A Secured Protocol for IoT Devices in Tactical Networks

Wei Wang*, Zicheng Chi†, Xin Liu‡, Ananth Vishnu Bhaskar§, Ankit Baingane§, Ryan Jahnige§, Qingquan Zhang§, and Ting Zhu‡
Saint Louis University *
Cleveland State University †
University of Maryland, Baltimore County§
The Ohio State University‡
Email: *wei.wang.5@slu.edu, †z.chi@csuohio.edu, §{abhask1, ankitb1, jahn6, q}@umbc.edu, ‡{liu.10663, zhu.3445}@osu.edu

MILCOM 2022 - 2022 IEEE Military Communications Conference (MILCOM) | 978-1-6654-8534-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/MILCOM55135.2022.10017581

***Abstract*—Nowadays, Internet-of-Things (IoT) have shown great potential to improve the performance of different applications. However, the advances of IoT also make the security issue become one of the biggest challenges for these applications. Researchers in the past have shown that Symmetric key cryptography is generally considered infeasible and public key cryptography, at times, fails to provide sufficient security and integrity to data. In contrast to this prejudice, our paper presents a novel approach that establishes security to data through encryption techniques like RSA. Moreover, it identifies the shortest path to route messages from the source to the destination and ensures that packets are delivered safely even when intermediate nodes are attacked by identifying Alternate paths between the source and the destination. We implement and evaluate our design by using J2ME, and the evaluation results show the effectiveness of our design.**

***Index Terms*–Wireless Sensor Networks, security, individual sensors, Symmetric Key cryptography, Alternate Paths**

I. Introduction

The use of Internet-of-Things (IoTs) in numerous sensitive areas like health-care, military, habitat monitoring, etc., has resulted in the need to safeguard and protect the integrity of data. For example, in battlefield applications- the location of a soldier must be confidential while broadcasting queries. These queries should be transmitted to the destination through encryptions via trusted intermediate nodes. Similarly, in habitat monitoring application scenarios, like the Great Duck Island or Save-the-panda applications, plenty of sensor nodes are deployed to observe the vast habitat of ducks and pandas. In scenarios like these, adversaries can try to capture data regarding pandas or ducks by backtracking the route traversed by the data to reach the destination from the source sensor nodes.

To prevent the interference of adversaries in such circumstances, techniques like identity, route and location privacy mechanisms must be enforced. With respect to these application scenarios, network level privacy has often been categorized into four categories: a) Sender node identity privacy: no intermediate node can get any information about who is sending the packets except the source, its immediate neighbors and the destination b) Sender node location privacy: no intermediate node can have any information about the location (in terms of physical distance or number of hops) about the sender node except the source, its immediate neighbors and the destination. c) Route privacy: no node can predict the information about the complete path (from source to destination). Also, a mobile adversary cannot trace back to the source node either from the contents and/or directional information of the captured packet(s) d) Data packet privacy: no node can see the information inside a payload of the data packet except the source and the destination.

Currently, researchers mainly use cryptographic approaches to ensure the security of the network [1]–[3]. However, due to the complexity of designing and analyzing cryptographic procedures, security remains a pertinent concern. For example, mIDS [2] proposes an intrusion detection system that monitors network activity to detect malicious behavior. SQSR [3] presents a secure routing protocol which utilizes a combine cost function that considers both the desired performance and security requirements for route discovery.

Different from above approaches, the main objective of this work is to simulate and analyze protocols associated with Alternate path routing (APR) which can be used to achieve efficient routing in communications networks. We describe the link state routing approach to APR in connection-oriented networks, and evaluate their performance using simulations. The source node transmits data to a random neighbor node closer to the destination, alternate paths can be computed at intermediate nodes based on a trust value associated with each node. We present initial simulation results to show the working mechanism of APR using the J2ME Wireless Toolkit. Existing privacy schemes of wireless sensor networks only provide partial network level privacy. In this project, we propose a full network level privacy solution that addresses this problem in APR protocols. This solution comprises of Identity, Route, Location (IRL) and data privacy mechanisms that collectively provide protection against privacy disclosure attacks such as eavesdropping and hop-by-hop trace back attacks. Overall, we aim to develop a system that a) Simulates Link State Routing Protocol using J2ME Wireless Toolkit b) Simulates our proposed basic and secured routing protocols c) Provides full network security with the help of the IRL algorithm d) Compares the performances of basic and secure routing e)

Provides a user friendly and well designed user interface.

The contribution of this paper is summarized as follows:

• We design secure routing and encryption algorithms that can provide integrity to data being transmitted across the IoT network.

