Data Provenance for Multi-Hop Internet of Things

J. Gowthami (AP/IT)  
Department of Information Technology,  
Nandha College of Technology,  
Erode, India.  

K. Aishwariya, P. Divya,  
M. Kavin Kumar, A. Elangovan  
(UG Scholar)  
Department of Information Technology,  
Nandha College of Technology,  
Erode, India.  

Abstract— Security protocols for Internet of Things (IoT) need to be light weighted due to limited resources and scalability are available in the internet. Small and low-energy devices are suitable for cryptographic solution because of their energy and space limitations. The Received Signal Strength Indicator (RSSI) is used to generate link fingerprints for communicating IoT nodes. Correlation Coefficient is used for matching the link fingerprint. Co-relation Coefficient is used to communicating secured data transfer. Adversarial Node can be detected for specific link in between the two nodes. Data Provenance has achieved by comparison of packet header. 

Keywords— Light weight, Data Provenance, Fingerprint, Adversarial Node, Received Signal Strength Indicator (RSSI)

I. INTRODUCTION

INTERNETS of Things become necessary for the future work. It controls all devices in network. More than 50 million devices will be connected through the IoT. The IoT can be connected by the following process, Medical staff, Automobile performance and statistics, Home device control, Transportation domain, Smart grids and smart meters. Data gathered from sensors and it is propagated to Internet cloud. The nodes are Large in number, Small in size, mostly accessible. At the receiving end the measures should be make that the data is secured and effectively received.[16] IoT nodes are not physically protected therefore data security and data provenance act as backbone to implement IoT network. If any proper security. Not taken then the data can be easily forged or tempered.

The security primitives are Detection of certain attacks, Masking channel stated, Intrusion detection, Location distension and Data provenance the origin of data are find by the data provenance.[15] A single change can occur in the data that might cause big problems.[2] Due to the energy limitations of the IoT nodes the traditional cryptographic techniques are not viable solution in IoT. For enabling end-end production it acquires, less space, Energy efficient security primitives with, less computational complexities. The IoT based network should be secured for the trust of users. The low energy requirements of security mechanism involved should be light-weighted. The authentication between the IoT nodes and the server should be secured. Accurate data provenances in the IoT node are used for improving the trust level. [4][The data starting from the original resource that is useful for determining and describing the derivation history from the data provenance. The intellectual property and its relevance can be protected by records.

The security algorithms and the cryptography techniques are used and also contain high computational complexities with high energy consumption.[17] Without using any extra hardware the light-weight security algorithm proposed for secured IoT based information exchange. By correlating the link fingerprints the adversarial node can be detected effectively in the adjacent IoT nodes. The same link fingerprints can be achieved by data provenance to find the intrusion detection in the IoT network.

II. LITERATURE REVIEW

This paper describes design approaches that blend high energy-efficient circuit techniques with optimal accelerator micro architecture data path, and hardware friendly arithmetic to achieve ultra-high energy consumption. Security platforms for adoption in area/battery constrained and self-powered systems.[1] Industry leading low energy efficiency is demonstrated. Fabricated and measured in advanced process technologies. Securing data provenance focuses only security elements on providing partial part of in their mechanisms. Cloud not provides full protection to the data provenance as a whole Paper presents the provenance and challenges description in providing security assurances in the Cloud.

