2014

# A Security Protocol for Advanced Metering Infrastructure in Smart Grid

Feng Ye
*University of Nebraska-Lincoln*

Yi Qian
*University of Nebraska-Lincoln*, yqian2@unl.edu

Rose Qingyang Hu
*Utah State University*

Globecom 2014 - Communication and Information System Security Symposium
Pages: 649 - 654, DOI: 10.1109/GLOCOM.2014.7036881

# A Security Protocol for Advanced Metering Infrastructure in Smart Grid

Feng Ye, Yi Qian
Department of Computer and Electronics Engineering
University of Nebraska-Lincoln, NE, USA

Rose Qingyang Hu
Department of Electrical and Computer Engineering
Utah State University, Logan, UT, USA

***Abstract*—In this paper, we propose a security protocol for advanced metering infrastructure (AMI) in smart grid. AMI is one of the important components in smart grid and it suffers from various vulnerabilities due to its uniqueness compared with wired networks and traditional wireless mesh networks. Our proposed security protocol for AMI includes initial authentication, secure uplink data aggregation/recovery, and secure downlink data transmission. Compared with existing researches in such area, our proposed security protocol let the customers be treated fairly, the privacy of customers be protected, and the control messages from the service provider be delivered safely and timely.**

## I. Introduction

In smart grid, security issues are more important than those in traditional power grid since the communication network has been updated to bidirectional and the data is transmitted in much larger quantity and more frequently [1, 2]. An advanced metering infrastructure (AMI) is the system that collects and analyzes data from smart meters, and giving intelligent management of various power-related applications and services based on that data [3]. An AMI is basically a wireless mesh network (WMN) where each smart meter is a node, a data aggregate node (DAP) which usually locates in the center of a neighborhood functions as the gateway of all the smart meters in that neighborhood [4, 5]. The smart meters connect the meter data management system (MDMS) through the DAP in multi-hop mode, and the DAP connects to the MDMS through a backbone network (e.g., optical fiber).

In AMI, the data in uplink transmission from smart meters to the MDMS includes secret information, for example, the power usage of a household. Those data will be collected by the MDMS and be further applied to determine the power generation and the usage of renewable energy. The control data in downlink transmission involves the price and tariff information, which affect the demand side response and finally lead to a more efficient power grid [6, 7]. Although AMI appears to be a WMN and security issues have been widely discussed for traditional WMN [8–10], however, AMI is different from traditional WMN mainly in threefold. First, each smart meter must be available and be treated equally in the network since fairness must be applied to each of the customers while traditional WMN does not emphasize availability for each wireless node let alone fairness. Second, the deployment of smart meters are fixed and in specific orders since they are deployed in each household and the houses are in fixed position in most cases, while the wireless nodes in traditional WMN are usually deployed randomly and redundantly. Third, the uplink transmission and downlink transmission in AMI are asymmetric where the uplink transmission consists of different data from each smart meter to the MDMS and the most of the downlink transmissions are in broadcast mode, while in traditional WMN, the uplink or downlink can even barely be distinguished. Thus the security protocols must be redesigned to fit the uniqueness of AMI.

There are several researches for the security issues in AMI [11–14], however, there are very few comprehensive security protocols for AMI. In [14], the authors proposed a protocol called integrated authentication and confidentiality (I-AC) which involves the initial authentication of a smart meter, and the security in both uplink and downlink transmissions. However, IAC has several problems to be addressed. 1) The smart meters are not treated equally where some of them are chosen to be the backbone nodes and proceed with security protocol, while the others must go through the backbone nodes however the backbone nodes selection does not have any security concern. 2) The initial authentication process cannot prevent replay attack or even forgery if the initial request is overheard by the attacker. 3) The security protocol in uplink transmission cannot handle multiple incoming data at an intermediate node. 4) Compromise of a node will at least endangers another node since they share the same secret key for message encryption. 5) The security protocol for downlink transmission is too complicated since IAC did not consider broadcast scenario as the main transmission mode for downlink. 6) Once a node malfunctions in the network, IAC cannot function any longer. In this paper, we propose a comprehensive security protocol which addresses those shortages in IAC while maintaining the good security features.

The rest of the paper is organized as follows. In section II, we present the studied network model and discuss the recovery of malfunctioning nodes. In section III, we present the proposed security protocol for AMI. In section IV, we show the results of performance analysis. In section V, we give the conclusion and future work.