• To prove the effectiveness of our design, we implement our algorithms by using J2ME Wireless Toolkit and experiment results show that our algorithms can improve the integrity for the data being transmitted through the IoT network.

II. Related Work

Researchers have conducted extensive research on IoT, including Communication [4]–[9], localization [10]–[12], data forwarding [13]–[16], energy optimization [17]–[25], security [26]–[30], and utilizing sensor networks for applications in smart grids [31]–[38], drones [39]–[41], smart sensing [42] and smart health [43]–[46]. Most related work can be divided into the following categories: routing, data forwarding, and security.

1) Routing: Several techniques have been used to improve routing efficiency and reliability in wireless networks. WOSPF [47] takes a traffic engineering approach that minimizes the difference between the maximum and minimum link utilizations across a network. ABLQ [48] evaluates the link quality and availablity of mobile nodes to perform energy effecient routing. DDS-MWST [49] and [50] reduce routing overhead by using a 3-hop message relay algorithm to construct connected dominating sets. CRF [51] and CF [52] reduce flooding delay within ZigBee networks by using a collective acknoledgement which can be applied in heterogenous WiFi and ZigBee networks to increase throughput.

2) Data Forwarding: The overcrowded Industrial, Scientific, and Medical (ISM) band has given rise to significant work in optimizing cross-technology data forwarding. Researchers have leveraged unique features of WiFi, Bluetooth, and ZigBee to enable concurrent Cross-Technology Communication (CTC) [53]–[58]. For example, B^2W^2 [54] uses the WiFi CSI to demodulate the BLE data while PIC [53] uses customized devices to support cross-technology communication among IoT devices with different physical layers. However, since the duration of a wireless packet is in the range of milliseconds, the throughput and communication range of packet-level CTC are limited. To overcome this challenge, the most recent and widely applied work – WEBee [59] first proposes the physical layer emulation technique to directly conduct communication from WiFi to ZigBee without modification of hardwares. Other research, such as iCore [60] and ECT [61] exploit CTC strategies to employ dynamic data forwarding techniques which reduce data collision and delivery delay, respectively. Since many IoT devices have limited battery life, a CTC backscattering technique has been proposed by the authors of [62] which reflects WiFi packets for ZigBee demodulation to reduce power consumption.

3) Security: Due to the complexity of designing and analyzing cryptographic procedures, security remains a pertinent concern. Researchers have identified tracking [1], [63] and Denial-of-Service (DoS) [64] attacks against sensor nodes in wireless networks. However, other work has made significant contributions in detecting and preventing privacy attacks. mIDS [2] proposes an intrusion detection system that monitors network activity to detect malicious behavior. SQSR [3] presents a secure routing protocol which utilizes a combine cost function that considers both the desired performance and security requirements for route discovery.

Unlike previous research, this paper introduces secure routing and encryption algorithms that collaboratively provide identity, route, and location privacy in IoT networks.

III. Proposed System

We propose a system that incorporates the Identity, Route and Location (IRL) privacy algorithm to ensure integrity of data being transmitted across the network. In brief, our system carries out the following functions: a) A new Identity, Route and Location (IRL) privacy algorithm is proposed that ensures the anonymity of source nodes identity and location b) It also assures that the packets will reach their destination by passing through only trusted intermediate nodes. c) .The anonymity of the current intermediate node is maintained d) A new data privacy mechanism is proposed, which provides payload encryption and decryption e) The intermediate nodes are oblivious to the previous path taken

IV. Module Decomposition

1) *Secure Routing:* Implements IRL-1 and IRL-2 privacy algorithms on top of our Link State Routing Protocol.
2) *Basic Routing:* Implements Link State Routing Protocol to send data from source to destination.
3) *Private Node:* Node closest to the centroid in each cell and the only node in a cell to access the path array
4) *Gateway Node:* Node farthest from the centroid
5) *Centroid Calculation:* Module to calculate the centroid of each cell
6) *Sender:* J2ME emulation of a sending device
7) *Receiver:* J2ME emulator of a receiving device
8) *Client:* Nodes which connect to form the topology in the Initial simulation. Each client has its own socket to connect to other clients
9) *Network Manager:* Runs a server Socket, deploys the Applet and provides a command line interface to the user

V. Secure Routing and Encryption Algorithms

In our project, we designed two vital algorithms that provide integrity to data being transmitted across the sensor network. The first algorithm describes the procedure used to identify the neighboring nodes and fill entries into the routing table. A routing table maintains the distance of a specific node from the other nodes in the network. The second algorithm provides encryption to data and the path taken.

