*2019 IEEE 18th International Symposium on Network Computing and Applications (NCA)*

# Digital Signature Based Countermeasure Against Puppet Attack in the Internet of Things

Cong Pu Logan Carpenter
Weisberg Division of Computer Science
Marshall University, Huntington, WV 25755, USA
Email: {puc, carpenter190}@marshall.edu

***Abstract*—In order to achieve the goal of smooth interaction and communication, a novel distance vector and source routing protocol, also officially referred to as RPL, has been proposed for IPv6-based Low Power and Lossy Networks (LLNs) which serve as a major component in the architecture of Internet of Things (IoT). Unfortunately, IoT devices are often equipped with limited energy and extremely constrained with regard to the capabilities of computing and communicating, thus, IoT and its applications are seriously vulnerable to diverse cyber attacks, and investigating possible attacks against IoT-related routing protocol is a top priority to enhance the security of IoT systems in the future. In this paper, we propose a digital signature based countermeasure along with other techniques to defend against puppet attack in LLNs running with RPL. The experimental results indicate that the proposed digital signature based countermeasure can not only reduce the performance impact of puppet attack significantly, but also can accurately detect and effectively mitigate puppet attack.**

***Index Terms*—Puppet Attack, Signature Based Countermeasure, RPL, Low Power and Lossy Networks, Internet of Things**

## I. Introduction

Internet of Things (IoT) interconnects a large number of heterogeneous smart devices and serves as a bridge between devices and the Internet to build a comprehensive and ubiquitous communication environment. As a major building block of IoT architecture, IPv6-based Low Power and Lossy Networks (LLNs), where a large collection of highly resource-constrained devices are interconnected directly or indirectly through unreliable wireless links, have attracted attention from academia, industry, and standard organization. In order to enable resource-constrained devices to efficiently communicate and deliver data through multihop relays, a significant amount of research effort has led to the standardization of the IPv6-based routing protocol for Low Power and Lossy Networks, also officially referred to as RPL [1].

Unfortunately, IoT devices are characterized by constraints in terms of endowed resources and underlying communication technologies, thus, LLNs running with RPL are undoubtedly vulnerable to various Denial-of-Service (DoS) attacks [2]. Although the Internet Engineering Task Force Working Group has presented the potential threats and attacks and suggested naive countermeasures in [3], this still leaves RPL open to new cyber attack wherein an attacker can update the information of packet header and send the manipulated packets to attack the network. In this paper, we investigate the puppet attack targeting on the vulnerabilities of RPL routing protocol in LLNs. In puppet attack, an attacker turns legitimate nodes into puppet nodes by sending them counterfeit data packets piggybacked with manipulated source route. When the puppet nodes receive counterfeit data packets with invalid source route, they have to discard all counterfeit data packets because of invalid source route, and reply a large number of Error messages back to the DODAG root to RPL. Consequently, all data packets are dropped by legitimate nodes and an excessive number of Error messages exhaust the network communication bandwidth and node energy, finally resulting in a denial of service. The puppet attack is more concealment and deception than selective forwarding attack or flooding attack because the adversary can send attack packets to legitimate nodes and turn them into puppet nodes to actually attack the network.

In order to defend against puppet attack, we propose a digital signature based countermeasure along with other techniques, also referred to as *SIG*. The basic idea of the SIG is that the DODAG root processes the source route with a secure hash algorithm to generate a message digest, uses its private key to encrypt the message digest to form a digital signature which can only be decrypted using its public key, and then attaches the digital signature in the data packet. When an intermediate node along the forwarding path receives a data packet but cannot verify the piggybacked digital signature and source route by using the public key of the DODAG root, it prosecutes the packet forwarding node for performing puppet attack. We develop a customized discrete event-driven simulation framework by using OMNeT++ [4] and evaluate its performance through extensive simulation experiments in terms of detection rate, false detection rate, and energy consumption. The simulation results show that the proposed countermeasure can not only reduce the performance impact of puppet attack significantly, but also can accurately detect and effectively mitigate puppet attack.