## II. Network Model

Let a conflict graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be the wireless mesh network of AMI, where each node $n_i \in \mathcal{V}$ (other than the final gateway which is a DAP) is a smart meter, an edge is a communication link for the two corresponding nodes. In the studied AMI, there are two types of nodes, one is *active* and the other is *uninitialized*. An active node is a node such that it has been authenticated by the authentication server (AS) to join the AMI communication and is functioning in a healthy status. An uninitialized node is (but not limited to) the newly installed one. If a node has recovered from malfunctioning status, or updated, or lost connection to all of its active neighbors, or moved to another location, it is also defined as uninitialized. An example of such network is shown in Fig. 1, where the DAP functions as a gateway and the AS.

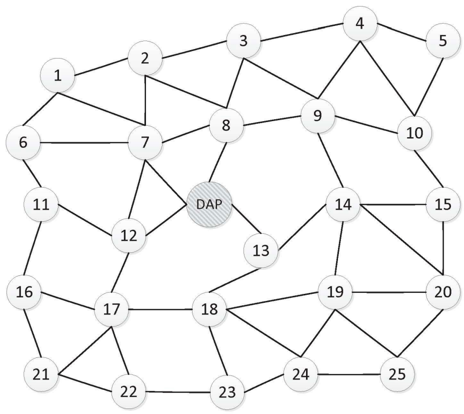



Fig. 1. An example of the conflict graph $\mathcal{G}$.

Routing in the uplink transmission is achieved by building a shortest path routing tree based on all the active nodes in the conflict graph. Finding the shortest path for every active node to the DAP can be achieved using any well-know algorithm such as Dijkstra's algorithm [15]. Assuming all the nodes in $\mathcal{G}$ are active, we then show one routing example as a rooted shortest path tree in Fig. 2. Note that the parent node is always initialized before its child nodes, details will be discussed in section III.

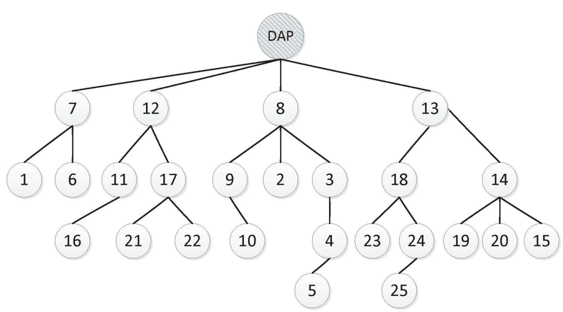



Fig. 2. The corresponding rooted spanning tree $\mathcal{T}$.

## A. Uplink Transmission Recovery

Since the proposed security protocol is for an AMI network in operations, once the network is broken due to some malfunctioning nodes, the network must be recovered before the secure transmission recovers. In this subsection, we discuss the recovery of the uplink transmission. In the shortest path routing tree, once a node failure occurs, the links connecting its child nodes will break. Therefore the child nodes should refer to the conflict graph $\mathcal{G}$ and look for the active neighbors which are not descendent in the shortest path tree reroute themselves through the closest ones to the DAP.

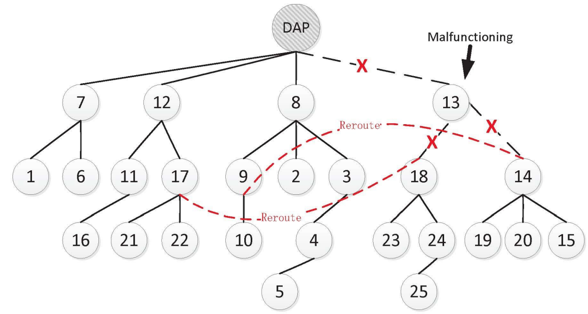



Fig. 3. Example of routing recovery when $n_{13}$ malfunctions.

An example of malfunctioning $n_{13}$ is shown in Fig. 3. Assuming that $n_{17}$ is initialized before $n_{18}$, which enables $n_{18}$ to be the child node of $n_{17}$ without initialization. The same assumption applies to $n_9$ and $n_{14}$ where $n_9$ is initialized before $n_{14}$. Once $n_{13}$ is down, links $(13, 18)$ and $(13, 14)$ no longer exist. Then, $n_{18}$ will reroute itself through $n_{17}$ and $n_{14}$ will reroute itself through $n_9$, no other nodes need to change in the rerouting process. Once $n_{13}$ recovers, it will start the initialization process and directly connect to the AP. $n_{18}$ and $n_{14}$ will discover their *new* shortest path through $n_{13}$ and start the initialization process again in order to reroute through it.