The authors of show that the received signal strength indicator (RSSI) values of a wireless channel can be used as the fingerprint of a wireless link. The RSSI values are quantized using typical quantization mechanisms such as level crossing or ranking techniques. The current RSSI value of the wireless link between two parties is then used as the wire-less fingerprint for a given session.[3] The taxonomy of attacks in IoT are spoofing, altering, replaying routing information, Sybil attack, Denial of Service attacks, attacks based on node property, attacks based on access level, attacks based on adversary location and attacks based on information damage level etc. A secured enough to gain the light-weighted trust of IoT users. IoT network is provided using cryptographic solution to secure the Advanced Encryption Standard (AES)-128. These solutions deal with cryptography and computational complexities.[5] That is why AES-128 algorithm is not suitable for IoT considering a large number of IoT nodes. IoT devices are secured, many research works have been conducted to counter measure those problems and find a better way to eliminate those risks, or at
least minimize their effects on the user’s privacy and security requirements. The survey consists of four segments. They will explore the most relevant first segment limitations of IoT devices and their solutions. The second one will present the classification of IoT attacks. The security of different PUFs designs against these modeling attacks has been evaluated in multiple research efforts.[21] However, a more accurate evaluation that incorporates the area fingerprints of the different PUFs is missing from the literature. Such evaluation is rather essential in order to determine the most suitable for the constrained IoT devices different PUF design. To perform this evaluation, we implemented several strong delay-based PUFs.[6] The results that when factoring in the PUFs implementation area costs, high PUFs compare differently as some enhanced PUF designs turned out to have inferior efficiency compared to their simpler counterpart. We recognized the most effective design elements in PUFs. Based on the efficiency obtained, we recommended the optimal different security schemes used in constrained IoT devices PUF design.

III. BACKGROUND MATERIAL

The two IoT nodes are communicated between the various metric Received Signal Strength Indicator (RSSI). To generate the link fingerprint, Time of arrival, Phasor information, Error vector magnitude is used. For any connected nodes there is a linear relation between the RSSI variations. It is helpful in generating the link fingerprints and it is highly correlated for two connected nodes. This information is to develop the link fingerprints. The RSSI values can be recorded in real time and its duration of RSSI values at each IoT node can be maximized or minimized depending on the power availability to the nodes. The IoT nodes are power limited and realistic approach to take the record time larger and it is not affecting the results. In IoT network there is no adversarial node is present. It is present only in between the two nodes. If the server is secured the adversarial node can send its data.[7] The intrusion can find at last by using the data provenance. It used in real-time experiments.
1. IoT network present in no adversarial node.
2. Two communicating IoT nodes present in Adversarial node.
3. Packet tempered at any IoT node.
4. IoT node is replaced by adversarial node.
5. Adversarial node can send its data to the server but cannot access the data present at the server not secure.
6. Data using provenance algorithm

IV. MODULES

- Adversarial Node
- Data Provenance
- Link Fingerprint

A. ADVERSARIAL NODE

Each node records its RSSI values for every 20 seconds. Friis transmission equation is using for calculating the signal strength

\[ P_r = \frac{P_t \times G_t \times G_r \times L}{d^2} \]

\[ P_t \text{ transmitted power} \]
\[ P_r \text{ received power} \]
\[ G_t, G_r \text{ transmitting and receiving antennas} \]
\[ d \text{ distance between the two communicating nodes} \]

\[ L = \left( \frac{4 \pi d}{\lambda} \right) \]

The result of RSSI values are quantized by the use of word-length of 8 bits. The finite sets of values are mapped onto the amplitude values. It is divided by the Distance between minimum and maximum RSSI values.

\[ \Lambda = \frac{P_r(\text{max}) - P_r(\text{min})}{L} \]

\[ P_r(\text{max}) \text{ maximum received power} \]
\[ P_r(\text{min}) \text{ minimum received power} \]

A value from 0 to \(L-1\) is assigned for each zone of midpoint. The value of the midpoint is approximated for each sample falling in a zone. [20] Each zone is assigned as 8 bit and it is represented as 8 bit word length of link fingerprint (LF). The link fingerprints are encoded with an 8 bit secret key for IoT node 1, IoT node 2 and also IoT node 3.

\[ LF_{encoded} = LF_{ encoded } = K_i \]

\[ LF_{encoded} - encoded link fingerprint \]

In the above formula it represents logical exclusive -OR operation. \( LF_{encoded} \) sends to the server and also keeps a
copy with itself, adversarial node can send its data to the server by replacing IoT node. [8] The server is assumed as highly secured and the data is stored after the authentication is successful.