## II. Related Work

Previous work from author have concentrated on investigating a variety of attacks and proposing countermeasures in various networks, such as camouflage-based detection against crafty packet drop attack in Energy Harvesting Motivated Networks (EHNets) [5], [6], acknowledgment-based countermeasure against stealthy collision attack in EHNets [7], single checkpoint-assisted approach against selective forward-

978-1-7281-2522-0/19/$31.00 ©2019 IEEE

ing attack in Wireless Sensor Networks [8], DSR-based bait detection scheme against routing attack in Mobile Ad Hoc Networks [9], jamming-resilient multipath routing protocol against jamming attack in Flying Ad Hoc Networks [10], and self-adjusting share-based countermeasure against interest flooding attack in Named Data Networking [11].

Along with the emergence of IoT, over the past few years, investigating possible attacks against IoT-related routing protocol is a top priority to enhance the security of IoT systems. A significant amount of research effort from author has been devoted to investigate various attacks in LLNs running with RPL, such as forwarding misbehaviors [12], DAO inconsistency attack [13], suppression attack [14], [15], spam DIS attack [16], hatchetman attack [17], and energy depletion attack [18], [19]. Many other researchers also make a large contribution to RPL security. In [20], a mechanism named SecRPL that restricts the number of forwarded DAO messages by a parent node is proposed to address a DAO insider attack in RPL, where an adversary repeatedly and judiciously replays eavesdropped DAO messages from legitimate nodes to trigger the transmission of multiple DAO messages from intermediate parent nodes. The SecRPL can either restrict the entire number of forwarded DAO messages regardless where they are from, or restrict the number of forwarded DAO messages according to destination. The [21] first modifies the RPL routing protocol with the security techniques of a nonce identity value, timestamp, and network whitelist, and then proposes a trust-based security architecture for mobile IoT network. In [22], two defense mechanisms, called Elimination and Shield respectively, are proposed to mitigate RPL version number attack in LLNs, where an adversary manipulates the version number of DIO messages to achieve the goal of increasing communication delay and overhead.

## III. The Proposed Digital Signature Based Countermeasure

**Overview of RPL Routing Protocol:** In order to provide a specific routing solution for Low Power and Lossy Networks, where a set of resource-constrained devices (later nodes) communicate directly or indirectly through lossy links with high packet error rate and link outages, a novel distance vector and source routing protocol, named *RPL*, is proposed in [1]. RPL organizes nodes in one or more Destination-Oriented Directed Acyclic Graphs (DODAGs) to maintain the network state information. Each DODAG consists of a number of normal nodes and one DODAG root, and is distinguished by RPL Instance ID, DODAG ID, and DODAG Version Number. In DODAG, normal nodes collaboratively collect and forward information to the DODAG root, while the DODAG root is responsible for connecting to the Internet.

After the deployment of LLNs, the DODAG root will first issue a DAG Information Object (DIO) control message to construct a DODAG and build upward routes directed from other nodes to the DODAG root. The DIO control message includes the DODAG root's ID, the rank of the DODAG root, and an Objective Function which describes the routing metrics and constraints. When a node receives the DIO message and is willing to join the DODAG, it adds the sender of DIO message to its parent list, computes its own rank according to the piggybacked Objective Function, and passes on the DIO message with the updated rank information. When the DIO message reaches the border node, the upward route is built and each node can send/forward the information to the DODAG root through its parent list. In order to build end-to-end communication (downward routes) from the DODAG root to other nodes, the border node needs to issue a Destination Advertisement Object (DAO) control message to propagate reverse route information and record the nodes visited along the upward routes. If a new node wants to join the existing DODAG, it can request topology information from the adjacent nodes in the existing DODAGs by multicasting a DAG Information Solicitation (DIS) control message.