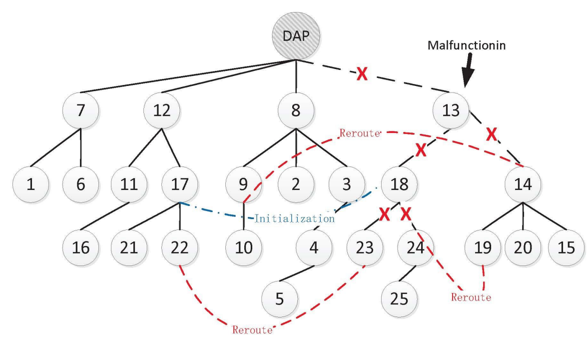



Fig. 4. First step of routing recovery when $n_{13}$ malfunctions where $n_{18}$ cannot reroute through $n_{17}$.

If $n_{18}$ is initialized before $n_{17}$, then $n_{18}$ is not allowed to connect to $n_{17}$ without initialization. The descendent nodes of $n_{18}$ ($n_{23}$, $n_{24}$ and $n_{25}$) will reroute themselves to their active neighbors (as illustrated in Fig. 4). Once $n_{18}$ finishes the initialization and connects to $n_{17}$, there is no need for $n_{23}$ or $n_{24}$ to reconnect to $n_{18}$ since the distance will be the same. If later on $n_{18}$ reconnects with $n_{13}$, $n_{23}$ and $n_{24}$ will start initialization again in order to route through $n_{18}$ for shortest path to the AS.

The worst case is that there exists no available route without going backwards (through child nodes), all the nodes in the sub-rooted tree with root $n_{18}$ will then start initialization process until all of them have connection to the AS. In order to insure the recovery process of most situations, the conflict graph $\mathcal{G}$ should have connectivity constraints, for example, the smallest degree of a node $\delta \geq 3$. However, this connectivity issue is beyond the scope of this paper, we assume that each node should at least have connection with its physical neighbors and the neighborhood follows grid (or pseudo-grid) topology and thus the connectivity is good enough.

## III. Proposed Security Protocol for AMI

### A. Initial Authentication

There are two types of nodes in the AMI, one is *active* and the other is *uninitialized*. An active node is a node such that it has been authenticated by the AS to join the AMI communication and is functioning in a healthy status. An uninitialized node is (but not limited to) the newly installed one. If a node has recovered from malfunctioning status, or updated, or lost connection to all of its active neighbors, or moved to another location, it is also defined as uninitialized and thus must start the initial authentication process to join the AMI. Whatever status it is, an uninitialized node that does not function in the AMI properly must be authenticated through the initialization process.

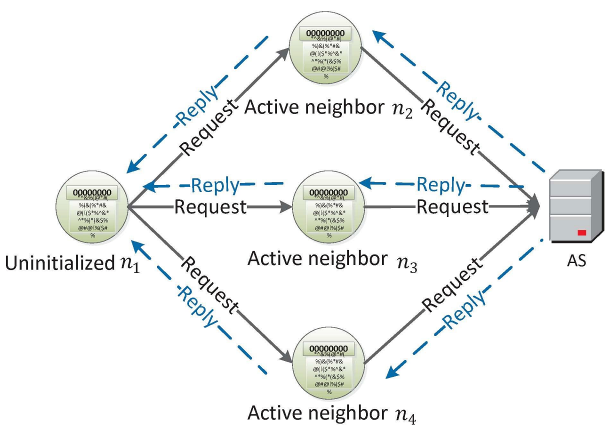



Fig. 5. Initial authentication process.

For example, if $n_1$ wants to join the AMI, the initialization process goes through all of its active neighbors (e.g., $n_2$, $n_3$ and $n_4$). As illustrated in Fig. 5, $n_1$ sends requests to the AS through all of its active neighbors, and receives different reply messages from the AS through its active neighbors as well. Through this initial authentication process, there are mainly three tasks accomplished,

- $n_1$ is authenticated to be an active node and join the AMI;
- $n_1$ establishes secure connection to the AS through one of its active neighbors which has the shortest distance to the AS;
- $n_1$ establishes backup secure connection to the AS through the rest of its active neighbors.

