Poster Abstract: Toward Efficient and Secure Code Dissemination Protocol for the Internet of Things

Jun young Kim, Wen Hu, Sanjay Jha
School of Computer Science and Engineering
UNSW Australia
{junyoungkim,wen.hu,sanjay}@cse.unsw.edu.au

Hossein Shafagh
Department of Computer Science
ETH Zurich, Switzerland
shafagh@inf.ethz.ch

Mohamed Ali Kaafar
National ICT Australia
dali.kaafar@nicta.com.au

ABSTRACT

Current Wireless Sensor Networks (WSNs) approaches do not provide an efficient and secure code dissemination function due to emerging issues of IoT applications. In this work, we adopt a multicast approach instead of the existing end-to-end or epidemic approaches. In order to enable the multicast approach, we propose an efficient/robust group key distribution scheme. We will evaluate and quantify the performance of our prototype implementation in a public testbed, while emulating several practical IoT settings, and show our security measures against known attack models.

1. MOTIVATION

Securely updating remotely deployed nodes, commonly referred to as code dissemination is a necessary function for the ongoing success of the IoT. Thread group[1] depicts a good use-case example, which is organized by major IT companies such as ARM, Qualcomm, and Samsung. The main goal of this IoT project is to securely connect/manage a large number (+250) of various devices in home networks. Since the vision is to run even battery-powered devices for years, low-power and secure over-the-air update is a necessary and essential functionality. Communication and computation are two premium resources for the energy friendly design, as radio activities and security operations consume the major share of energy on constrained devices. Unfortunately, existing code dissemination solutions for WSNs exhibit poor efficiency in the context of the IoT due to several emerging IoT issues [6], such as:

- **Heterogeneity**: It is challenging to design a framework that can manage various types of devices. In order to support the heterogeneity with existing solutions, applications coherently become more complex and suffer from performance degradation.
- **Mobility**: Having a coherent and stable view of the network topology is a significant factor for managing IoT applications. Mobility related dynamics make this process difficult and result into inefficiencies.
- **Connectivity**: IoT applications naturally come with the Internet connection capability, which results in exposure to a wider types of adversaries. It is desirable to identify these emerging threats, and be able to quickly provide efficient solutions.

Existing over-the-air (re)programming protocols for WSNs are based on the epidemic communication [5] which assumes homogenous sensor nodes in the network with all nodes participating in the (re)programming process. Given the multitude of systems in an IoT network, the epidemic approach would require non-target nodes to participate in the propagation resulting in unnecessary cryptographic operations and additional packet forwarding overhead. In this paper, we adopt a multicast communication approach, where only target nodes involved in the code-image update actively take part in the process.

In order to enable a secure multicast, we rely on secure group key distribution among dynamic set of target nodes. However, existing group key distribution schemes may not be suitable for IoT deployments since they assume known/fixed group sizes, unscalable overhead, or neighbor pairings. For example, End-to-End (E2E) schemes exhibit high communication overhead, Pairwise schemes require neighbor pairing, and Pre-Shared-Key (PSK) solutions cannot support key revocation. Although Broadcast Encryption (BE) [3] exhibits excellent distinctions, most BEs are k-resiliency systems, where $k+1$ exposed keys can collude to generate shared secrets. Since IoT nodes are vulnerable to physical node capture attack, where adversaries compromise nodes to extract pre-installed keys, achieving a full or higher collusion resistance requires infeasible communication/storage overhead.

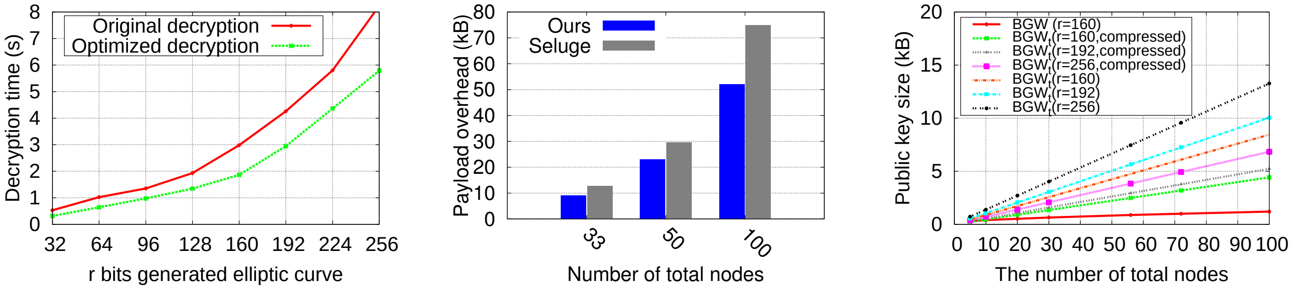
2. OUR APPROACH

We investigated several Public key cryptographic-based Broadcast Encryption schemes (PBE) (see Table 1) to address the collusion issue of BE schemes. Based on our initial study, we propose to use BGW_t [1], which results in additional storage but improves overall communication/computation efficiency with following benefits:

1. **The shortest ciphertext size**: Shorter or fixed size ciphertext can improve the communication overhead for the group key distribution. BGW_t achieves the shortest ciphertext size of $O(1)$, but trades off linear storage to store all other nodes' public keys.