\[ L_{F_{1-r}} = K_i = L_{F_{encoded}}(1-r) \]

The server decodes all the received encoded link fingerprints of each IoT node using key, the server, which are assumed to be fully protected. The binary coded link fingerprints are converted to the respective decimal values. The value is between 0.8 and 1 is considered as highly correlated in a multi-hop network.

\[ \rho_{XY} = \frac{\text{cov}_X, Y}{\sigma_X \sigma_Y} \]

Cov- covariance

\[ \Sigma \] - standard deviation

A simplified equation is,

\[ \sum_{i=1}^{n} \left( X_i - \bar{X} \right) \left( Y_i - \bar{Y} \right) \]

\[ \sum_{i=1}^{n} (X_i - \bar{X})^2 \sum_{i=1}^{n} (Y_i - \bar{Y})^2 \]

The RSSI values of the packet received at communicating IoT nodes and it respective for IoT sequence of n packets. Were 1 indicates anti-correlation, 0 indicates no correlation, 1 indicates perfect indication. Various cases are implemented and the simulation results are presented for adversarial node detection. The results are achieved by using two methods:

1) Finding Pearson correlation coefficient without using any filter
2) Finding Pearson correlation coefficient by applying Savitzky-Golay filter.

**ALGORITHM 1: Link fingerprint**

IoT node can be initialized

RSSI node can be read from adjacent node

RSSInew[i] ← RSSI[i]+gain

Quantize RSSInew[i]

Link Fingerprint[i] ← Assign binary code to the Quantized RSSInew[i]

RSSInew[i] ← XOR(LinkFingerprint[i],Keynode(j))

RSSInew[i] bundled up with session identifiers

IoT node sends link fingerprint to the server

Keeps a copy of same with itself.

**ALGORITHM 2: Adversarial node**

Link fingerprint[i] ← XOR(RSSInew[i],Keynode(a))

RSSInew[i] ← binary to decimal conversion (LinkFingerprint[i])

Link fingerprint[i] ← XOR(RSSInew[j],Keynode(b))

RSSInew[j] ← binary to decimal conversion (LinkFingerprint[j])

P(RSSInew[i],RSSInew[j])

if -1<p<0.9 then

return adversarial node is present

else if 0.9<p<1 then

return No adversarial node is present

else

return The RSSI values are not correctly measured

end if

**ALGORITHM 3: Data Provenance**

For Headeri,

\[ i = n \rightarrow 1 \text{do} \]

LinkFingerprintHeaderi= = XOR(Headeri,Keyi)

Correlate LinkFingerprintHeaderi with link fingerprints received from IoT node[i]

if Correlation>95% then

return i←i-1

else

Data forged between IoT node[i]and IoT node[i-1]

end if

The packet of the origin is IoTnode[i]

**B. DATA PROVENANCE**

The data provenance, header information is used to reach the origin from which the data is originated. As discussed earlier, each IoT node sends the copy of the link fingerprints to the server, so all the header information will already be
**TABLE 1: Pearson correlation coefficient \( r \) calculated various case**

<table>
<thead>
<tr>
<th>Scenario</th>
<th>IoT node 1 and IoT node 2</th>
<th>IoT node 2 and IoT node 3</th>
<th>Confidence Interval (CI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>( r = 0.9270 )</td>
<td>( r = 0.8420 )</td>
<td>( r = 0.9713 )</td>
</tr>
<tr>
<td>Case 2</td>
<td>( r = -0.0038 )</td>
<td>( r = 0.9280 )</td>
<td>( r = 0.9515 )</td>
</tr>
<tr>
<td>Case 3</td>
<td>( r = 0.8913 )</td>
<td>( r = 0.0628 )</td>
<td>( r = 0.2056 )</td>
</tr>
<tr>
<td>Case 4</td>
<td>( r = -0.0063 )</td>
<td>( r = -0.1740 )</td>
<td>( r = -0.5125 )</td>
</tr>
<tr>
<td>Case 5</td>
<td>( r = -0.2753 )</td>
<td>( r = 0.8369 )</td>
<td>( r = 0.9520 )</td>
</tr>
<tr>
<td>Case 6</td>
<td>( r = 0.8382 )</td>
<td>( r = 0.8590 )</td>
<td>( r = 0.5269 )</td>
</tr>
</tbody>
</table>