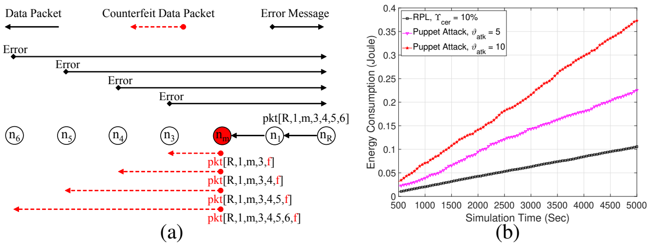


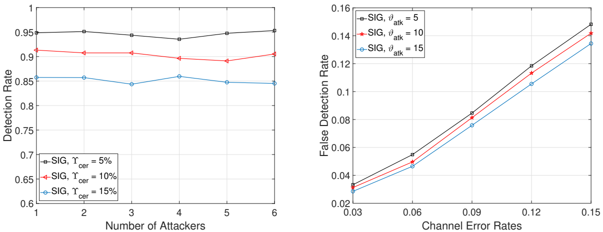
Fig. 1. An example of puppet attack and the performance impact of puppet attack: (a) A snapshot of the network, where a malicious node $n_m$ sends the manipulated packets piggybacked with invalid source route to legitimate nodes. Here, $f$ is the fictitious node address that does not exist in the network. (b) The performance of energy consumption against simulation time.

In RPL, each node is not required to store routing information of downward routes which can be used to send data packets to other nodes, since the routing information of downward routes is stored at the DODAG root. When the DODAG root wants to send a data packet to a node, it first searches its downward routing table. If a downward route exists, it sends the data packet piggybacked with the cached downward route to the node. If a node has a data packet to send, it has to send the data packet to the DODAG root through the upward route (parent list) since it does not store any downward routing information. When the DODAG root receives the data packet, it forwards the data packet to the destination node through the stored downward route. In this process, if an intermediate node along the forwarding path fails to forward the data packet according to the piggybacked source route, it discards the data packet and replies an Error message back to the DODAG root. In order to implement a strict source routing, the piggybacked source route in the data packet specifies every hop between the source node and destination node, including the source and destination nodes.

**Puppet Attack:** The basic idea of puppet attack is that an attacker turns legitimate nodes into puppet nodes by sending them counterfeit data packets piggybacked with manipulated source route. When the puppet nodes receive counterfeit data packets with invalid source route, they have to discard all counterfeit data packets because of invalid source route, and reply a large number of Error messages back to the DODAG root according to RPL. Consequently, all data packets are dropped by legitimate nodes and an excessive number of Error messages exhaust the network communication bandwidth and node energy, finally resulting in a denial of service. For instance, as shown in Subfig. 1(a), suppose that the DODAG root $n_R$ has a data packet to send to node $n_6$. $n_R$ first searches its downward routing table and find a source route, $\{n_R, n_1, m, n_3, n_4, n_5, n_6\}$, and then, send the data packet piggybacked with the found source route to $n_6$. However, when an attacker $n_m$ receives the data packet, it may manipulate the source route by replacing all the post-hops (i.e., $\{n_4, n_5, n_6\}$) of the intermediate node (i.e., $n_3$) with a fictitious destination (i.e., $n_f$), generate a counterfeit data packet piggybacked with an invalid source route, and send the counterfeit data packet to $n_3$ and turn it into puppet node. Here, $n_f$ is the fictitious node that does not exist in the network. When $n_3$ receives the counterfeit data packet, $pkt[R,1,m,3,f]$, it has to discard the data packet and reply an Error message back to $n_R$ since the counterfeit data packet cannot be forwarded further to the next-hop node $n_f$ according to the piggybacked source route. In the worst scenario, an attacker can generate multiple counterfeit data packets piggybacked with invalid source route, and send them to multiple downstream nodes to turn them into puppet nodes. As a result, all puppet nodes have to discard the received data packets, and reply a large amount of Error messages back to the DODAG root. In Subfig. 1(b), the energy consumption of normal nodes is observed against simulation time under puppet attack. As the simulation time elapses, the energy consumption of normal nodes increases quickly, compared to the authentic RPL without puppet attack.

