ISBN 978-981-11-3671-9
Proceedings of 2017 the 7th International Workshop on Computer Science and Engineering (WCSE 2017)
Beijing, *25-27 June*, *2017*, *pp. 968-976*
doi: 10.18178/wcse.2017.06.168

Performance Analysis of Certificate Renewal Scheme for AMI Networks

Ahmad Alsharif [1], Samet Tonyali [2], Mohamed Mahmoud [1], Kemal Akkaya [2], Muhammad Ismail [3], Erchin Serpedin [3+]

[1] Department of Electrical and Computer Engineering, Tennessee Tech University, Cookeville, TN, USA
[2] Department of Electrical and Computer Engineering, Florida International University, Miami, FL, USA.
[3] Department of Electrical and Computer Engineering, Texas A&M University at Qatar, Doha, Qatar.

Abstract. Public-key cryptography is indispensable for securing the communications in Advanced Metering Infrastructure (AMI) networks. However, few works have studied the efficient use of public key cryptography certificates in such a network and most of them focus on certificates' revocation. In this paper, we extensively investigate the performance our previous proposal on an efficient certificate renewal scheme that we proposed for AMI networks. First, quantitative analysis is carried out to compare our scheme against signature-based certificate renewal schemes. Then, all schemes are implemented in a realistic network model using NS-3 to evaluate their performance. Simulation results demonstrate the improved performance of our scheme in computational cost, communication overhead, end-to-end delay, packet delivery ratio, and required bandwidth compared with the signature-based certificate renewal scheme.

Keywords: Public key management system, certificate renewal, smart grid communication security, authentication, authorization, and access control.

1. Introduction

The Smart Grid (SG) initiative aims to enhance the reliability and efficiency of the traditional power grid by utilizing two-way communications between its major components. A main component in the SG is the Advanced Metering Infrastructure (AMI) networks that allow two-way communications between Smart Meters (SMs) installed at the customers' side and the utility. AMI networks are used to provide the utility with the consumers' fine-grained power consumptions (every few seconds) for monitoring, management, and state estimation. They also provide consumers with real-time electricity prices to enable demand/response applications to reduce the power consumption at peak hours. Communications' security should be guaranteed before the wide deployment of the AMI networks.

Public Key Cryptography (PKC) is the most practical and common cryptosystem that can be used to secure the AMI communications by achieving the main security requirements such as message authentication and integrity, non-repudiation, accountability, and access control [1]–[3]. In PKC, a pair of public/private keys is issued for each SM and the announcement of the public key is usually done by a trusted Certificate Authority (CA) whose public key is known to all SMs in the network. This announcement is done through a public key certificate signed by the CA that binds the certificate holder's identity to its public key. To authenticate a message, a SM signs the message with its private key and a digital signature algorithm. The verifier of the message's signature first validates the authenticity of the public key of the signer by verifying the certificate. The purpose of this verification is to ensure that the message's sender is a legitimate member

+ Corresponding author.
E-mail address: eserpedin@qatar.tamu.edu.

in the CA's domain. Then, it verifies the signature using the signer's public key (extracted from the certificate) and a signature verification algorithm.

When a certificate is issued, its lifetime is limited by an expiration date. After this date, the certificate is considered expired and will not be accepted by the SMs. Certificates should be periodically updated before they are expired. However, there are several motivations that necessitate revoking the certificates before their expiry date [4]. Examples of these motivations include key compromise, malicious behaviour of a certificate holder, broken signature scheme, etc.

The public key management system governs the distribution and management of the cryptographic keys and certificates to enable the use of the PKC. The main tasks performed by the system are key generation, certificate issuance, certificate renewal, certificate revocation, and key backup/recovery. A good public key management system for the AMI networks should consider the unique requirements and characteristics of such networks. These characteristics include scalability, large geographical spread, and immobile, low-resource, and unattended nodes. Nevertheless, very few works have investigated the efficient use of PKC in SG AMI networks. While the works in [5]–[9] focused on efficient certificate revocation in SG, our proposed scheme in [10] focused on efficient certificate renewal scheme. Reducing the overhead of certificate renewals in terms of generation, distribution and verification can also expedite message authentication and enhance certificate revocation by shortening the certificates' lifetime with acceptable overhead. However, very limited performance evaluations are reported in [10].