A. Secure Routing Algorithm

This algorithm divides the entire network topology into hexagonal cells with a collection of nodes within them. The

```
range <- 50
n <- (2-6)

for all nodes in network do
      Place n nodes within the boundaries of each cell
      Calculate centroid()
end for

for each node in the cell do
      Calculate Euclidian distance b/w node (i) and centroid
      Private <- node closest to centroid
      Gateway[] <- all nodes in cell != to centroid
end for

while destination not reached do
      id <- name of current node
      neighbour[] <- null
      neighbourcount <- 0

      trustValue <- 0
      neighbour1[] <- null
      neighbour2[] <- null

      Fill routing table by calculating distance from the current
      node to every node in the network

      for node in table with (distance<=range) do
            add node to neighbour[]
            neighbourcount <- neighbourcount + 1
      end for

      neighbour1[] <- split(neighbours[] closer to dest. than current node)
      neighbour2[] <- neighbour1[]

      while trustValue == 0 and length of neighbours > 0 do
            nextNode <- random index of neighbour1
            if neighbour1[nextNode].trustValue >= 50 do
                  trustValue <- neighbour1[nextNode].trustValue
                  Encryption Algorithm <- id
                  hop to neighbour[nextNode]
                  break
            end if
            remove neighbour1[nextNode] from neighbour1[]
      end while

      if trustValue == 0 do
            Encryption Algorithm <- id
            hop to node with highest trust value in neighbour2[]
      end if
end while

Encryption Algorithm <- destination
Respond with ACK along same path taken
if ACK received do
      update trust value of node that sent ACK
else
      wait until time out and update trust value
end if
```

Fig. 1: Routing algorithm with trusted random route generation

number of nodes within each cluster can vary from two to six. Once the user has selected the number of nodes to be placed inside the cells, the centroid for each cell is calculated. Next, the algorithm identifies the private nodes and gateway nodes. For this, the distance of the nodes from the centroid are considered. For instance, the node that lies closest to the centroid is labeled as the private node and all others are labeled as gateway nodes (i.e., public nodes). This step will iterate until all nodes in the network have been divided into a hexagonal cell of n nodes. The current node is identified as the node that currently holds the data packet that needs to be routed. The neighbors of this node are all initially null. This is represented by the neighbor array. The number of neighboring nodes is initially zero and increments only once neighbors are encountered. Each node maintains a routing table of it own. The entries of the routing table represent that distance of a node from all the other nodes in the topology. Neighbor nodes are defined as those that are one hope away from the current node. Each time a neighbor is encountered, the neighbor array is updated and the neighbor count variable is incremented.

Motivated by the work done by Prabhat Kumar et al. [65], each node will chose a random next hop node closer to the destination for each packet. This is to ensure that the path between any given source and destination is not predictable by the adversary. Furthermore, to ensure that packets are only routed through trusted nodes, every node in the network maintains a trust value for each of its neighbors. When a new node is introduced to the network, its neighbors should set its initial trust value to 100. Using the equation proposed by R.A. Shailkh et al. [66], the trust value of each node is then updated by its neighbors whenever a packet is routed though this node. This is accomplished by sending an ACK response back along the same path the was taken to route the packet. If an ACK is received, then the current node will increase the trust value of the node that sent the ACK. On the contrary, if no ACK is received after a certain time out period, each node should decrease the trust value of the neighbor that was forwarded the packet.

B. Encryption Algorithm

```
id <- name of current node
dest <- name of destination node

if id == source do
      path[] <- null
      intermediatePath[] <- null
      message <- message sent by source to destination

      data <- Encrypt message under the public key of the destination
      Add id to path[]
      intermediateDest <- next hop private nodes name

      Return data, path[] and dest encrypted under next hop private
      nodes public key, intermediatePath[] and intermediateDest
      encrypted under next hop nodes public key

else if id != destination and id != private node do
      decrypt intermediatePath[] and intermediateDest
      Add id to intermediatePath[]

      Return data, path[] and dest encrypted under next hop private
      nodes public key, intermediatePath[] and intermediateDest
      encrypted under next hop nodes public key

else if id != destination and id == private node do
      decrypt intermediatePath[] and intermediateDest
      decrypt path[] and dest

      Add intermediatePath[] to path[]
      Add id to path[]
      intermediatePath <- null
      intermediateDest <- next hop private nodes name

      Return data, path[] and dest encrypted under next hop private
      nodes public key, intermediatePath[] and intermediateDest
      encrypted under next hop nodes public key

else if id == dest do
      decrypt intermediatePath[]
      decrypt path[]

      Add intermediatePath[] to path[]
      nextHop <- last node in path[]

      pathChain <- null
      for node in path[] do
            Add current node to pathChain
            pathChain <- encrypt pathChain with public key of
                         current node
      end for

      Return nextHop and pathChain
end if
```