**The Proposed Digital Signature Based Countermeasure:** First, each node including the DODAG root in the DODAG is assigned a public and private key-pair. The public key is publicly available to all nodes in the DODAG, and the private key is the node's private information that is used to validate a node's identity. When the DODAG root $n_R$ has a data packet to send or forward, it first searches its downward routing table for the source route to the destination node, and then puts its address and the stored source route into the source address field and source route field of the data packet. In addition, the DODAG root $n_R$ applies the secure hash algorithm on the source route to generate an output size of 256 bits (SHA-256) as a message digest ($msg_{digest}$). The calculation of a message digest is represented as $msg_{digest} = Hash(src_{route})$, where $src_{route}$ is the piggybacked source route in the data packet and *Hash()* is a predefined hash function. Then, the DODAG root $n_R$ encrypts the message digest $msg_{digest}$ with its private key $PrK^R$ to form a digital signature $SIG_R$ according to $SIG_R = Enc_{PrK^R}(msg_{digest})$. Here, $Enc_{PrK^R}(msg_{digest})$ denotes the encryption on a fixed-length message digest $msg_{digest}$ with private key $PrK^R$ of the DODAG root $n_R$. Finally, the DODAG root $n_R$ attaches the calculated digital signature $SIG_R$ in the data packet and sends the data packet $pkt^{src_{route}}$ along the source route to the destination node.

Second, when an intermediate node (e.g., $n_i$) along the forwarding path receives the data packet ($pkt^{src^*_{route}}$) forwarded by an upstream node (e.g., $n_j$), it verifies the attached digital signature ($SIG^*_R$) using the public key of the DODAG root $n_R$. The intermediate node $n_i$ first hashes the piggybacked source route $src^*_{route}$ into a message digest $msg^*_{digest}$ according to $msg^*_{digest} = Hash(src^*_{route})$. And then, it decrypts the piggybacked digital signature $SIG^*_R$ by using the source node's public key $PuK^R$ according to $msg_{digest} = Dec_{PuK^R}(SIG^*_R)$, where $Dec_{PuK^R}(SIG^*_R)$ denotes the decryption on digital signature $SIG^*_R$ with public key $PuK^R$. If $msg^*_{digest}$ based on the piggybacked source route $src^*_{route}$ equals to $msg_{digest}$ retrieved from attached digital signature, the source route is valid and $n_i$ chooses to forward the received data packet to the next node located in the source route of data packet. Otherwise, the source route is invalid and has been modified, and $n_i$ prosecutes upstream node $n_j$ for performing puppet attack. Thus, $n_i$ drops the received data packet and increases the number of detected puppet attack of suspected node $n_j$ by one. When the number of detected puppet attack of suspected node $n_j$ reaches a threshold ($\varsigma$), the detection node $n_i$ will broadcast an *Alarm* packet to its one-hop neighbor nodes. When the neighbor nodes receive the *Alarm* packet, they will stop receiving the data packets from the suspected node $n_j$, which can isolate the suspected node $n_j$ from the network.



(a) Detection Rate (b) False Detection Rate

Fig. 2. The performance of detection rate and false detection rate against the number of attackers and channel error rate.

## IV. Performance Evaluation

Voluminous simulations are conducted using OMNeT++ [4] to investigate the performance of the proposed digital signature based countermeasure. In a 150×150 m$^2$ square network simulation area, there are 50 nodes and one DODAG root are deployed and uniformly distributed. The communication range of each node is set to 30 meters. The radio model simulates CC2420 with a normal data rate of 250 Kbps, and 802.15.4 MAC/PHY operates with a default configuration in the 2.4 GHz band [23]. Packet injection rate is set to 0.1 pkt/sec to emulate low data rate scenario in LLNs running with RPL. The total simulation time is 5000 seconds.