In this paper, we extensively investigate the performance of our proposal in [10]. We use different public key cryptosystems including 2048-bit RSA [11] and 256-bit ECDSA [12]. We also use the Crypto++ library [13] to implement these cryptosystems and measure the computation times. Finally, the certificate renewal scheme has been implemented in realistic settings using NS-3 network simulator [14] in an AMI network to evaluate the network performance under different settings such as the number of SMs in the network. The evaluation results have demonstrated that our certificate renewal scheme requires much less overhead than the traditional approaches. The simulation results demonstrate a significant advantage in terms of delay and required bandwidth when our scheme is adopted. This reduction in the amount of traffic in the network is important because the AMI networks can be used in many applications other than sending meters' readings.

The remainder of this paper is organized as follows. Section 2 presents our certificate renewal and expedited message authentication scheme. Evaluations are given in Section 3. The related works are discussed in Section 4, followed by conclusion and future work in Section 5.

2. Certificate Renewal and Expedited Authentication Scheme For AMI Networks

Certificates are valid only for a limited period of time. Therefore, the CA has to renew them before they expire. If a SM does not obtain a renewed certificate before the expiry date, it will no longer be able to communicate with other SMs. Issuing certificates with permanent or long lifetime, e.g., 20 years, looks attractive because of the low overhead in renewing them. However, it will have the following serious implications:

1. Certificate rekeying: Regularly changing the private/public key pair is a desirable security practice to make cryptanalysis infeasible. The risk of key compromise increases when it is used for a long time. The SMs can change the key pair during certificate renewals.
2. Inefficient certificate revocation: Revoked certificates' serial numbers must stay on a certificate revocation list (CRL) until the certificates expire. With permanent certificates, the certificates' serial numbers will be kept in the CRL for the certificate lifetime. Over time, the CRL will dramatically grow as more and more certificates are added to the list. Large CRLs will consume much resources in disseminating and storing them. To prevent the CRL from growing too large, the CA should determine an appropriate lifetime for the certificates.

Generally speaking, shorter-lifetime certificates are more desirable from security perspective because the malicious nodes can be revoked faster after their certificates expire. However, much communication and computational overhead is required for computing the renewed certificates' signatures and also for

distributing them. The AMI network scalability will obviously worsen this overhead. In order to improve the scalability of certificate renewal process, we have proposed an efficient certificate renewal scheme for AMI

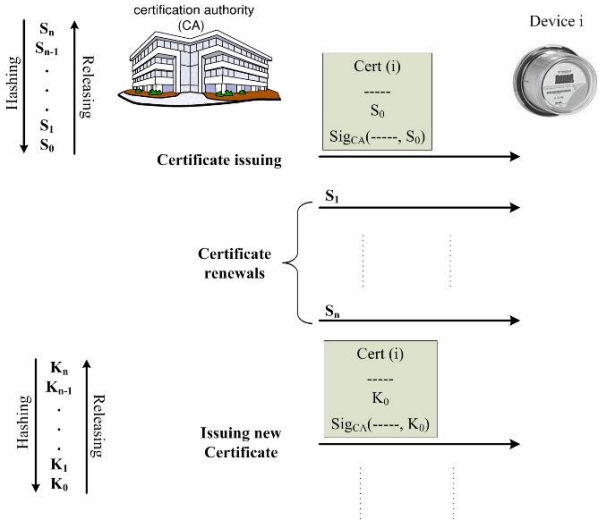

Fig. 1: The proposed certificate renewal scheme in [10].

networks in [10]. The basic idea of the scheme is that for each certificate, the CA creates a one-way hash chain by iteratively hashing a random secret seed (S_n) n times to obtain the root hash value S_0. $S_{i-1} = H(S_i)$, $H(S_i)$ is the hash value resulted from hashing S_i for $1 \leq i \leq n$. The root hash value is included in the certificate, as indicated in Fig. 1. A certificate's lifetime (T) is divided into n shorter time slots, where each time slot $\tau = T/n$. Each hash chain element S_i is used to renew the certificate for one time slot. The node needs a fresh hash chain element instead of a new signature to renew a certificate, as indicated in Fig. 1.

Fig. 1 shows that after time period T, a new certificate with a new hash chain, signature, and probably new public/private keys should be issued. For certificate verification, the nodes need to verify the certificate's signature only one time when they verify it for the first time. Then, they need to perform one efficient hashing operation to verify the certificate when its last hash value changes to ensure that S_{i-1} is obtained from hashing S_i.