Fig. 2: Encryption Algorithm that provides data and path confidentiality

This algorithm provides encryption for the data contained in each packet and the contents of the path array. Confidentiality of data is provided by encrypting the message being sent by the source to the destination using the public key of the destination. Since only the destination knows the associated private key, the data will be kept secret. To prevent external entities from accessing the path traversed or the nodes present on a specific path several different techniques are used for each

type of node (i.e., source, private, public, and destination). The source node will start by generating two path arrays; one that can be used by private nodes, and another that can be used by public nodes. The public nodes will add their encrypted identity to the intermediate path array. This is done to prevent public nodes from accessing the contents of the full path array. The private nodes are then responsible for appending the contents of the intermediate path array to the full path array as well as adding their encrypted identity to the full path array. To avoid leaking the destination of each packet to the public nodes, the private nodes will also set an intermediate destination equal to the identity of the next hop private node. The intermediate path array and full path array are also encrypted under the next hop nodes public key and the next hop private nodes public key, respectively. This is to ensure that an adversary eavesdropping on the communication cannot determine any information about hop count. When the destination receives the data packet it will decrypt the contents of the path array and use a block chain approach to generate an ACK response that can be sent along the same path traversed. This is accomplished by first encrypting the identity of the source node under the sources public key. Then, the identity of the node that received the original packet from the source and the encryption of the source identity is encrypted under this nodes public key. This is repeated for each identity within the path array. Therefore, when the ACK response is sent, each node only know the identity of the next hop node which ensures that no information is leaked about the path taken.

VI. Implementation

We implemented our system model in two separate stages. This has been described in the following sections:

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus, it can be considered the most critical stage in achieving a successful new system and in giving the user confidence that the new system will work and be effective. The implementation stage involves careful planning, investigation of the existing system and its constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

1) Phase 1: In phase 1, we implement the Network model. We created the network Manager that provides the user interface for alternate path routing. It runs server sockets that listen to incoming client requests that desire to connect to the existing network. If any intermediate node fails, then an alternate path is identified and the packet is re-routed along the new path, We have used the J2ME Wireless Toolkit to emulate the Sender and the Receiver nodes and designed the interface for the user to enter the message (text) that must be delivered. Once, the user clicks send, the message is delivered over a randomized path to the destination. This message is encrypted using Quasigroup Encryption techniques that protect the integrity of data. The entire simulation has been implemented using JAVA. The GUIs have been designed using Java Swing and batch files have been created to enable one-click applet launching. The topology is crated using socket programming where the client and server sockets are installed on every node. The other intermediate nodes can be specified by entering in/out text on the input line. The initial state of our topology contains five nodes.

2) Phase 2: In the second phase, we simulated our link state routing protocol by integrating IRL security. We used JAVA Swings to create the GUI. A network topology was constructed, which divided the complete topology into hexagonal cells (clusters). Each cell can have a maximum of six nodes and a minimum of two nodes. We provided two input dialogs- a) to select the number of nodes and b) to select the encryption algorithm for IRL. After the cell and node were placed, the user was prompted to initialize the nodes by clicking the Initialize Nodes button. As an action, the centroid for each cell was created and the private and gateway nodes were determined. The node which lied closest to the centroid was labelled as the private node and the one farthest was labelled the gateway node. Next, the user was required to select the source and destination nodes by clicking on them. If basic routing was selected, then the data would be routed without any security implementation. A routing table was created with the distance of each node from every other node in the network. The source node selected only those nodes which were at a distance of 50 units or less from it (Transmission range). It forwarded the packet to a random neighbor which is closer to the destination and adds it to the path array. The same process was repeated for the next node until the destination was reached. In secure routing, the path array can only be accessed by the private nodes and its contents are encrypted. Hence the identity of the source, route taken, and the location of the current node are private. The user can compare the path, number of hops, and the time taken for both the routing protocols.