**TABLE 2: Data Provenance**

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Correlation of IoT node header with all available LFs at the server</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>100% 100% 100%</td>
<td>100%</td>
</tr>
<tr>
<td>Case 2</td>
<td>100% 100% 100%</td>
<td>40.7407%</td>
</tr>
</tbody>
</table>

Present at the server.[9] If the information is received at IoT node 3 from IoT node 1 via IoT node 2, the link fingerprints of header are compared at the server in sequence with copies of link fingerprints previously sent by the IoT nodes. From whichever IoT node the last header information matches, the data is originated from that IoT node. Size of header depends on the selection of packet size.

1. Link fingerprints are observed highly matched all the header data is exhausted origin. Last IoT node N from which the header data is matched.
2. Link fingerprints showing that the data has been tempered that mismatch occurs.

The server knows the size of header that each IoT node attaches and the adjacent IoT nodes of each IoT node.[10] In order to check the origin from which the data is originated, the server decodes the header with the keys present at the server and correlates the link fingerprint with the already present link fingerprints received from that node. If the link fingerprints match, the same process is repeated for the adjacent IoT node(s).[19] While finding the origin of data, if adversarial node is present between any two IoT nodes and the packet flows through adversarial node then the server will still get high correlated result by comparing the link fingerprints.

The link fingerprints will match the link fingerprints present at the server received from the IoT node.[11] The reason is that if we consider the mentioned situation in the adversarial node is between IoT node 1 and IoT node 2, the IoT node 1 adds the link fingerprint at the header which is of the link between IoT node 1 and adversarial node.[14] Similarly, IoT node 2 adds the link fingerprint of the link between adversarial node and IoT node 2 to the packet header received from adversarial node and forwards it. The last IoT node on receiving it, adds its link fingerprint.[12] The server checks the header for the origin and gets high correlated value after decoding the header inserted by IoT node 2. As IoT nodes will be large in number, the physical protection will not be possible for most of the nodes.[18] The data can be easily forged or tempered. If the data is tampered at IoT node 2 and sent to IoT node 3 afterwards, the data provenance cannot be achieved rather the adversarial node’s involvement can be detected.

The process can tell exactly between which link the data has been forged. This is very useful information in data forensics. The highly uncorrelated result is achieved when comparing the link fingerprints in the header and the ones present at the server. Algorithm 3 represents the achievement of data provenance.

C. **LINK FINGERPRINT**

Performed experiment show that highly correlated fingerprint are acquired. After every 10 to 15 minutes, a link fingerprint of 128 bit is generated by using RSSI. The fingerprint and the secret key will not be shared with any other IoT nodes. High trust can be achieved using the same model in IoT environment.

The server assumes that the link fingerprint is encoded and decodes it with the key of that node which is replaced by adversarial node. Here after performing multiple experiments, it is observed that the correlation coefficient can be high at times but not high enough to remain unnoticed.

V. **EXISTING SYSTEM**

The numbers of nodes are large in number, small in size and mostly accessible, the measures should be taken to make sure that the data is secured and efficiently received at the receiving end.[13] Data security and provenance act as backbone in order to implement IoT network because the IoT nodes are not physically protected easily be forged or tempered if proper security primitives are not taken.

Security primitives include detection of certain attacks, masking channel state, intrusion detection, and location distinction and data provenance. Provenance is to find the origin of the data. A single change in data might cause big
problems e.g., in terms of medical health report generated by an IoT node sent to a doctor, meter reading sent to the company for billing according to the consumption and change in transportation system information. Therefore, the traditional cryptographic techniques are not the variable solution in IoT because of the energy limitations of the IoT nodes.

