Implementation of Link Fingerprint Verification and Data Provenance using RSSI

Ch. Venkateswarlu, P. Uma Maheswari

Abstract: Due to constrained assets and adaptability, security protocol for Internet of Things (IoT) should be light-weighted. The cryptographic solutions are not possible to apply on little and low-power devices of IoT in view of their power and space impediments. In this paper, a light-weight protocol to verify the information and accomplishing information provenance is introduced for multi-hop IoT arrange. The Received Signal Strength Indicator (RSSI) of conveying IoT nodes are utilized to produce the connection fingerprints. The connection fingerprints are coordinated at the server to process the relationship coefficient. Higher the estimation of connection coefficient, higher the rates of verified information move. Lower worth gives the recognition of ill-disposed node in the middle of a particular connection. Information provenance has additionally been accomplished by examination of pocket header with all the accessible connection fingerprints at the server. The time unpredictability is processed at the node and server level, which is O(1). The power scattering is determined for IoT nodes and overall network. The outcomes demonstrate that the power utilization of the framework and time complexity. Exploratory outcomes show that up to 97% connection is accomplished when no attacker node is available in the IoT network.

Keywords: Internet of Things, Access Control Scheme, cryptography.

I. INTRODUCTION

Internet of Things (IoT) contains an multifaceted system of smart gadgets, which very often trade data through the Internet [1]. IoT has turned into the need for the future correspondence. It is assessed that 50 billion smart gadgets will be associated through IoT by 2020 [2]. The data of a patient to a medicinal staff, vehicle's performance and measurements, home automation, transportation domain, smart networks and smart meters will be founded on IoT. The information procured from sensors or IoT nodes is spread to Internet cloud where it acquired by the concerned body. The obtained information should be exact and ought to have the data about its origin.

As the quantity of nodes are enormous in number, lower in magnitude and generally open, the measures ought to be taken to ensure that the data is verified and proficiently acknowledged at the terminal. Information security and provenance go about as support so as to actualize IoT organize due to fact that the IoT nodes are not physically ensured [3]. The data can be simply produced or altered if appropriate security natives are not followed. Security natives incorporate recognition of specific attacks, concealing channel state, interruption discovery, area differentiation what's more, data provenance.

Revised Manuscript Received on November 10, 2019.

Ch. Venkateswarlu, Assistance Professor, Department of Computer Science and Engineering, Visvodaya Engineering College, Kavali, (A.P.) India.

P. Uma Maheswari, Department of Computer Science and Engineering,, PBR Visvodaya Institute Of Technology And Science, Kavali, (AP), India

Provenance is to obtain the root of the data. A solitary change in information may cause enormous issues e.g., as far as medical- health report created by an IoT node sent to a specialist, meter-reading sent to the organization for charging, as per the utilization and change in transportation framework data [1]. In this way, the conventional cryptographic procedures are not the practical solution in IoT as a result of the energy restrictions of the IoT nodes [4]. Less space gaining furthermore, energy- proficient security natives with less computational complexities are structural blocks for empowering end-to-end content security, client authentication, and customer privacy in the IoT world [2].

To guarantee the trust of clients, the IoT-based system ought to be verified enough. The security mechanism included ought to be light-weighted due to the alleviated energy necessities for IoT nodes [5]. The joint validation between IoT nodes with the server ought to be verified and credible [3]. Accurate and protected data provenance in the IoT are utilized for enhancing the degree of trust. The data provenance is helpful for obtaining and depicting the derived history of data that begins from the resource. The records can be utilized to secure Intellectual-property and its pertinence from the point of view of regulatory-mechanisms. In any case, the data provenance integrity is a central issue. The data provenance can be imitated or altered by an unapproved party if the provenance isn't appropriately ensured by executing unproductive security methods. So as to build up the trust of IoT, a solution for security ought to be planned which is light-weight and profoundly verified [6]. The majority of the security algorithms and cryptography procedures utilized today comprises of huge computational complexity with heightened energy utilization.

The resolution projected in this article joins lightweight security algorithms for verified IoT-based data trade without utilizing additional equipment. contentious node is recognized successfully by associating the link fingerprints produced by the neighbouring IoT nodes. The correlation-coefficient is analyzed at server. Data provenance is additionally accomplished utilizing a similar link fingerprints created to obtain the intrusion recognition in the IoT network. Henceforth, fingerprints are utilized to verify the integrity of information and in the discovery of intrusion. The projected solution has mitigated time complexity contrasted with other best resolutions. The power computations are exhibited too demonstrating alluring outcomes when contrasted with the previously existed work in [7].

II. RELATED WORK

Because of versatility of IoT gadgets, it is hard to ensure them. That is the reason they are inclined to attacks [3]. The scientific classification of assaults in IoT are satirizing, modifying, replaying directing data, Sybil attack[8], Denial of Service (DoS) assaults [9], assaults dependent on node property, assaults dependent on combatant position, assaults dependent on combatant area and assaults dependent on data damage level [1] and so forth. So as to handle these assaults, a required arrangement should be light-weighted and protected down to get the trust of IoT clients [10]. A cryptographic key for secure the IoT system is given utilizing Advanced Encryption Standard (AES)- 128 Algorithm and Inverse AES-128 Algorithm [11].These solutions contains extreme cryptography and computational complexities.