Moreover, in some cases, the CA needs to temporarily suspend a SM's certificate for a certain time period, e.g., from t_x to t_{x+b} . For example, some devices will be temporarily out of service when there is a maintenance or extension in the power system. In our scheme, certificate suspension can be done efficiently by not releasing the relevant hash chain elements, so that the certificate will not be accepted by the nodes. This certificate suspension technique is much more efficient than the traditional approach that requires first revoking the certificate at t_x and then issuing a new one at t_{x+b}. To authenticate a message, the verifier has to verify the signer's signature and certificate. Our scheme can expedite message authentication by reducing the certificate verification time because it can replace the certificate's signature verification with a fast hashing operation. When verifying a certificate for the first time, the verifier should verify the certificate's signature. Then, it verifies the hash chain elements by repeatedly hashing the most recent hash value S_i until it obtains the root hash value (S_0) listed in the certificate. The certificate is invalid if S_0 cannot be obtained from hashing S_i . The verifier calculates the certificate's expiry date as follows: $\text{expiry date} = \text{issuance date} + \tau \times (\text{number of hashing operations} + 1)$. The certificate is valid if the current date is less than the expiry date.

At the beginning of each time slot, the signer sends the new hash value instead of the whole certificate. To verify the renewed certificate, the verifier needs only one hashing operation to hash the last hash value S_{i+1} to obtain the previous hash value S_i. The verifier can affirm that the certificate's lifetime extension must have been done by the CA because no one can compute S_{i+1} except the CA that created the hash chain. When the certificate lifetime expires, the signer receives a new certificate with a new signature and hash chain. The verifier has to store the new certificate and verifies the new signature.

In the traditional PKC-based authentication scheme, the verifier has to verify the CA's signature each time the certificate is renewed because the signature of the new certificate is different from the old one. Also,

Table I. The Communication overhead (bytes) for our Scheme and Signature-Based Certificate Renewal Scheme

		Initial certificate size	2 certificate renewals	5 certificate renewals	10 certificate renewals
Our Scheme	RSA	370	390	450	550
	ECDSA	178	198	258	358
Signature based schemes	RSA	300	458	842	1,482
	ECDSA	138	266	650	1,290

Table II. The Computational Times and Energy Consumptions for RSA, ECDSA and SHA-1 Operations.

		Certificate verification	Certificate issuance
RSA	Signing	17 ms	2302.7 mJ
	Verifying	0.36 ms	53.7 mJ
ECDSA	Signing	6.33 ms	807 mJ
	Verifying	16.51 ms	963 mJ
SHA-1		1.19 μs / 20 bytes	0.76 J / 20 bytes

Table III. The Computational Overhead for our Scheme and Signature-Based Certificate Renewal Scheme.

		Certificate verification	Certificate issuance
Our Scheme	RSA	Initial certificate: 0.36ms Renewed certificate: 1:19 μs	Initial certificate: 17ms Renewed certificate: 1:19 μs
	ECDSA	Initial certificate: 16.51ms Renewed certificate: 1:19 μs	Initial certificate: 6.33ms Renewed certificate: 1:19 μs
Signature based schemes	RSA	0.36 ms	17 ms
	ECDSA	16.51 ms	6.33 ms

the signer must send the new certificate to the verifier. In contrast, our scheme can reduce the authentication delay because it needs only one lightweight hashing operation to verify a renewed certificate. The communication overhead is also reduced because the sender does not need to send the certificate when it is renewed, the sender only sends the latest hash value, which is much shorter than a certificate.

Our scheme is specifically useful when the verifier and the signer communicate often. Fortunately, this is very usual in the AMI network because of the stationary nature of most of the devices. The scheme is also very useful for delay critical application, where there is a tight restriction on the messages authentication delay. An example for these applications includes the messages that carry the status of the grid and devices' failures.