VII. Evaluation

Our evaluation procedure was carried out on each module that had been designed separately. We initially evaluated the components of our User Interfaces to ensure all the required components to help the user form the topology of nodes were present. We evaluated the behaviour of our network manager with different combinations of source and destination nodes to check if the components worked appropriately. Next, we tested if each node had a server socket and client socket installed to ensure that all nodes were capable of behaving like either the client or the source. Our primary objective was to test the behaviour of our program within a topology consisting of five nodes.

VIII. Conclusion

With the constant increase in the applications of IoT networks, the issues associated with achieving energy efficient communication and secured routing to prevent loss of data due to attacks becomes much more important. If necessary measures to defend such attacks are not designed, it becomes highly difficult to assure safe transmission of messages. Also, it is impossible to completely trust the results reported from sensor networks deployed outside of controlled environments. In our project, our primary focus was on providing security to

the sensitive information stored in the routing table of every node in the network. As discussed earlier, the routing table contains the distance of a node from all the other nodes in the topology and attacking the path array provides the attacker the opportunity to access all nodes in the path traversed. We successfully designed our own encryption algorithm which works in tandem with RSA encryption. This helps in encrypting the data present in the path array which can only be accessed by the private nodes in the topology. We executed our algorithms on a network consisting of five nodes in the initial stage and successfully carried out reliable and efficient transfer of messages from the source to destination.

IX. Future Work

With the advances in wireless sensor technology and the emergence of microelectromechanical systems (MEMS), there have been large scale developments in wireless communications. Tactical IoT networks have become one of the most interesting areas of research in the past few years. Future developments in sensor nodes must aim to improve the efficiency in transmission and design cost effective devices, so that they can be used in applications like underwater acoustic sensor systems, sensing based cyber physical systems, time critical applications, cognitive sensing and spectrum management, and security and privacy management.

As the future work, we plan to leverage artificial intelligence (AI) to help future networks operators to provide the best predictive and proactive elements and overcome all the challenges that the current network systems are facing (i.e., congestion, security, and low throughput, etc.) because of the use of some manual and conventional approaches to deal with the data.

We also plan to explore to apply our approach to edge computing. To be more specific, edge computing normally require the nodes to dynamically allocate tasks to different nodes. Since the allocated tasks normally contains the data that is user-specific, it is really important to guarantee the security and privacy of edge computing. By using the proposed algorithm in this paper, we propose to design secure routing protocols for edge computing to improve the security-level of potential applications.

X. Acknowledgements

This project was supported by NSF grants CNS-1652669, CNS-2127908, and CNS-2127881.

References

[1] Z. Zhong, T. Zhu, D. Wang, and T. He, "Tracking with unreliable node sequences," in *IEEE INFOCOM 2009*, pp. 1215–1223, 2009.

[2] P. yi, T. Zhu, J. Ma, and Y. Wu, "An intrusion prevention mechanism in mobile ad hoc networks," *Ad-Hoc and Sensor Wireless Networks*, vol. 17, pp. 269–292, 01 2013.

[3] T. Zhu and M. Yu, "Nis02-4: A secure quality of service routing protocol for wireless ad hoc networks," in *IEEE Globecom 2006*, pp. 1–6, 2006.

[4] X. Liu, Z. Chi, W. Wang, Y. Yao, P. Hao, and T. Zhu, "Verification and redesign of {OFDM} backscatter," in *18th USENIX Symposium on Networked Systems Design and Implementation (NSDI 21)*, 2021.

[5] Z. Chi, X. Liu, W. Wang, Y. Yao, and T. Zhu, "Leveraging ambient lte traffic for ubiquitous passive communication," in *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*, 2020.

[6] X. Liu, Z. Chi, W. Wang, Y. Yao, and T. Zhu, "{VMscatter}: A versatile {MIMO} backscatter," in *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20)*, 2020.

[7] Z. Chi, Y. Li, X. Liu, W. Wang, Y. Yao, T. Zhu, and Y. Zhang, "Countering cross-technology jamming attack," in *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, 2020.

[8] Z. Chi, Y. Yao, T. Xie, X. Liu, Z. Huang, W. Wang, and T. Zhu, "Ear: Exploiting uncontrollable ambient rf signals in heterogeneous networks for gesture recognition," in *Proceedings of the 16th ACM conference on embedded networked sensor systems*, 2018.

[9] Z. Chi, Y. Yao, T. Xie, Z. Huang, M. Hammond, and T. Zhu, "Harmony: Exploiting coarse-grained received signal strength from iot devices for human activity recognition," in *2016 IEEE 24th International Conference on Network Protocols (ICNP)*, IEEE, 2016.