The initial authentication process through each active neighbor is similar, we give a detailed illustration of the process through one active neighbor (e.g., $n_2$). Throughout the process, there are mainly three entities involved, $n_1$, $n_2$ and the AS, although it is possible that other nodes are involved for relaying the message, however they do not affect the process as long as they relay the message correctly since they do not get new information from either communications. Among the three entities, there are three mutual authentications to be achieved in order to guarantee the security, one is between $n_1$ and the AS, one is between $n_2$ and the AS, and the other one is between $n_1$ and $n_2$. The mutual authentication between $n_1$ and the AS is obvious since the AS will only allow genuine node join the AMI and the node will also trust the AS. The mutual authentication between $n_2$ and the AS is to ensure that $n_2$ is active and is trusted to relay the request from $n_1$. The mutual authentication between $n_1$ and $n_2$ is to help further establish secure communications from $n_1$ to $n_2$. It is assumed that each node has a pre-shared secret key (e.g., $K_1$ for node $n_1$ and $K_2$ for node $n_2$) with the AS before initialization. Each active node has been assigned with an active secret key (e.g., $k_2$ for $n_2$) mainly for uplink data encryption, this active secret key is also used to verify if this node is active or not. Similar to $K_2$, $k_2$ is only known to $n_2$ and the AS. In order to establish a secure connection from $n_1$ to the AS, an active secret key $k_1$ must be generated by the AS and assigned to $n_1$ during the initialization process. Note that $n_1$ does not bare $k_1$ before initialization process, only $K_1$ is known to $n_1$.

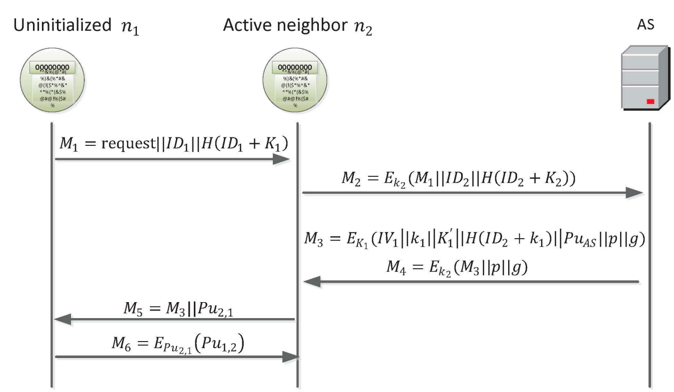



Fig. 6. Detailed initial authentication process through one active neighbor.

As shown in Fig. 6, the whole initialization process involves 5 hand-shakes and 6 messages. In the first hand-shake between $n_1$ and $n_2$, $n_1$ sends $M_1 = \text{request}||ID_1||H(ID_1 + K_1)$ to the AS through $n_2$, where $H(\cdot)$ is a simple hash function, and '$+$' is *XOR*. $H(ID_1 + K_1)$ is used to provide authentication of $n_1$ at the AS side since the AS is the only one besides $n_1$ to compute the hash value.

In the second hand-shake, $n_2$ appends $H(ID_2 + K_2)$ to $M_1$ which is used for the AS to authenticate $n_2$ as a genuine node, $n_2$ then encrypts the entire message and appended its own identification with $k_2$, it is used to protect its identity verification code $H(ID_2 + K_2)$ and also let the AS authenticate its active status. The message send from $n_2$ to the AS is $M_2 = ID_2||E_{k_2}(M_1||H(ID_2 + K_2))$, where $E_k(\cdot)$ is the encryption function with key $k$.

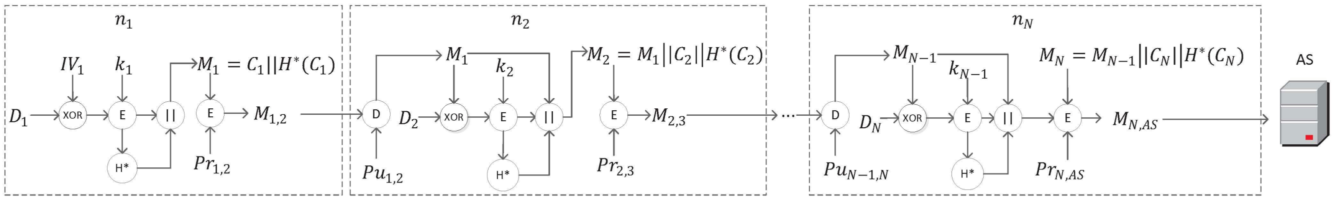



Fig. 7. Data aggregation process in uplink transmission.