3. Evaluations

Using the traditional signature-based certificate renewal scheme, the certificate size is 330 and 138 bytes for RSA and ECDSA, respectively. For our scheme [10], the initial certificate size is 370 and 178 bytes for RSA and ECDSA, respectively. There are 40 bytes more in our scheme compared with the traditional signature-based certificate renewal scheme because two hash chain elements should be added. However, for a group of certificate renewals, our scheme requires much less communication overhead than other schemes because the hash value size is much less than the signature size. One hash value with 20 bytes is needed in each renewal instead of a signature with 256 or 64 bytes in case of RSA and ECDSA, respectively. For example, let each certificate be used for 10 renewals and RSA is used, the signature-based certificate renewal scheme requires a total of 1,482 bytes communication overhead, but our certificate renewal scheme requires only 550 bytes. Using ECDSA, our scheme can reduce the overhead of 10 certificate renewals from 1,290 to 358 bytes. This reduction is very useful when the network is scalable and the channels have low bandwidth. All these results are summarized in Table 1. It can be noticed that ECDSA requires less communication overhead than RSA because its signature size is shorter.

In order to estimate the computational times for the signing, verifying, and hashing operations, we have implemented 2,048 bit RSA, 256-bit ECDSA, and SHA-1 hash function using the Crypto++ library and Intel Core 2.83 GHz processor. We have selected these cryptosystems because they are secure and widely used.

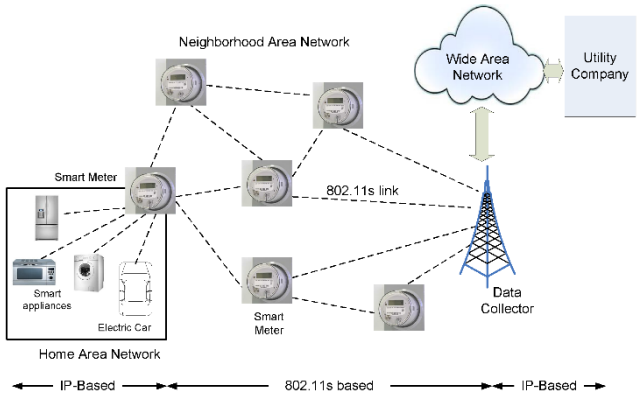

Fig. 2: The AMI network implemented using an IEEE 802.11s-based WMN.

The measured computational times are given in Table 2. Moreover, the energy consumptions of the RSA and SHA-1 operations are measured in [15], [16] and the results are also given in Table 2.

Our scheme can reduce certificate verification time from performing one signature verification to one lightweight hashing operation. The traditional signature-based certificate verification takes 0.36 ms and 16.51 ms for RSA and ECDSA, respectively. However, our scheme requires one signature verification for the initial certificate and one hashing operation with 1.19 μs for the renewed certificates. For the renewed certificates, our scheme requires around 0.007% and 0.019% of the certificate verification time of the signature-based scheme using RSA and ECDSA, respectively. Table 3 gives the certificate verification and issuance times. Since ECDSA signature verification time is greater than that of RSA, ECDSA requires more certificate verification time than RSA.

For the certificate issuance, the CA computes one signature with 17 ms and 6.33 ms for each certificate in the signature-based certificate renewal scheme using RSA and ECDSA, respectively. In our scheme, the CA computes a signature for the initial certificate and a lightweight hashing operation with 1.19 μs for each renewed certificate. Our scheme can reduce the computational cost on the CA from performing one signing operation to one hashing operation for each certificate renewal. It is worth noting that the computational time of one RSA and ECDSA signature is equivalent to 14,286 and 5,319 hashing operations, respectively using the measurements given in Table 2. For 10 certificate renewals and using RSA, the total computational times are 170 ms and 17.011 ms using signature-based scheme and our scheme, respectively. The computational time for computing renewed certificates in our scheme is around one tenth of that of signature-based schemes for the same number of certificate renewals. The reduction from 170.3 ms to 17.011 ms will be useful due to the scalability of the AMI network. It can be seen that ECDSA requires less certificate issuance time than the RSA because it needs less time to compute the signature as indicated in Table 2.

Case Study

Our scheme has also been implemented in a more realistic settings using NS-3 network simulator. In this subsection, we evaluate our scheme in a realistic AMI network. We are interested to assess the impact of the distribution of the renewed certificates on the regular traffic of the network.