[10] J. Jun, Y. Gu, L. Cheng, B. Lu, J. Sun, T. Zhu, and J. Niu, "Social-loc: Improving indoor localization with social sensing," in *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, SenSys '13, (New York, NY, USA), Association for Computing Machinery, 2013.

[11] P. Yi, M. Yu, Z. Zhou, W. Xu, Q. Zhang, and T. Zhu, "A three-dimensional wireless indoor localization system," in *Journal of Electrical and Computer Engineering*, no. 149016, 2014.

[12] J. Jun, L. He, Y. Gu, W. Jiang, G. Kushwaha, V. A., L. Cheng, C. Liu, and T. Zhu, "Low-overhead wifi fingerprinting," *IEEE Transactions on Mobile Computing*, 2018.

[13] J. Jun, L. Cheng, L. He, Y. Gu, and T. Zhu, "Exploiting sender-based link correlation in wireless sensor networks," in *2014 IEEE 22nd International Conference on Network Protocols*, pp. 445–455, 2014.

[14] Y. Gu, L. He, T. Zhu, and T. He, "Achieving energy-synchronized communication in energy-harvesting wireless sensor networks," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 13, 01 2014.

[15] T. Zhu and D. Towsley, "E2r: Energy efficient routing for multi-hop green wireless networks," in *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2011.

[16] W. Wang, X. Liu, Y. Yao, and T. Zhu, "Exploiting wifi ap for simultaneous data dissemination among wifi and zigbee devices," in *2021 IEEE 29th International Conference on Network Protocols (ICNP)*, IEEE, 2021.

[17] Q. Zhang, T. Zhu, Y. Ping, and Y. Gu, "Cooperative data reduction in wireless sensor network," in *2012 IEEE Global Communications Conference (GLOBECOM)*, pp. 628–633, 2012.

[18] T. Zhu, A. Mohaisen, Yi Ping, and D. Towsley, "Deos: Dynamic energy-oriented scheduling for sustainable wireless sensor networks," in *2012 Proceedings IEEE INFOCOM*, pp. 2363–2371, 2012.

[19] T. Zhu, Z. Zhong, Y. Gu, T. He, and Z.-L. Zhang, "Leakage-aware energy synchronization for wireless sensor networks," in *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services*, MobiSys '09, (New York, NY, USA), p. 319–332, Association for Computing Machinery, 2009.

[20] L. He, L. Kong, Y. Gu, J. Pan, and T. Zhu, "Evaluating the on-demand mobile charging in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 14, no. 9, pp. 1861–1875, 2015.

[21] C. Zhou and T. Zhu, "Highly spatial reusable mac for wireless sensor networks," in *2007 International Conference on Wireless Communications, Networking and Mobile Computing*, 2007.

[22] T. Zhu, Sheng Xiao, Yi Ping, D. Towsley, and Weibo Gong, "A secure energy routing mechanism for sharing renewable energy in smart microgrid," in *2011 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 143–148, 2011.

[23] T. Zhu, Y. Gu, T. He, and Z.-L. Zhang, "Eshare: A capacitor-driven energy storage and sharing network for long-term operation," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys '10, (New York, NY, USA), p. 239–252, Association for Computing Machinery, 2010.

[24] Y. Hu, P. Yi, Y. Sui, Z. Zhang, Y. Yao, W. Wang, and T. Zhu, "Dispatching and distributing energy in energy internet under energy dilemma," in *2018 Third International Conference on Security of Smart Cities, Industrial Control System and Communications (SSIC)*, IEEE, 2018.

[25] J. Xu, P. Yi, T. Xie, W. Wang, X. Liu, and T. Zhu, "Charge station placement in electric vehicle energy distribution network," in *2017 IEEE International Conference on Communications (ICC)*, IEEE, 2017.

[26] P. Yi, T. Zhu, Q. Zhang, Y. Wu, and J. Li, "Green firewall: An energy-efficient intrusion prevention mechanism in wireless sensor network,"

in *2012 IEEE Global Communications Conference (GLOBECOM)*, pp. 3037–3042, 2012.

[27] J. Kulkarni, K. Nair, A. Pappu, S. Gadre, G. Gore, and J. Joshi, "Using on-chip cryptographic units for security in wireless sensor networks," in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2017.

[28] D. Jiang, Z. Xu, P. Zhang, and T. Zhu, "A transform domain-based anomaly detection approach to network-wide traffic," *Journal of Network and Computer Applications*, vol. 40, pp. 292 – 306, 2014.

