, "Diffie-Hellman Key Agreement Method (RFC 2631)," in The Internet Society, 1999.
[3] E. Yoon and K. Yoo, "An Efficient Diffie-Hellman-MAC Key Exchange Scheme." pp. 398–400, 2009.
[4] W. Diffie and M. E. Hellman, "New Directions in Cryptography," in IEEE Transactions on Information Theory, 1976.
[5] S. Wolfram, "Cryptography With Cellular Automata," in Advances in Cryptology: Crypto'85 Proceedings, Lecture Notes in Computer Science, 1986.
[6] R. Alur and D. Dill, "A theory of timed automata," in Theoretical Computer Science, 1994.
[7] M. Kurkowski and W. Penczek, "Timed automata based model checking of timed security protocols," in Fundamenta Informaticae, 2009.

[8] M. Mukherjee, N. Ganguly, and P. Chaudhuri, "Cellular automata based authentication (CAA)," Cell. Autom. Lect. Notes Comput. Sci., vol. 2493, pp. 259–269, 2002.
[9] S. Gürgens, P. Ochsenschläger, and C. Rudolph, "Role based specification and security analysis of cryptographic protocols using asynchronous product automata," in 13th International Workshop on Database and Expert Systems Applications, 2002.
[10] F. Bao, "Cryptanalysis of a partially known cellular automata cryptosystem," IEEE Trans. Comput., vol. 53, no. 11, pp. 1493–1497, Nov. 2004.
[11] S. E. Corin, R., P. H. Hartel, and A. Mader, "Timed model checking of security protocols," in ACM Workshop on Formal methods in Security Engineering, 2004.
[12] D. Dolev and A. C. Yao, "On the security of public key protocols," IEEE Trans. Inf. Theory, vol. 29, no. 2, pp. 198–208, Mar. 1983.
[13] A. S. Durgin, N., P. Lincoln, Mitchell J., "Multiset rewriting and the complexity of bounded security protocols," Journal of Computer Security, vol. 12, 2004.
[14] A. L. F. Blundo, C., S. Cimato, R De Prisco, "Modeling a certified email protocol using I/O automata," in Electronic Notes in Theoretical Computer Science, 2004.
[15] N. Lynch, "I/O automaton models and proofs for shared-key communication systems," in 12th IEEE Computer Security Foundations Workshop, 1999.
[16] A. del Rey, "Message authentication protocol based on cellular automata," in Applications of Evolutionary Computing, 2007.
[17] R. Reitzig, "Modelling and Proving System Security Using Finite Automata and Noninterference," in Seminar "Software Engineering" TU Kaiserslautern, 2008.
[18] M. O. Koltuksuz, A., B. Kulahcioglu, "Utilization of timed automata as a verification tool for security protocols," in Fourth IEEE International Conference on Secure Software Integration and Reliability Improvement Companion, 2010.
[19] W.-Y. Z. Liu, N. and Z. Y.-F, "Security protocol analysis based on rewriting approximation," in Second International Symposium on Electronic Commerce and Security, 2009.
[20] C. H. Gennaro, R. and J. S. Sorensen, "IACR Cryptology ePrint Archive," IACR Cryptology ePrint Archive. [Online]. Available: eprint.iacr.org/2010/484.pdf? [Accessed: 15-Sep-2013].
[21] G. Oliveira and L. Martins, "Secret key specification for a variable-length cryptographic cellular automata model," Parallel Probl. Solving from Nature, PPSN XI, Lect. Notes Comput. Sci., vol. 6239, pp. 381–390, 2010.
[22] Raspberry Pi Foundation, "Raspberry Pi," 2014. [Online]. Available: http://www.raspberrypi.org/downloads.
[23] R. Cramer and V. Shoup, "A Practical Public Key Cryptosystem Provably Secure Against Adaptive Chosen Ciphertext Attack," in Lecture Notes in Computer Science: Advances in Cryptology—CRYPTO'98, 1998.
[24] T. Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," IEEE Trans. Inf. Theory, vol. 31, no. 4, pp. 469–472, Jul. 1985.
[25] Mohd Anuar Mat Isa, Habibah Hashim, Syed Farid Syed Adnan, Jamalul-lail Ab Manan, and Ramlan Mahmod, "An Experimental study of Cryptography Capability using Chained Key Exchange Scheme for Embedded Devices," in Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering 2014, (WCE 2014), 2014.
[26] Mohd Anuar Mat Isa, Habibah Hashim, Syed Farid Syed Adnan, Jamalul-lail Ab Manan, and Ramlan Mahmod, "A Secure TFTP Protocol with Security Proofs," in Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering 2014, (WCE 2014), 2014.
[27] Nik Fifi Sofia Pauzi, Mohd Anuar Mat Isa, Habibah Hashim, Syed Farid Syed Adnan, and Lucyantie Mazalan, "Performance Measurement of Secure TFTP Protocol for Smart Embedded Devices," in IEEE Asia Pasific Conference on Wireless and Mobile, 2014.
[28] S. Wolfram, "Cryptography with cellular automata," in Advances in Cryptology: Crypto'85. Lecture Notes in Computer Science, 1986.
[29] Mohd Anuar Mat Isa, Nur Nabila Mohamed, Habibah Hashim and R. M. Syed Farid Syed Adnan, Jamalul-lail Ab Manan, "A Lightweight and Secure TFTP Protocol in the Embedded System," in 2012 IEEE Symposium on Computer Applications and Industrial Electronics (ISCAIE 2012), 2012.