Simulation Setup

The considered AMI network is shown in Fig. 2. The underlying communication infrastructure is assumed to be based on Wireless Mesh Networks (WMNs). Specifically, we assume the availability of IEEE 802.11s-based WMNs, which collect data from SMs. In IEEE 802.11s-based WMN, all the nodes act as relays to communicate with each other, which are referred to as Mesh Points (MPs). If an MP is connected to another network, it is assumed to be Mesh Portal Point (MPP) which is the gateway in our case. Note that each SM will act as an MP. However, we will assume the availability of MPs which are used as relays in areas when the SMs cannot find any other neighbouring meters for communication. This wireless infrastructure is one of

the options to implement AMI applications for the SG [17]. Therefore, we created a scenario where the gateway distributes the renewed certificates to each of the SMs in the network.

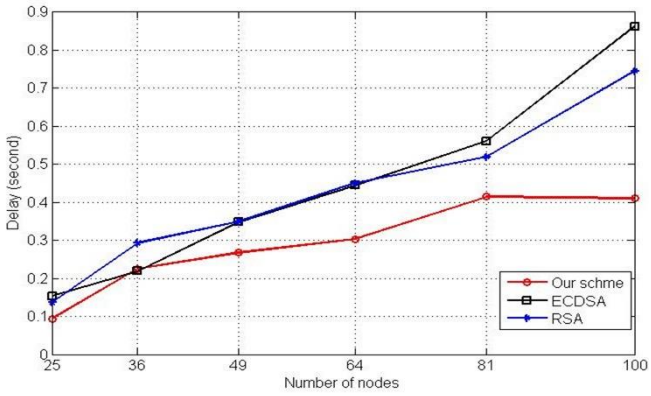

Fig. 3: Average end-to-end delay values vs. network size.

In order to assess the impact of the data traffic on the performance of certificate distribution, we assumed that all the SMs are sending their power readings to the gateway at the time of certificate renewal. As the worst-case scenario, we also assumed that all the SMs are sent renewed certificates in one phase. The meters are sending their power readings every 10 seconds, which is used by some utilities in the real-life applications of AMI [18]. We also assumed a transmission range of 120 m for each of the meters. In each neighbourhood, a gateway will be deployed at one of the power posts and will use 4G/LTE transmission to communicate with the utility's main office. Assuming the availability of multiple gateways, the size of the WMN is assumed to be smaller than the whole network of meters in a city. We generated WMN topologies in the form of grid by using varying number of meters.

The simulations are performed under NS-3 that has a built-in implementation of IEEE 802.11s. The underlying MAC protocol used is 802.11g. The simulations are run for 100 seconds and the results are the snapshot at the end of 100th second.

Performance Metrics and baselines

To assess the performance of our scheme when the renewed certificates are to be distributed, three key metrics are defined for assessment and comparison against signature-based renewal schemes.

- End-to-end Delay: This metric indicates the time needed for a certificate to reach the intended SM including the processing of signatures' verification. We will assess the delay for all meters and get the average. The goal is to minimize this delay since it is important for the SMs to receive these renewals in a timely manner so that they can communicate.
- Packet Delivery Ratio (PDR): This metric indicates the ratio between the number of received packets by SMs to the number of packets generated at the gateway. Due to wireless environments and interference, there may be some packet loss. While TCP retransmits the lost packets, it retries for a given maximum count. Therefore, some packets can still be lost. The goal is to maximize the PDR.
- Required bandwidth: This metric indicates the required bandwidth for certificate renewal for each scheme.

We compared our approach with two other baselines as mentioned in the previous sections. We considered using RSA or ECDSA for signing every renewed certificate. The distribution of these certificates will be done in the same manner. We compared the performance after the initial transmission has been completed in our scheme.

Simulation Results

The experiments are run by using varying number of meters from 25 nodes to 100 nodes by assuming grid topologies. When looking at the delay results in Fig. 3, we observe that our scheme reduces the end-to-end delay in a systematic manner with the increased network size.

This reduced delay is attributed to two things. First, the size of the packets for our scheme is much smaller and this reduces the transmission delay in the network. The possibility of packet loss with smaller packet sizes

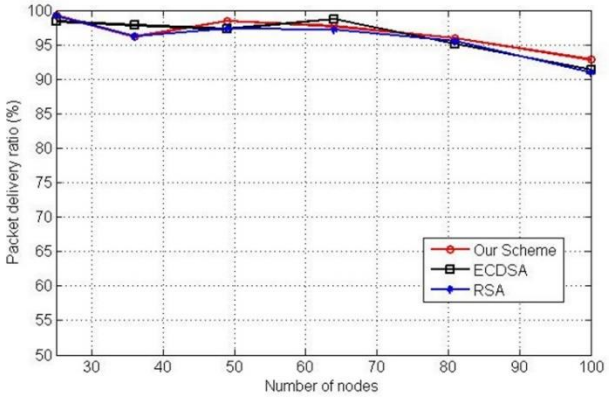

Fig. 4: PDR vs. network size.