[29] W. Wang, Y. Yao, X. Liu, X. Li, P. Hao, and T. Zhu, "I can see the light: Attacks on autonomous vehicles using invisible lights," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1930–1944, 2021.

[30] S. Gu, P. Yi, T. Zhu, Y. Yao, and W. Wang, "Detecting adversarial examples in deep neural networks using normalizing filters," *UMBC Student Collection*, 2019.

[31] Z. Huang and T. Zhu, "Real-time data and energy management in microgrids," in *2016 IEEE Real-Time Systems Symposium (RTSS)*, pp. 79–88, 2016.

[32] Z. Huang and T. Zhu, "Leveraging multi-granularity energy data for accurate energy demand forecast in smart grids," in *2016 IEEE International Conference on Big Data (Big Data)*, pp. 1182–1191, 2016.

[33] Z. Huang, T. Zhu, D. Irwin, A. Mishra, D. Menasche, and P. Shenoy, "Minimizing transmission loss in smart microgrids by sharing renewable energy," *ACM Trans. Cyber-Phys. Syst.*, 2016.

[34] A. Mishra, D. Irwin, P. Shenoy, J. Kurose, and T. Zhu, "Smartcharge: Cutting the electricity bill in smart homes with energy storage," in *Proceedings of the 3rd International Conference on Future Energy Systems: Where Energy, Computing and Communication Meet*, e-Energy '12, (New York, NY, USA), Association for Computing Machinery, 2012.

[35] T. Zhu, Z. Huang, A. Sharma, J. Su, D. Irwin, A. Mishra, D. Menasche, and P. Shenoy, "Sharing renewable energy in smart microgrids," in *2013 ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS)*, pp. 219–228, 2013.

[36] A. Mishra, D. Irwin, P. Shenoy, and T. Zhu, "Scaling distributed energy storage for grid peak reduction," in *Proceedings of the Fourth International Conference on Future Energy Systems*, e-Energy '13, 2013.

[37] Z. Huang, T. Zhu, H. Lu, and W. Gao, "Accurate power quality monitoring in microgrids," in *2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pp. 1–6, IEEE, 2016.

[38] Z. Huang, T. Zhu, Y. Gu, and Y. Li, "Shepherd: sharing energy for privacy preserving in hybrid ac-dc microgrids," in *Proceedings of the Seventh International Conference on Future Energy Systems*, pp. 1–10, 2016.

[39] Y. Pan, S. Li, X. Zhang, J. Liu, Z. Huang, and T. Zhu, "Directional monitoring of multiple moving targets by multiple unmanned aerial vehicles," in *GLOBECOM 2017-2017 IEEE Global Communications Conference*, IEEE, 2017.

[40] Y. Pan, S. Li, J. L. Chang, Y. Yan, S. Xu, Y. An, and T. Zhu, "An unmanned aerial vehicle navigation mechanism with preserving privacy," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, IEEE, 2019.

[41] Y. Pan, B. Bhargava, Z. Ning, N. Slavov, S. Li, J. Liu, S. Xu, C. Li, and T. Zhu, "Safe and efficient uav navigation near an airport," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, IEEE, 2019.

[42] E. Miller, Z. Li, H. Mentis, A. Park, T. Zhu, and N. Banerjee, "Radsense: Enabling one hand and no hands interaction for sterile manipulation of medical images using doppler radar," *Smart Health*, 2020.

[43] J. Gao, P. Yi, Z. Chi, and T. Zhu, "Enhanced wearable medical systems for effective blood glucose control," in *2016 IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*, pp. 199–208, 2016.

[44] D. Jiang, Z. Yuan, P. Zhang, L. Miao, and T. Zhu, "A traffic anomaly detection approach in communication networks for applications of multimedia medical devices," *Multimedia Tools Appl.*, 2016.

[45] Y. Yao, Z. Ning, Q. Zhang, and T. Zhu, "Paris: passive and continuous fetal heart monitoring system," *Smart Health*, 2020.

[46] Y. Liu, Z. Xia, P. Yi, Y. Yao, T. Xie, W. Wang, and T. Zhu, "Genpass: A general deep learning model for password guessing with pcfg rules and adversarial generation," in *2018 IEEE International Conference on Communications (ICC)*, IEEE, 2018.

[47] A. Mishra, A. Sahoo, B. Dalvi, and T. Zhu, "Wospf: A traffic engineering solution for ospf networks," in *2016 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–7, 2016.

[48] A. Malvankar, M. Yu, and T. Zhu, "An availability-based link qos routing for mobile ad hoc networks," in *2006 IEEE Sarnoff Symposium*, 2006.