V. DISADVANTAGES

Primitives with less computational complexities are key building blocks for enabling end-to-end content protection, user authentication, and consumer confidentiality in the IoT world. Less space acquiring and energy efficient security

V. PROPOSED SYSTEM

The proposed trust model is described for cloud computing. Improved energy efficiency is achieved by using Gale-Shapley algorithm which matches D2D pair with cellular user equipment (UEs). Lightweight security algorithms for secured IoT-based information exchange without using extra hardware. Adversarial node is detected effectively by correlating the link fingerprints generated by the adjacent IoT nodes. The correlation coefficient is computed at the server.

Adversarial node is detected effectively by correlating the link fingerprints generated by the adjacent IoT nodes. The correlation coefficient is computed at the server. Data provenance is also achieved using the same link fingerprints generated to find the intrusion detection in the IoT network. Fingerprints are used to authenticate the integrity of data and in the detection of intrusion. The proposed solution has less time complexity compared to other state-of-the-art available solutions

VI. ADVANTAGES

1) No adversarial node is present in the IoT network.
2) Adversarial node is present in between two communicating IoT nodes.
3) The packet is forged or tempered at any IoT node.
4) The IoT node is replaced by adversarial node.
5) Cannot access the data present at the server.
6) Finding the intrusion in later data using provenance Algorithm.

XI. RESULTS

EXPERIMENTAL RESULTS

The MICAz is a 2.4 GHz, IEEE 802.15.4 compliant mote used for enabling low-power wireless sensor networks. The RSSI values are taken in real time using MICAz mote. It features a compliant radio which transceivers use in the 2400 MHz to 2483.5 MHz band, offering both high speed (250 kbps) and hardware security (AES-128). The range of the radio is 75m to 100 m outdoors and 20 m to 30 m Indoors. The MICAz MPR2400CA platform provides 4 KB of RAM, 128 KB of program flash memory and 512 KB measurement (serial) flash memory. It is very energy efficient with current draw of 8 mA in active mode and less than 15 μA in sleep mode.

The MICAz is capable of running TinyOS 2.1.2, which we use to program the MICAz motes to get the desired RSSI Values. The experiment is performed in an indoor environment. The base station and MICAz motes are shown in Fig 2a and 2b, respectively, while the layout of experimental premises.

The base station is positioned at the lobby to generate log files having RSSI values in dBm of each MICAz mote. Three MICAz motes move randomly in the lobby, halls and labs to generate RSSI values and send their respective RSSI values to the static base station. The MICAz motes do not cross each other. The orientation of the MICAz motes is kept in a way as shown in Fig 1. The RSSI values are plotted in Fig 4 and 6 with a gain provided to all RSSI values received in order to make them positive.

1. DATA PROVENANCE

Data provenance has been achieved using the same data received at the base station as described in subsection IV-A. Simulation is performed for two cases. They are as under:

Case 1: No forging of data

The first case is when the packet is transferred from IoT node 1 to IoT node 3 via IoT node 2, IoT node 1 attaches the encoded link fingerprint to the header and sends it to IoT node 2 attaches two encoded link fingerprints to the header. One of link A and other of link B as shown in Fig 1. IoT node 3 upon receiving the packet adds its encoded link fingerprint to the packet. When data provenance has to be performed, the packet header is decoded in sequence at the server. Firstly, the last inserted packet is decoded with the key associated with IoT node 3 and link fingerprints are compared with all the available link fingerprints received from IoT node.

Case 2: Packet is forged at the node level. This case represents a situation when packet is forged at IoT node 1 and is received at IoT node 3 via IoT node 2. The process described in case 1 of subsection IV-B2 is applied by decoding the header in sequence with the key of that IoT node and comparing it with all the available link fingerprints.

2. ADVERSARIAL NODE

The results are achieved by using two methods:

Link fingerprints are observed highly matched all the header data is exhausted origin. Last IoT node N from which the header data is matched.