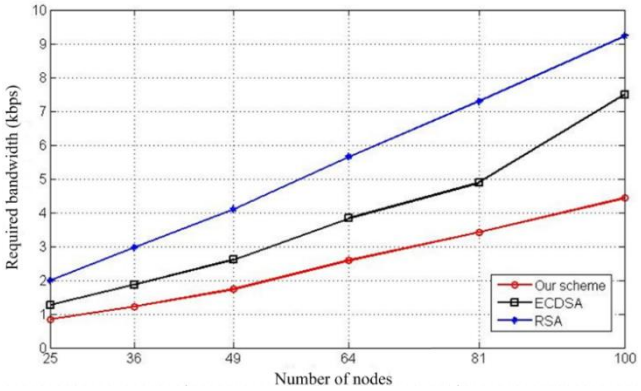

Fig. 5: Required bandwidth vs. network size

is also less making the retransmissions less. Second, the processing delay at the nodes is also less since the verification of a hash is much quicker than RSA or ECDSA signatures as explained before.

We also observe that ECDSA and RSA delays are comparable. While ECDSA can use smaller key sizes, the problem is with the verification times. ECDSA has advantage in signing the messages but we do not take this into account since we assume that this is done at the CA. ECDSA verification takes longer time and thus this increases the end-to-end delays for the packets.

In addition, we looked at PDR to check whether the reduced delay is at the expense of any packet losses in the network. Fig. 4 indicates that the PDR values for our scheme is very similar to RSA and ECDSA. This indicates that in terms of reliability, our approach can provide the same guarantees as ECDSA and RSA. Given the smaller packet sizes in our scheme, the PDR values are not surprising.

Finally, we checked the required bandwidth for all approaches. The results in Fig. 5 demonstrated that the required bandwidth for our scheme is the least. This is because of the smaller data size carried in the network. Thus, the amount of available bandwidth for other data traffic will be much higher in the case of using our scheme. Note that the required bandwidth of the RSA-based scheme is the highest due to the large size of signatures.

Overall, the results are promising in the sense that there is significant advantage in terms of delay and required bandwidth when our scheme is used. Given that 802.11s-based WMN infrastructure will be used in many applications other than AMI (e.g., communication of electric vehicles, distributed demand response, etc.), it is crucial to reduce the amount of traffic in the network due to certificate renewals.

4. Related Works

PKC is the most effective cryptography for securing the AMI communications [19]. It has been used in many proposed schemes for smart grid communications, such as [3], [20]. In [21], Khurana et al. have identified public key management as a challenge in smart grid due to the system scalability and complexity.

In spite of its importance, very few works have investigated the efficient use of PKC in smart grid and most of them focused on certificate revocation. Different aspects of certificate revocation problem in smart grid applications were discussed in [5]–[7] without providing a solution to AMI networks. In [8], Akkaya et. al proposed an efficient grouping algorithm to distribute CRLs in an IEEE 802.11s based AMI networks. The CA generates a single revocation list for each group. In [9], an efficient certificate eradication scheme is proposed for large scale AMI networks that use Bloom filters and without false positives.

Unlike these works, this paper presents promising results for the implementation of our prior proposed scheme in [10] on efficient certificate renewal for AMI networks.

5. Conclusions

In this paper, we have extensively investigated the performance our previous proposal about efficient certificate renewal scheme for AMI networks. First, quantitative analysis was carried out to compare our scheme against signature-based certificate renewal schemes and our evaluations have demonstrated that our proposals require much less overhead than the traditional approaches. Moreover, all schemes were implemented in a network model with AMI realistic settings using NS-3 to evaluate their performance.

Simulation results demonstrate the improved performance in computational cost, communication overhead, end-to-end delay, packet delivery ratio, and required bandwidth for our scheme over signature-based certificate renewal scheme. The simulation results are promising in the sense that there is significant advantage in terms of delay, packet delivery ratio and required bandwidth when our scheme is adopted.