Once the AS receives $M_2$, it authenticates $n_2$ by decrypting $M_2$ using $k_2$ and compute $H(ID_2 + K_2)$. The AS then authenticate $n_1$ by computing $H(ID_1 + K_1)$. Once $n_1$ is authenticated, the AS generates an initial vector $IV_1$ for further uplink transmission, an active secret key $k_1$, a new $K_1'$ to replace $K_1$ for further initial authentication of $n_1$ since $M_1$ was transmitted unprotected and $H(ID_1+K_1)$ as a verification code is revealed (note that $H(ID_2+K_2)$ is encrypted and thus $K_2$ is not required to update). Public key of the AS $Pu_{AS}$ is sent to $n_1$ for further downlink transmission protocols. The AS also generates $H(ID_2+k_1)$ for $n_1$ to authenticate $n_2$. Finally, the AS generates $p,\ g$ for public key generation between $n_1$ and $n_2$. It is possible for the AS to generate the public keys or session keys for $n_1$ and $n_2$ if the devices are not advanced enough to finish the tasks, however, it is safer to keep the nodes as independent as possible to other nodes and the AS. Therefore, let $n_1$ and $n_2$ generate their own public/private keys or session keys within one-hop transmission is recommended in this protocol. In summary, the message sent from the AS to $n_1$ is $M_3 = E_{K_1}(IV_1||k_1||K_1'||H(ID_2+k_1)||PU_{AS}||p||g)$. Since $M_3$ is relayed by $n_2$, the AS appends $p,\ g$ to $M_3$ and encrypts it by $k_2$ so that $n_2$ can follow the public/private key generation of $n_1$. Overall, the message sent from the AS to $n_2$ in the third hand-shake is $M_4 = E_{k_2}(M_3||p||g)$.

After $n_2$ receives $M_4 = E_{k_2}(M_3||p||g)$, it decrypts $M_4$ to authenticate the AS and get $p,\ g$. Based on $p,\ g,\ n_2$ generates a pair of public/private keys $Pu_{2,1}/Pr_{2,1}$, and sends $M_5 = M_3||Pu_{2,1}$ to $n_1$ in the fourth hand-shake. Until now, $n_1$ has received all the information from the AS and is just one step from being an active node if it wants to route through $n_2$. Based on $p,\ g,\ n_1$ generates a pair of public/private keys $Pu_{1,2}/Pr_{1,2}$, and sends $M_6 = E_{Pu_{2,1}}(Pu_{1,2})$ to $n_2$ in the fifth hand-shake. Note that $Pu_{1,2}$ is sent in a secure way since it is encrypted with $Pu_{2,1}$. Although $Pu_{2,1}$ is sent in plaintext, only $n_2$ is able to reveal the public key of $n_1$. After $n_2$ receives $Pu_{1,2}$, both $Pu_{2,1}/Pr_{2,1}$ will be discarded.

Until now, $n_1$ is fully initialized and it is able to join uplink communications through $n_2$. The initial authentication processes through other active neighbors are similar, the AS sends back the same $IV_1,\ k_1,\ K_1',\ Pu_{AS},\ p$ and $g$. In the final hand-shake, $n_1$ will send the same $Pu_{1,x}$ to node $n_x$ encrypted with $Pu_{x,1}$. By doing so, $n_1$ shares the same public key to all of its active neighbors. Therefore, $n_1$ is able to join the uplink transmission through any of the active neighbors, in other words, both operating and backup secure communication channels are established through the initial authentication process.

*1) Security Analysis:* Confidentiality of the authentication request is unnecessary, therefore it is not provided. Applying this protocol makes a genuine node unforgeable. As discussed before, the initial authentication of $n_i$ relies on the pre-shared key $K_i$, which cannot be forged since it is pre-installed in both the AS and $n_i$. On the other hand, Although the request and valid verification code can be overheard, however the information is useless since $K_i$ is subject to change after each successful initial authentication. Therefore, reply attack will not harm the initial authentication as well.

### *B. Security Protocol in Uplink Transmission*

In the uplink transmission, data from each node is aggregated in a chain topology and is finally delivered to the DAP. Among all the security requirements, data confidentiality is the most important issue for the uplink data. Data integrity is also very important since the wrong data may cause unnecessary loss of the power generation. Sender authentication shall be considered as well if there is enough computational resources since the nodes shall only aggregate the data sent from active nodes. To achieve all those requirements mentioned above, we propose the security protocol for data aggregation in uplink transmission as shown in Fig. 7. Suppose in one path there are $N$ nodes with an order of $(n_1, n_2, \ldots, n_N)$. As the first one of the aggregation, $n_1$ mixes its raw data $D_1$ with $IV_1$ and encrypts it with $k_1$ so that confidentiality can be achieved. $H^*(\cdot)$ is a hashed message authentication code function which provides data integrity. Finally, $n_1$ signs the total message with $Pu_{1,2}$ so that $n_2$ can verify that the data is from $n_1$ which is an active node. The intermediate nodes first decrypt the incoming data with the private key of the child node and mix their raw data with the previous data and then follow the same steps as the first node.