[1] http://threadgroup.org/

SenSys'15, November 1–4, 2015, Seoul, South Korea..
ACM 978-1-4503-3631-4/15/11.
DOI: http://dx.doi.org/10.1145/2809695.2817896.



(a) Our group key decryption time before/after optimization.

(b) Key distribution overhead comparison between our approach and Seluge.

(c) Our storage overhead is linear, but feasible in practical settings

Figure 1: Initial evaluation results of our prototype implementation.

	⟵Key/ciphertext size⟶			Complexity	
	Public	Private	Ciphertext	λ_E	λ_D
Trivial	$O(n)$	$O(1)$	$O(t)$	$O(t)$	$O(1)$
Delerablée$_1$ [2]	$O(n)$	$O(1)$	$O(r)$	$O(r^2)$	$O(r)$
Delerablée$_2$ [2]	$O(n)$	$O(n)$	$O(1)$	$O(r^2)$	$O(r^2)$
BGW [1]	$O(\sqrt{n})$	$O(1)$	$O(\sqrt{n})$	$O(\sqrt{n})$	$O(\sqrt{n})$
BGW_t [1]	$O(n)$	$O(1)$	$O(1)$	$O(n)$	$O(1)$

Table 1: Comparison of our short listed PBEs [2] in an n-node network targeting arbitrary t nodes, r non-target nodes. Note that λ_E denotes encryption complexity, λ_D denotes decryption complexity.

2. **Lower decryption complexity**: On a dissemination protocol, encryption is performed at the level of a resource-rich server while decryption is performed at the level of constrained nodes. Lower decryption complexity can enhance the energy consumption.

3. **Fixed size private key**: BGW_t has $O(1)$ private key size and the key can be stored in a tamper resistant storage to effectively mitigate physical attacks.

4. **Key revocation efficiency**: Key revocation is necessary when a key is lost or a node is compromised. The majority of BE schemes requires linear or logarithmic overhead for the key revocation process. However, our revocation process only requires updating one ciphertext regardless of the number of revoked keys.

5. **Dynamic targeting**: PBEs including BGW_t can dynamically change the target group.

Once we securely distribute a group key to arbitrary targets, we leverage the key for further security requirements.

3. EARLY RESULTS

We implement our node side BGW_t on the Openmote platform[2]. Although the group key decryption is only required once per dissemination session, it is still computationally expensive. To address this, we optimize our node side BGW_t implementation with elliptic curve optimization techniques by means of the pre-computation technique that Boneh et al. [1] suggested for an efficient implementation. Fig. 1(a) shows the optimized group key decryption (e.g., from 2982 ms to 1864 ms in 160 bits generated curves).

We show the efficiency gain of our key distribution overhead compared to Seluge [4], a state-of-the-art epidemic approach. Fig. 1(b) depicts our approach's superior key distribution and performance in medium sized networks. The efficiency gain gap between ours and Seluge is proportional to the size of the network since Seluge's key establishment overhead depends on the number of neighboring nodes. This gap will be wider in dense networks such as home IoT applications. We plan to present the propagation gain for bulk data transfer. We expect our approach to improve all aspects from Seluge such as group key distribution, code propagation, security, collusion resistance, and key revocation, except pre-key-installation overhead. Although our storage overhead is linear, it is feasible in practical settings as it is quantified in Fig. 1(c). For instance, 4.4 kB storage is required in a 100 node network with 160 bits generated elliptic curves.

4. CONCLUSION

In this project, we propose a secure code dissemination protocol to address emerging IoT issues. Initial results suggest that our approach is more efficient and robust compared to existing WSNs solutions. We will evaluate the performance of our prototype implementation in a public testbed.

5. ACKNOWLEDGMENT

This work is partially supported by Australian Research Council Discovery grant DP150100564.

6. REFERENCES

[1] D. Boneh, C. Gentry, and B. Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In *Advances in Cryptology–CRYPTO 2005*, pages 258–275, 2005.

[2] C. Delerablée, P. Paillier, and D. Pointcheval. Fully collusion secure dynamic broadcast encryption with constant-size ciphertexts or decryption keys. In *Pairing*, pages 39–59. 2007.

[3] A. Fiat and M. Naor. Broadcast encryption. In *Advances in Cryptology-CRYPTO*, pages 480–491, 1994.

[4] S. Hyun, P. Ning, A. Liu, and W. Du. Seluge: Secure and dos-resistant code dissemination in wireless sensor networks. In *ACM/IEEE IPSN*, 2008.

[5] P. A. Levis, N. Patel, D. Culler, and S. Shenker. *Trickle: A self regulating algorithm for code propagation and maintenance in wireless sensor networks*. Computer Science Division, University of California, 2003.

[6] M. Zorzi, A. Gluhak, S. Lange, and A. Bassi. From today's intranet of things to a future internet of things: a wireless-and mobility-related view. *Wireless Communications, IEEE*, 17(6):44–51, 2010.

[2] `http://www.openmote.com/`