Lower Power Data Transport Protection for Internet of Things (IoT)

Chih-Ta Lin, Cheng-Yu Tsai and Chuan-Kai Kao
CyberTrust Technology Institute
Institute for Information Industry
Taipei, Taiwan

Abstract The concerns over IoT (Internet of Things) data transport security impede the development of smart service. The current data are transported mainly in clear text with a lack of privacy protection. IoT equipment resources are limited, thus complex encryption and decryption are not applicable. Google disclosed a Randomized Aggregatable Privacy-Preserving Ordinal Response (RAPPOR) algorithm that provides strong privacy guarantees for data transport, which has been efficiently implemented in Chrome for privacy protection of anonymous reporting. However, the data restoration rate is still subject to 1 to 5% confidence error. In this study we improved the Randomized Response process and proposed a Time Based Dynamic Response method. Experimental results demonstrate that the data transport error rate can be reduced to 0% in a limited range. Moreover, the computational operations can save more than 90% of time cost as compared to the common encryption and decryption techniques (such as AES), which is suitable for the applications in data transport protection between lower-power lightweight IoT devices.

Keywords—Data Transport Security, Internet of Things (IoT), Bloom Filter, Randomized Response, Data Privacy, Encryption.

I. INTRODUCTION

Concerns over the security of data transferred across existing network of Internet of Things (IoT) have been impeding its development. Currently, data is transported primarily in clear text, without adequate measures for privacy protection. In general, data encryption is used for confidentiality. Popular cipher algorithms for data encryption include RSA, ECC, AES, and 3DES. Most of these algorithms are very computation-intensive and require dedicated hardware or Digital Signal Processors (DSP) for time saving [1]. However, IoT equipment resources are limited and complex encryption and decryption procedures are often not feasible. Usually, data transfer for IoT and industrial applications is accomplished using a fixed data format. Mostly, the amount of data transferred is moderate, primarily comprising numerical values. This study presents a simplified data transfer protection mechanism based, largely, on a fixed data format and scope.

Úlfar *et al.* (Google Inc.) [2] published the Randomized Aggregatable Privacy-Preserving Ordinal Responses (RAPPOR) algorithm that provides robust privacy protection for data transport. RAPPOR uses Bloom Filter and Randomized Response method for encoding protection of string data, and then transfers it to the aggregator to estimate the original data through a novel decoding algorithm. The algorithm has been efficiently applied in Chrome for privacy protection of information reporting. These aggregators are based on two assumptions, namely "only need to learn the distribution of a single variable, in isolation" and "know the data dictionary of possible string values in advance." However the application-wise data restoration rate is still subject to a confidence error ranging from 1 to 5% [3].

In this paper, we proposed the Time Based Dynamic Response (TBDR) method to ensure the accuracy of data restoration by negotiating the randomized parameters and dynamic IDs of both parties during the data exchange. The experiment proved that the error rate of the data string transport can be reduced to 0% in a limited data entry range. Further, the computation can save more than 90% of time as compared to common encryption and decryption techniques (such as AES), which are suitable for data transport protection between lightweight IoT devices.

II. DATA ENCODING

RAPPOR [2] uses three processes to encode the data, which include generation of the signal by a Bloom filter [4], a Permanent Randomized Response to a fake Bloom filter, and an Instantaneous Randomized response to confuse. The identical data string remains the same after being processed by the Bloom filter and Permanent randomized response. However, differences would emerge after each processing using instantaneous randomized response, which increases the difficulty of cracking. However, there would still be errors in terms of the statistical restoration. The encoding workflow proposed in this study is shown in Figure 1, which is divided into the establishment of zero error model based on pre-training and parameter adjustment and the application of data encoding protection.

978-1-5090-5569-2/17/$31.00 ©2017 IEEE

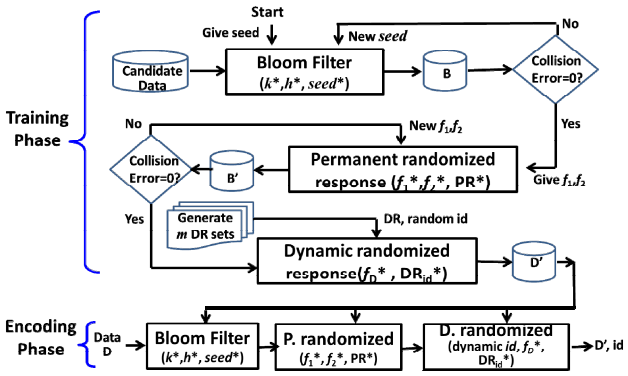

Figure 1 The flowchart of TBDR encoding method.

A. *Training Phase*

This program covers three processes including Bloom filter, Permanent randomized response, and Dynamic randomized response. First two processes are required to perform the zero-error-rate oriented parameter adjustment.

- Bloom filter: A *k*-size Bloom filter uses *h* hash functions to hash the data string *s* onto a signal. The collision rate of signals is related to *k* , *h and seed* of Bloom filter. For example, usage of $k = 256$ & $h = 4$ can discriminate C_4^{25} different values, and about 3 to 4 bytes of numerical data can be distinguished. Generally in terms of the *n*-bytes data, let $h = n + 2$, or increase the size of *k*, taking a commonly transferred industrial control Modbus packet as an example:

  20 00 00 00 00 06 01 06 00 00 00 01

  byte 1 - 2 indicate Transaction Identifier, byte 5 - 6 designate the length of the Modbus command, generally about 6 bytes, byte 7 is the Slave Address, and byte 8 - 12 are Function Code, Data Address, and value to write that require data protection. For the transport protection of general lightweight industrial control instructions, $k = 256$ & $h = 7$ should well satisfy the requirements. To confirm a Bloom filter with zero error rate, the *seed* must undergo a trial & error process to identify the optimal combination, wherein *seed* is a combination of a series of prime numbers.