If an intermediate node has multiple child nodes, it treats each of them as a separate chain and aggregates its own data to one of the incoming data while simply padding the data from the other child nodes to it with flags. The details are shown in Fig. 8. Assume $n_p$ has two child nodes $n_i$ and $n_j$, and $n_p$ chooses to aggregate incoming data from $n_i$. Then $n_p$ follows the usual steps dealing with $D_p$ and $M_{i,p}$. For $M_{j,p}$, $n_p$ authenticates the sender by getting $M_j$, and simply flags $M_j$ such that $f_0||M_j||f_1$ to the original $M_p$, thus $M_p = f_0||M_j||f_1||C_p||H^*(C_p)$.

Once the AS receives the aggregated data, it starts the recovery process of the data. The AS first authenticates the

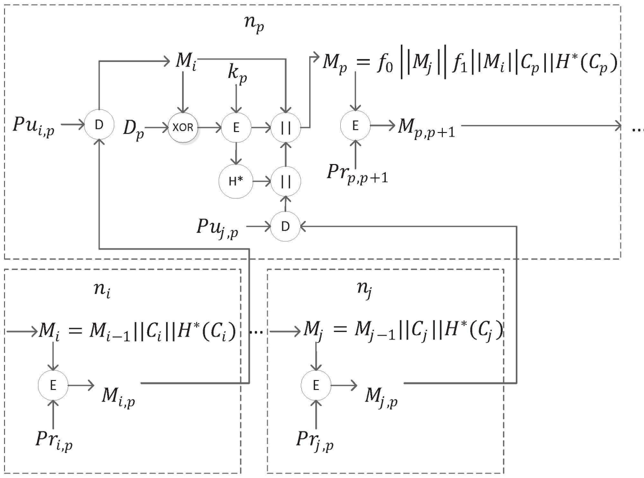



Fig. 8. Multi-flow data aggregation process.

child node by decrypting the receiving data with the pre-shared public key $Pu_{N,AS}$. Before recovering the raw data, the AS needs to verify the data integrity by the process shown in Fig.9. Since the data of each node are not further processed by nodes after it, if some of the data corrupt, the AS will simply discard them instead of wasting the whole message from that transmission path.

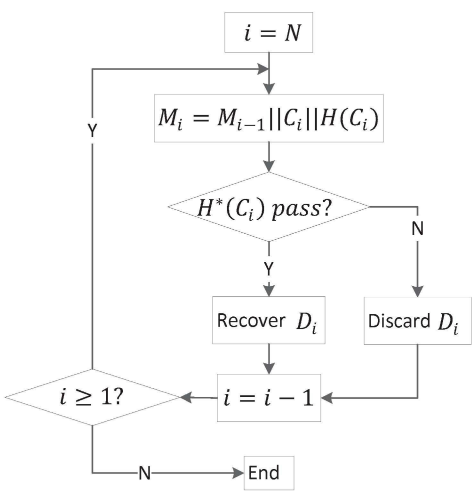



Fig. 9. Data integrity check in uplink transmission.

The detailed raw data recovery process (without integrity check) is shown in Fig. 10. Message $M_i = M_{i-1}||C_i||H^*(C_i)$, after verifying the data integrity, the AS decrypts $C_i$ and $XOR$ the result with $M_{i-1}$ to recover $D_N$. Note that $D_1$ is recovered by $XOR\ IV_1$. If the message includes data from multiple chains, the AS extracts the message between $f_0$ and $f_1$ first and recovers the data following the same process as shown in Fig. 10 without verifying the sender authentication (the decryption process with $Pu_{N,AS}$).

*1) Security Analysis:* Confidentiality is provided by mixing the raw data with previous incoming data and encryption with active secure key. The message cannot be manipulated since message integrity is verified using HMAC. The message is unforgeable unless a node is totally compromised since the HMAC and encryption must use both $K$ and $k$. The message is non-repudiable since it is signed with a private key of the sender.

### *C. Security Protocol in Downlink Transmission*

The downlink transmission involves control messages from the DAP to the nodes. Most of the control messages (e.g., price and tariff information) are for all the smart meters in the neighborhood, where the confidentiality is not as important as that of the uplink data. However, message integrity is important. Message manipulation will cause further responding in power usage and will finally result in unnecessary over- or under-power generation. Let $C_B$ be the control message to be broadcast. To provide message integrity, a MAC (achieved by HMAC function $H^*$) is appended to the original message, the entire message is then signed with $Pu_{AS}$ as digital signature to provide non-repudiation and sender authentication. In all, $M_B = E_{Pu_{AS}}(C_B||H^*(C_B))$.