Link fingerprints showing that the data has been tempered that mismatch occurs

Case 1: IoT network present in no adversarial node

The RSSI variation comparison of link A and link B respectively as shown in Fig 1. IoT node 1 communicating with IoT node 2 and IoT node 2 communicating with IoT node 3 are showing the highly correlated pattern. The correlation coefficients achieved are 0.9270 and 0.8420, respectively. A higher value of 0.9614 and 0.9713 are achieved by applying the filter, which further smooth’s down the RSSI variations. compared to the link fingerprints of IoT node 2. Fig 10 represents the uncorrelated plot for both filtered RSSI variations. The correlation is high between the RSSI variation plots of IoT node 2 and IoT node 2 will be different. The correlation is quite obvious in Fig 8 by observing the relationship in RSSI.
Case 2: Two communicating IoT nodes present in Adversarial node.

As the adversarial node is present between IoT node 1 and IoT node 2, the link fingerprints generated at IoT node 1 and Case 3: Packet tempered at any IoT node. The comparison of RSSI variations is presented in Fig 9. Both IoT node 2 and IoT node 3 send their respective encoded link fingerprints to the server.

Case 4: IoT node is replaced by adversarial node

Link fingerprints mismatch at the server because the RSSI variations comparison is uncorrelated. The reason is that they are connected to the adversarial node. The links are established through the adversarial nodes. We are getting low correlation coefficient.

Case 5: Adversarial node can send its data to the server but cannot access the data present at the server not secure.

In this case, IoT node 1 sends a different link fingerprint.

Case 6: Data using provenance algorithm

The encoded data to the server. The server assumes that the link fingerprint is encoded and decodes it with the key of that node which is replaced by adversarial node. Here after performing multiple experiments, it is observed that the correlation coefficient can be high at times.

3. LINK FINGERPRINT

The standard values specified for MICAz motes are used for energy calculations. Furthermore, the energy benchmarks of MICAz motes used in the literature are applied to the presented protocols. The energy consumption for AES-128 encryption (128 bits), SHA-1 Hash (64 bits), ECDSA-160 Sign and Transmit 1 bit are 1.83 µJ, 154 µJ, 52 µJ and 0.6 µJ respectively. As the decoding is carried out at the server, the energy calculations are not done for the server. The server is not energy limited. Two scenarios are presented:

1. After every 5 minutes and 20 seconds, each IoT node sends its respective quantized and encoded RSSI values of 16 bytes to the server.

2. IoT nodes add certain bytes as headers to the payload which contain encoded link fingerprints.

<table>
<thead>
<tr>
<th>System</th>
<th>Packet (Bytes)</th>
<th>Energy Dissipated (µJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our</td>
<td>32 (max.)</td>
<td>52.773</td>
</tr>
<tr>
<td>Level Crossing [7]</td>
<td>52</td>
<td>53.305</td>
</tr>
<tr>
<td>Ranking [7]</td>
<td>598</td>
<td>66.450</td>
</tr>
</tbody>
</table>

TABLE 3: Energy dissipation

<table>
<thead>
<tr>
<th>IoT Node</th>
<th>Energy Dissipated (mJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(123)</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>104.406</td>
</tr>
<tr>
<td>2</td>
<td>104.814</td>
</tr>
<tr>
<td>3</td>
<td>104.406</td>
</tr>
<tr>
<td>Network</td>
<td>312.626</td>
</tr>
</tbody>
</table>

TABLE 4: Energy dissipated each IoT node Network

XI. CONCLUSION

A light-weight solution for the security and data provenance in IoT environment in is proposed. The energy calculations show that less energy is consumed by applying link fingerprint generation protocol. The fingerprints generated between any two connected IoT nodes are highly correlated. Introducing an adversarial node gives very low correlation coefficient. It means that the detection of any adversarial node in an IoT network can be done for low power nodes. The data forensics can also be applied by looking at the header of the last received data. The origin of data is computed by extracting the header. Time complexity of the system remains the same no matter how lengthy the code becomes.

XII. REFERENCE