6. Acknowledgements

This publication was made possible by NPRP grant number 9-055-2-022 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

7. References

[1] M. Raya and J.-P. Hubaux, “Securing vehicular ad hoc networks,” Journal of Computer Security, vol. 15, no. 1, pp. 39–68, 2007.

[2] M. Mahmoud and X. Shen, “Esip: Secure incentive protocol with limited use of public-key cryptography for multi-hop wireless networks,” IEEE Transactions on Mobile Computing, vol. 10, no. 7, pp. 997–1010, July 2011.

[3] R. Anderson and S. Fuloria, “Who controls the off switch?” in First IEEE International Conference on Smart Grid Communications (Smart-GridComm), October 2010, pp. 96 – 101.

[4] P. Wohlmacher, “Digital certificates: A survey of revocation methods,” in Proc. of the ACM Workshops on Multimedia, Los Angeles, California, USA, 2000, pp. 111–114.

[5] N. Islam, “Certificate revocation in vehicular ad hoc networks: a novel approach,” in 2016 International Conference on Networking Systems and Security (NSysS), Jan 2016, pp. 1–5.

[6] M. Mahmoud, J. Misic, and X. Shen, “Efficient public-key certificate revocation schemes for smart grid,” Proc. of IEEE Global Communication Conference, Atlanta, GA, USA, December 9-13 2013.

[7] M. Mahmoud, J. Misic, K. Akkaya, X. Shen, “Investigating public-key certificate revocation in smart grid,” IEEE Journal on Internet of Things (IoT), to appear.

[8] K. Akkaya, K. Rabieh, M. Mahmoud, and S. Tonyali, “Customized certificate revocation lists for IEEE 802.11s-based smart grid AMI networks,” IEEE Transactions on smart grid, to appear.

[9] K. Rabieh, M. Mahmoud, K. akkaya, and S. Tonyali, “Scalable certificate revocation schemes for smart grid ami networks using bloom filters,” IEEE Transactions on Dependable and Secure Computing, vol. PP, no. 99, pp. 1–1, 2015.

[10] M. Mahmoud, J. Misic, and X. Shen, “A scalable public key infrastructure for smart grid communications,” Proc. of IEEE Global Communications Conference (GLOBECOM), pp. 784–789, USA, 2013.

[11] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” Communications of the ACM, vol. 26, no. 1, pp. 96–99, 1983.

[12] D. Johnson, A. Menezes, and S. Vanstone, “The elliptic curve digital signature algorithm (ecdsa),” International Journal of Information Security, vol. 1, no. 1, pp. 36–63, 2001.

[13] W. Dai, “Crypto++ library 5.6.0,” http://www.cryptopp.com.

[14] “Network simulator - ns - 3,” http://www.isi.edu/nsnam/ns/index.html.

[15] N. Potlapally, S. Ravi, A. Raghunathan, and N. Jha, “A study of the energy consumption characteristics of cryptographic algorithms and security protocols,” IEEE Transactions Mobile Computing, vol. 5, no. 2, pp. 128–143, Mar./Apr. 2006.

[16] Y. Xiao, F. Li, and H. Chen, “Handbook of security and networks,” World Scientific Publishing Company, ISBN-13: 978-9814273039, 2011.

[17] N. Saputro, K. Akkaya, and S. Uludag, “A survey of routing protocols for smart grid communications,” Comput. Netw., vol. 56, no. 11, pp. 2742–2771, Jul. 2012.

[18] “Korean electric power research institute,” 2013. [Online]. Available: http://www.kepri.re.kr/

[19] A. Metke and R. Ekl, “Security technology for smart grid networks,” IEEE Transactions on Smart Grid, vol. 1, no. 1, pp. 99–107, June 2010.

[20] P. Akula, M. Mahmoud, K. Akkaya, and M. Song, “Privacy-preserving and secure communication scheme for power injection in smart grid,” Proc. of IEEE International Conference on Smart Grid Communications (IEEE SmartGridComm), Miami, Florida, USA, 2-5 November 2015.

[21] H. Khurana, M. Hadley, N. Lu, and D. Frincke, “Smart-grid security issues,” IEEE Security and Privacy, vol. 8, no. 1, pp. 81–85, January-February 2010.