Some of the control messages (e.g., request for update) are for a specific node (e.g., $n_i$), let such control message be $C_i$. Apparently, message integrity, non-repudiation and sender authentication shall still be provided, moreover, confidentiality of the message is also important, therefore the message is encrypted with $k_i$ such that $M_B = E_{Pu_{AS}}(E_{k_i}(C_B||H^*(C_B)))$. Unlike $M_B$, broadcasting $M_i$ is a waste of resource and is unnecessary. However, sending $M_i$ through the corresponding uplink path may reduce the availability of the message. Therefore, we propose to send such specific control message to $n_i$ through all of its active neighbors. For example (regarding to Fig. 1 and Fig. 2), if specific control message $M_{23}$ is to be transmitted to $n_{23}$, the AS will send it to $n_{18}$, $n_{22}$ and $n_{24}$ which are active neighbors of $n_{23}$, all of them will then forward $M_{23}$ to $n_{23}$. The downlink transmission of $M_{23}$ is shown in Fig. 11.

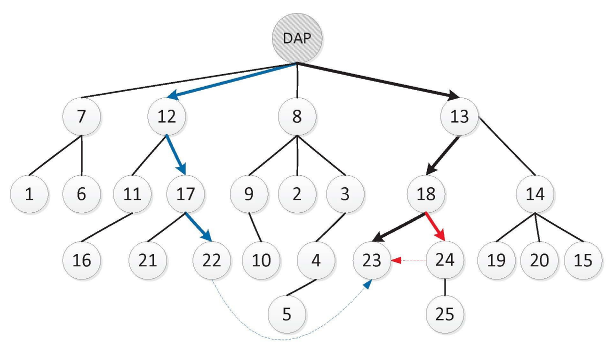



Fig. 11. Example routing of downlink message $C_{23}$.

*1) Security Analysis:* First of all, the control message is unforgeable since it is signed by the AS. For the same reason, the control message is also non-repudiable. The control message cannot be manipulated since HMAC is applied for data integrity. For downlink transmission to a specific node, confidentiality is provided by encrypting the message with the

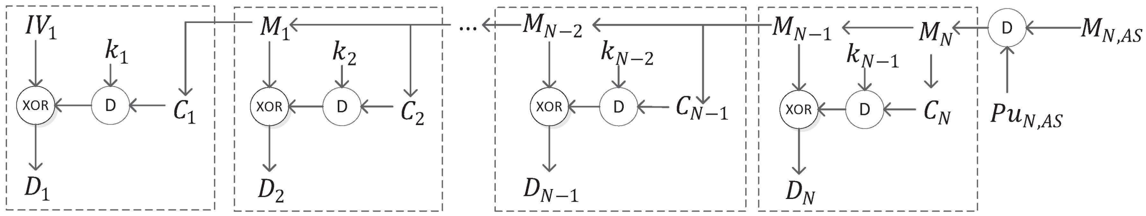
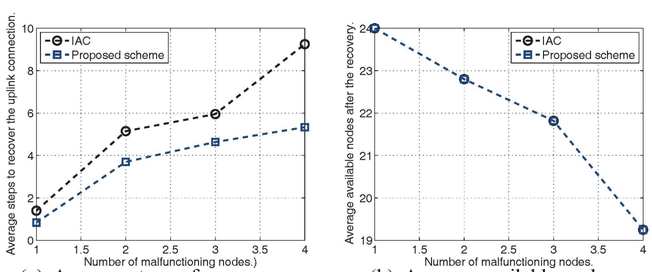



Fig. 10. Data recovery process in uplink transmission.

pre-shared key. Moreover, the harm of DoS is reduced by increasing of availability and delivering the control message through multiple paths for both broadcasting control message and specific control message.

## IV. Performance Analysis

The most important improvement in the proposed security protocol compared with IAC is the uplink recovery process, we then focus on the comparison of the recovery performance. The analyzed network is $\mathcal{G}$ shown in Fig. 1 and the shortest path routing tree $\mathcal{T}$ is shown in Fig. 2. In Fig. 12(a) we show the average steps of recovering the uplink connection w.r.t the number of malfunctioning nodes. Since IAC must recover all the nodes that are prior to a malfunctioning node while our proposed scheme focuses on the child nodes of the malfunctioning node only, it is obvious that the average recovery steps can be lower.




(a) Average steps of recovery. (b) Average available nodes.

Fig. 12. Network performance in connection recovery.