First, we measure the detection rate against the number of attackers and channel error rate in Subfig. 2(a). Overall, the detection rate of the proposed countermeasure is not sensitive to the number of attackers. For example, with the channel error rate $\Upsilon_{cer} = 10\%$, the detection rate is maintaining around 90% as the number of attackers increases in the network. Since each intermediate node verifies the piggybacked digital signature and source route in the received data packet, if the attacker modifies the source route or attaches the invalid digital signature, the intermediate node can immediately detect the misbehavior. As the channel error rate $\Upsilon_{cer}$ increases, a lower detection rate is observed because worse channel quality will cause more bit errors during the packet transmissions. When the attacker performs puppet attack by modifying the source route, if the bit error happens in the forwarded data packet, the misbehavior of the attacker will not be detected.

Second, we measure the false detection rate against the channel error rate and the number of attackers in Subfig. 2(b). As the channel error rate $\Upsilon_{cer}$ increases, the false detection rate increases as well. This is because a larger channel error rate will cause the bit error in more data packets during the packet transmissions, the bit error in the data packets can hide the modification of source route by the attackers and make puppet attack undetected. As the number of attackers $\vartheta_{atk}$ increases, a lower false detection rate is observed. Since more attackers exist in the network and perform puppet attack, more invalid data packets with error route will be generated and sent to the legitimate nodes. However, these invalid data packets with error route can be detected by the proposed countermeasure, thus, a higher detection rate is observed.

Third, we measure the energy consumption against the number of attackers and channel error rate in Fig. 3. For the original RPL under puppet attack, as the number of attackers increases, the energy consumption significantly increases. This is because more attackers can generate and send more invalid data packets with error route to cause the legitimate nodes to drop the data packets and reply an excessive number of Error messages. As a result, each intermediate node along the forwarding path will receive and forward a large number of packets, which consume more energy. With a larger channel error rate, a higher energy consumption is observed. However, the proposed countermeasure can significantly decrease the energy consumption under puppet attack. Since each intermediate node can verify the piggybacked digital signature and source route, it can detect any modification of source route from the suspected node. When the number of detected misbehavior of suspected node reaches a threshold value, it will isolate the attacker by rejecting all data packets from the attacker. In this way, a less number of invalid data packets will be received by the legitimate node, thus, the number of Error messages can be significantly reduced, leading to a lower energy consumption.

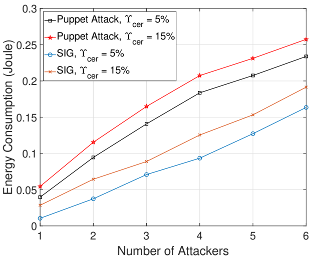


Fig. 3. The performance of energy consumption against the number of attackers.

## V. Conclusion

In this paper, we investigate the puppet attack targeting on the vulnerabilities of RPL routing protocol, and then propose a corresponding countermeasure against puppet attack in LLNs. We first analyze the basic RPL operations and its potential vulnerabilities, and investigate the performance impact of puppet attack with a preliminary result in terms of energy consumption. Then a digital signature based countermeasure along with other techniques, also referred to as *SIG*, is proposed to detect and mitigate puppet attack. Extensive simulation results indicate that the proposed countermeasure can not only detect puppet attack with high detection rate, but also can reduce the performance impact of puppet attack in terms of energy consumption, suggesting a new approach against puppet attack in the Internet of Things.

## References

[1] T. Winter and P. Thubert, “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks,” *RFC Standard 6550*, March 2012.

[2] A. Nia and N. Jha, “A Comprehensive Study of Security of Internet-of-Things,” *IEEE Trans. Emerg. Topics Comput.*, vol. 5, no. 4, pp. 586–602, 2017.

[3] T. Tsao, R. Alexander, M. Dohler, V. Daza, A. Lozano, and M. Richardson, “A Security Threat Analysis for the Routing Protocol for Low-Power and Lossy Networks (RPLs),” *RFC Standard 7416*, January 2015.

[4] A. Varga, *OMNeT++*, 2014, http://www.omnetpp.org/.

[5] C. Pu and S. Lim, “Spy vs. Spy: Camouflage-based Active Detection in Energy Harvesting Motivated Networks,” in *Proc. IEEE MILCOM*, 2015, pp. 903–908.