- Permanent randomized (PR) response: RAPPOR[2] gives a fixed randomizer and probability ***f*** to change data bits to 1 or 0. This thesis proposed 2 probability factors to expand the flexibility of parameter adjustment.

$$B_i' = \begin{cases} 1, & \text{with probability } f_1 \\ 0, & \text{with probability } f_2 \\ B_i, & \text{with probability } 1 - f_1 - f_2 \end{cases}$$

  Where the probability f_1, f_2 is tunable parameters for 0 collision experiment, and $0 \le f_1, f_2 \le 1$. If $f=0$ means no changed on data.

- Dynamic randomized response: This step transports the original data as different content each time after dynamic obfuscation so that eavesdroppers cannot use the regularity to crack transport protection. This paper improved the Instantaneous randomized response method of RAPPOR [2] and proposed a one-time randomized response (OTRR) algorithm from a shared randomized response set and probability f_D. OTRR pre-generated *m* dynamic randomizers (DR), performed bits flip operation on the data using probability f_D.

B. *Encoding Phase*

IoT data transport protection employs the model and parameters with zero error rate for data encoding. For an original data string, first perform Bloom filter process using the tuned *k**, *h**, and *seed**, and then perform Permanent randomized response under the condition of probability = (*f1*, f2**). Finally randomly generate dynamic identifier *id* within the range 1 to *m*, and further encode the data by ***id***th (***DR***, f_D*) to output encoded data *D′*, and then pass *D′* and *id* to the aggregator.

III. Data Decoding

A candidate list table ***C*** was prior estimated for rapid restoration analysis：

- For each of candidate data strings $\boldsymbol{S_i}$，use Bloom filter and parameters ***k****, ***h****, and ***seed**** to establish the data set $\boldsymbol{B}_{\_all}$
- For those bits in $\boldsymbol{B}_{\_all}$ that have not been processed by Permanent randomized response, build candidate list table ***C*** of which bit=*true,* for example*:*

  bit 1: S_5, S_{55}, S_{235}, S_{276}, S_{470},…

  bit 2: S_{46}, S_{202}, S_{243}, S_{284}, S_{530},…

Where S_5 represents the 5th candidate string, indicating that the 1st bit is true after the string undergoes Bloom filter and PR operations. During the formal decoding process, the received encoded data ***D′*** will firstly undergo the inverse bits flip operation using ***id***th (***DR***, f_D*) to reconstruct into ***B′***, and then the correct original data strings will be obtained from ***C*** based on matching statistics:

- For those bits in ***B′*** that have not been processed by PR response, find each candidate list string ***id*** wherein bit = *true.*
- Collect statistics for each candidate list string ***id*** to obtain the total number of (bit=*true)* appeared, and the string with the highest $\boldsymbol{n}_{max}$ is the original data string.

$h = 4$ denotes that only 4 bits are true. Generally, the original string can be determined when $\boldsymbol{n}_{max} = 4$. In case the *true* bits are processed as 0s by the permanent randomized response, $\boldsymbol{n}_{max}$ will be reduced, and the candidate original string number will increase, then further accurate comparison

978-1-5090-5569-2/17/$31.00 ©2017 IEEE

of candidate $\boldsymbol{B}$ using B' is required. In this experiment, 98.81% of all tested strings have $\boldsymbol{n}_{max}$ = 4, which can be immediately identified as original strings. While in the remaining 1.19%, the maximum candidate original string number is 10, that is, it only needs to further compare 10 times to find the correct original string.

IV. EXPERIMENTS AND CONCLUSIONS

The candidate data strings from "0" to "9999" were tested by the encoding and decoding process. The experiment chose $\boldsymbol{seed}$ = [2,3,7,13], $\boldsymbol{h}$ = 4, $\boldsymbol{k}$ = 256 for Bloom filter, $\boldsymbol{f}_1$=0.02, $\boldsymbol{f}_2$=0.01 for Permanent randomized response, and $\boldsymbol{f}_D$=0.3, $\boldsymbol{m}$=100 for Dynamic randomized response. Experimental results show that the collision error rate is 0% at each processing stage, and the error rate of the original string after encoding / decoding is 0%. The time cost comparison between the method proposed in this study and the traditional encryption and decryption methods is shown in Table 1. The time and cost for 100,000 times of operations under the condition of 256-bit protection, the proposed method offers significantly lower time cost than the traditional AES and DES algorithms, thereby effectively achieving the low-power competitive performance.

TABLE I. COMPARISON OF TIME COSTS FOR 100K OPERATIONS OF DIFFERENT ALGORITHMS (SEC)

Method \ Key Length (bits)	128	192	256
AES	766	782	811
DES	954	962	1005
TBDR			56

References

[1] A. Ukil, J. Sen, and S. Koilakonda, "Embedded security for Internet of Things", Emerging Trends and Applications in Computer Science (NCETACS), 2nd National Conference on, pp. 1-6, March 2011.

[2] Ú. Erlingsson, V. Pihur, and A. Korolova, "RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response", Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, November 03-07, 2014, Scottsdale, Arizona, USA

[3] Fanti et Al. "Building a RAPPOR with the Unknown: Privacy-Preserving Learning of Associations and Data Dictionaries", Cornell Univ. Lib., Mar. 2015 summit

[4] B. H. Bloom. "Space/time trade-offs in hash coding with allowable errors". Commun. ACM, 13(7):422-426, July 1970.

978-1-5090-5569-2/17/$31.00 ©2017 IEEE