On the other hand, Fig. 12(b) shows the number of available nodes after the recovery of the connection. We can see that both the proposed scheme and IAC perform the same. When the number of malfunctioning nodes is 1, all the other nodes will not be affected after the recovery process. However, when the number of malfunctioning nodes grows to 2 or more, it is not guaranteed that all the other nodes can get access to the MDMS through the DAP since the node-connectivity of $\kappa(G) = 2$. In order to improve the availability, the connectivity issue will be considered in the future work by adding multiple DAPs or dummy nodes.

## V. Conclusion and Future Work

In this paper, we propose a comprehensive security protocol for AMI which includes the initial authentication of a smarter meter, secure uplink data aggregation/recovery, and secure downlink data transmission. Compared with existing IAC protocol, our proposed protocol addresses several concerns and achieves fairness for each smart meter in AMI. In the future work, we will focus on deploying an intrusion detection system to monitor the active nodes which will further enhance the security in AMI.

## References

[1] Y. Yan, Y. Qian, H. Sharif, D. Tipper, "A Survey on Smart Grid Communication Infrastructures: Motivations, Requirements and Challenges", *IEEE Communications Surveys and Tutorials*, vol.15, no.1, pp.5-20, First Quarter 2013.

[2] Y. Yan, Y. Qian, H. Sharif and D. Tipper, "A Survey on Cyber Security for Smart Grid Communications," *IEEE Communications Surveys and Tutorials*, vol.14, no.4, pp.998-1010, 4-th quarter, 2012.

[3] J. Zhou, R.Q. Hu, and Y. Qian, "Scalable Distributed Communication Architectures to Support Advanced Metering Infrastructure in Smart Grid," *IEEE Transactions on Parallel and Distributed Systems*, vol.23, no.9, pp.1632-1642, Sept. 2012.

[4] P. Kulkarni, S. Gormus, F. Zhong and F. Ramos, "AMI Mesh NetworksłA Practical Solution and Its Performance Evaluation," *IEEE Transactions on Smart Grid*, vol.3, no.3, pp.1469-1481, Sept. 2012.

[5] M.R. Abid, A. Khallaayoun, H. Harroud, R. Lghoul, M. Boulmalf and D. Benhaddou, "A Wireless Mesh Architecture for the Advanced Metering Infrastructure in Residential Smart Grids," *IEEE Green Technologies Conference*, pp.338-344, April 2013.

[6] H. Li, X. Liang, R. Lu, X. Lin and X. Shen, "EDR: An efficient demand response scheme for achieving forward secrecy in smart grid," *IEEE GLOBECOM'12*, pp.929-934, Dec. 2012.

[7] B. Kim and O. Lavrova, "Two hierarchy (home and local) smart grid optimization by using demand response scheduling," *IEEE PES Conference on Innovative Smart Grid Technologies Latin America*, pp.1-8, April 2013.

[8] M.A. Hamid, M.S. Islam and C. Hong, "Developing Security Solutions for Wireless Mesh Enterprise Networks," *IEEE WNCN'08*, pp.2549-2554, March 31-April 3 2008.

[9] K. Ren, S. Yu, W. Lou and Y. Zhang, "PEACE: A Novel Privacy-Enhanced Yet Accountable Security Framework for Metropolitan Wireless Mesh Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol.21, no.2, pp.203-215, Feb. 2010.

[10] E.L. Witzke, J.P. Brenkosh, K.L. Green, L.E. Riblett and J. M. Wiseman, "Encryption in mobile wireless mesh networks," *IEEE ICCST'12*, pp.251-256, Oct. 2012.

[11] R. Berthier and W.H. Sanders, "Specification-Based Intrusion Detection for Advanced Metering Infrastructures," *IEEE 17th Pacific Rim International Symposium on Dependable Computing*, pp.184-193, Dec. 2011.

[12] B. Vaidya, D. Makrakis and H. Mouftah, "Secure multipath routing for AMI network in Smart Grid," *IEEE IPCCC'12*, pp.408-415, Dec. 2012.

[13] M.S. Thomas, I. Ali and N. Gupta, "A secure way of exchanging the secret keys in advanced metering infrastructure," *IEEE POWERCON'12*, pp.1-7, Oct. 30-Nov. 2 2012.

[14] Y. Yan, R.Q. Hu, S.K. Das, H. Sharif and Y. Qian, "An efficient security protocol for advanced metering infrastructure in smart grid," *IEEE Network*, vol.27, no.4, pp.64-71, July-August 2013.

[15] E.W. Dijkstra, "A note on two problems in connexion with graphs". *Numerische Mathematik 1*, pp.269C271, 1959.