[6] C. Pu, S. Lim, B. Jung, and J. Chae, “EYES: Mitigating Forwarding Misbehavior in Energy Harvesting Motivated Networks,” *Elsevier Computer Communications*, vol. 124, pp. 17–30, 2018.

[7] C. Pu, S. Lim, J. Byungkwan, and M. Manki, “Mitigating Stealthy Collision Attack in Energy Harvesting Motivated Networks,” in *Proc. IEEE MILCOM*, 2017, pp. 575–580.

[8] C. Pu and S. Lim, “A Light-Weight Countermeasure to Forwarding Misbehavior in Wireless Sensor Networks: Design, Analysis, and Evaluation,” *IEEE Systems Journal*, vol. 12, no. 1, pp. 834–842, 2018.

[9] C. Pu, S. Lim, J. Chae, and B. Jung, “Active Detection in Mitigating Routing Misbehavior for MANETs,” *Wireless Network*, vol. 25, no. 4, pp. 1669–1683, 2019.

[10] C. Pu, “Jamming-Resilient Multipath Routing Protocol for Flying Ad Hoc Networks,” *IEEE Access*, vol. 6, pp. 68 472–68 486, 2018.

[11] C. Pu, P. Nathaniel, and B. Jacqueline, “Self-Adjusting Share-Based Countermeasure to Interest Flooding Attack in Named Data Networking,” in *Proc. IEEE CPSCom*, 2019, pp. 142–147.

[12] C. Pu and S. Hajjar, “Mitigating Forwarding Misbehaviors in RPL-based Low Power and Lossy Networks,” in *Proc. IEEE CCNC*, 2018, pp. 1–6.

[13] C. Pu, “Mitigating DAO Inconsistency Attack in RPL-based Low Power and Lossy Networks,” in *Proc. IEEE CCWC*, 2018, pp. 570–574.

[14] C. Pu, X. Zhou, and S. Lim, “Mitigating Suppression Attack in Multicast Protocol for Low Power and Lossy Networks,” in *Proc. IEEE LCN*, 2018, pp. 251–254.

[15] C. Pu and X. Zhou, “Suppression Attack Against Multicast Protocol in Low Power and Lossy Networks: Analysis and Defenses,” *Sensors*, vol. 18, no. 10, p. 3236, 2018.

[16] C. Pu, “Spam DIS Attack Against Routing Protocol in the Internet of Things,” in *Proc. IEEE ICNC*, 2019, pp. 73–77.

[17] C. Pu and T. Song, “Hatchetman Attack: A Denial of Service Attack Against Routing in Low Power and Lossy Networks,” in *Proc. IEEE CSCloud*, 2018, pp. 12–17.

[18] C. Pu, “Energy Depletion Attack Against Routing Protocol in the Internet of Things,” in *Proc. IEEE CCNC*, 2019, pp. 1–4.

[19] C. Pu and B. Groves, “Energy Depletion Attack in Low Power and Lossy Networks: Analysis and Defenses,” in *Proc. IEEE ICDIS*, 2019, pp. 14–21.

[20] B. Ghaleb, A. Al-Dubai, E. Ekonomou, M. Qasem, I. Romdhani, and L. Mackenzie, “Addressing the DAO Insider Attack in RPL’s Internet of Things Networks,” *IEEE Commun. Lett.*, vol. 23, no. 1, pp. 68–71, 2019.

[21] P. Thulasiraman and Y. Wang, “A Lightweight Trust-Based Security Architecture for RPL in Mobile IoT Networks,” in *Proc. IEEE CCNC*, 2019, pp. 1–6.

[22] A. Arış, S. Yalçın, and S. Oktuğ, “New lightweight mitigation techniques for RPL version number attacks,” *Ad Hoc Networks*, vol. 85, pp. 81–91, 2019.

[23] A. Boulis, *Castalia*, 2014, http://castalia.forge.nicta.com.au.