[49] S. Ren, P. Yi, D. Hong, Y. Wu, and T. Zhu, "Distributed construction of connected dominating sets optimized by minimum-weight spanning tree in wireless ad-hoc sensor networks," in *2014 IEEE 17th International Conference on Computational Science and Engineering*, pp. 901–908, 2014.

[50] S. Ren, P. Yi, T. Zhu, Y. Wu, and J. Li, "A 3-hop message relay algorithm for connected dominating sets in wireless ad-hoc sensor networks," in *2014 IEEE/CIC International Conference on Communications in China (ICCC)*, pp. 829–834, 2014.

[51] W. Wang, X. Liu, Y. Yao, Y. Pan, Z. Chi, and T. Zhu, "Crf: Coexistent routing and flooding using wifi packets in heterogeneous iot networks," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pp. 19–27, 2019.

[52] T. Zhu, Z. Zhong, T. He, and Z.-L. Zhang, "Exploring link correlation for efficient flooding in wireless sensor networks," in *Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation*, NSDI'10, (USA), p. 4, USENIX Association, 2010.

[53] Z. Chi, Y. Li, X. Liu, Y. Yao, Y. Zhang, and T. Zhu, "Parallel inclusive communication for connecting heterogeneous iot devices at the edge," in *Proceedings of the 17th Conference on Embedded Networked Sensor Systems*, SenSys '19, (New York, NY, USA), p. 205–218, Association for Computing Machinery, 2019.

[54] Z. Chi, Y. Li, H. Sun, Y. Yao, Z. Lu, and T. Zhu, "B2w2: N-way concurrent communication for iot devices," in *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, SenSys '16, (New York, NY, USA), p. 245–258, Association for Computing Machinery, 2016.

[55] Z. Chi, Y. Li, Z. Huang, H. Sun, and T. Zhu, "Simultaneous bi-directional communications and data forwarding using a single zigbee data stream," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pp. 577–585, 2019.

[56] Z. Chi, Z. Huang, Y. Yao, T. Xie, H. Sun, and T. Zhu, "Emf: Embedding multiple flows of information in existing traffic for concurrent communication among heterogeneous iot devices," in *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, 2017.

[57] Z. Chi, Y. Li, Y. Yao, and T. Zhu, "Pmc: Parallel multi-protocol communication to heterogeneous iot radios within a single wifi channel," in *2017 IEEE 25th International Conference on Network Protocols (ICNP)*, pp. 1–10, 2017.

[58] Y. Li, Z. Chi, X. Liu, and T. Zhu, "Chiron: Concurrent high throughput communication for iot devices," in *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '18, (New York, NY, USA), p. 204–216, Association for Computing Machinery, 2018.

[59] Z. Li and T. He, "Webee: Physical-layer cross-technology communication via emulation," in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, MobiCom '17, pp. 2–14, 2017.

[60] L. Cheng, Y. Gu, J. Niu, T. Zhu, C. Liu, Q. Zhang, and T. Hel, "Taming collisions for delay reduction in low-duty-cycle wireless sensor networks," in *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pp. 1–9, 2016.

[61] W. Wang, T. Xie, X. Liu, and T. Zhu, "Ect: Exploiting cross-technology concurrent transmission for reducing packet delivery delay in iot networks," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, pp. 369–377, 2018.

[62] Y. Li, Z. Chi, X. Liu, and T. Zhu, "Passive-zigbee: Enabling zigbee communication in iot networks with 1000x+ less power consumption," in *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*, SenSys '18, 2018.

[63] Y. Li and T. Zhu, "Gait-based wi-fi signatures for privacy-preserving," in *Proceedings of the 11th ACM on asia conference on computer and communications security*, pp. 571–582, 2016.

[64] P. Yi, T. Zhu, Q. Zhang, Y. Wu, and J. Li, "A denial of service attack in advanced metering infrastructure network," in *2014 IEEE International Conference on Communications (ICC)*, pp. 1029–1034, 2014.

[65] P. Kumar, J. P. Singh, P. Vishnoi, and M. P. Singh, "Source location privacy using multiple-phantom nodes in wsn," in *TENCON 2015 - 2015 IEEE Region 10 Conference*, pp. 1–6, 2015.

[66] R. A. Shaikh, H. Jameel, B. J. d'Auriol, S. Lee, Y. Song, and H. Lee, "Network level privacy for wireless sensor networks," in *2008 The Fourth International Conference on Information Assurance and Security*, pp. 261–266, 2008.