Hence AES-128 algorithm isn't appropriate for IoT considering an enormous number of IoT nodes. Dealing with the shared authentication among RFID labels in IoT, analysts presented a light-weight protocol by encryption technique dependent on XOR function, rather than complex encryption, for example, utilizing the hash -function, for anti-replicate and security protection [12]. In unbounded RFID, the invader can replicate the Electronic Product Key (EPC) of the expected tag and program it to another tag. Physical Un-clonable Functions (PUFs) are utilized at the terminal of node to shield it from the aggressor to gain authorization to the data gathered in the node memory. PUFs might be utilized to give security in IoT frameworks without the requisite to store confidential in the nodes [13]. For interaction purposes, a light-weight messaging protocol known as MQ Telemetry Transport (MQTT) can be utilized. A unified "specialist" is used to interact with terminals. MQTT specialist controls the sort of data shared among terminals, which serves to ensure the security. Elliptic Curve Cryptography (ECC) is too preferable since it can ensure an equal quantity of security with less computation- power and bandwidth capacity than its Rivest, Shamir, and Adelman (RSA) partner [14]. In certain papers, the idea of common trust between security frameworks on IoT devices through the implementation of a structure for access control at the node level is analyzed. As per the researchers, trust is built up from the creation stage to the operation stage in IoT. This trust emerges through two phenomena's; the production of key and the token key made by the producer [15]. In view of, the new Lightweight Label-Based Access Control Scheme (LACS), the verification of approved haze nodes is accomplished to guarantee the authentication.

In particular, LACS verifies haze node by checking the trustworthiness of the estimated-value of the shared file with embedded label, where just the approved haze node approaches the reserving service [16]. A reliable Internet of Vehicles (IoV) system is proposed in [17]. Both the physical and social layer data are consolidated for acknowledging fast content spread in gadget to-gadget vehicle-to-vehicle (D2D-V2V)- based IoV systems.

In [7] paper, verifying the data provenance is accomplished by utilizing the RSSI values gotten by a static base station and a versatile body-worn gadget. Performed investigations demonstrate that exceptionally associated fingerprints are obtained. After each 10 to 15 minutes, a link finger print of 128 bits is created by utilizing RSSI at base station and body worn gadget. The accumulation and access of data provenance are additionally essential to be a verified procedure. The proposed trust model is portrayed for cloud computing in [6]. High trust can be accomplished utilizing a similar model in IoT condition. Improved power proficiency is accomplished by utilizing Gale-Shapley algorithm which matches D2D pair with Cellular User-Equipments (UEs). Correlation between UEs is broke down by exploiting a game-theoretic methodology.

III. RESEARCH METHOD

At the point when two IoT nodes impart, at that point different measurements like RSSI, Time of Arrival (ToA), phasor data and Error Vector Magnitude (EVM) are utilized to create finger print. Regarding RSSI, there is a direct connection between the RSSI varieties of any associated nodes. This data is useful in producing the link-fingerprints which are vastly related for two associated nodes by analyzing the Pearson correlation-coefficient. The RSSI values are recorded continuously by utilizing MICAz bits. The span of recording RSSI values at each IoT node can be expanded or diminished relying upon the accessibility of capacity to the nodes. Since the IoT nodes are power restricted, reasonable methodology is to take the recording - time enormous yet worthy in a way that the outcomes are not influenced. The accompanying scenarios are considered when accomplishing the investigations and simulations:

1) No contentious node is available in the IoT System
2) Adversarial node is available among the two communicating IoT nodes
3) The packet is imitated or tempered at any IoT node
4) The IoT node is supplanted by antagonistic node
5) The server isn't verified in a manner that contentious node can send its information to the server yet, can't explore the data which is present at the server .
6) Finding the interruption in later data utilizing provenance algorithm

The idea presented in this article guarantees security of IoT organize for all the situations referenced above devouring less power. It utilizes continuous exploratory values. MICAz bits are utilized as IoT nodes.

i. Adversarial Node Detection

In our test, each IoT node records its individual RSSI values after each 20 seconds. The RSSI values obtained are in dBmranging from - 48 dBm to 20 dBm. The signal capacity is determined utilizing Friis -transmission equation which expresses that

$$P_r = \frac{P_t G_t G_r}{L_p} \qquad (1)$$

where, P_r is the acquired power, Pt gives the transmitted-power, , G_r and G_t are the receiving and transmitting antenna's gains, separately and L_p is the path-loss. More the path-loss, less will be the received power and consequently low estimation of RSSI. Path-loss is represented as:

$$L_p = \left(\frac{4\pi d}{\lambda}\right)^2 \qquad (2)$$

where d is the separation between two imparting IoT nodes. is the wavelength which is around 416 m due to that the operating-frequency of MICAz bits is 2.4 GHz. An increase of 50 is given to make every one of the values positive. The subsequent RSSI values are quantized utilizing word-length of 8 bit giving 256 levels (L). The amplitude values are mapped onto a limited set of known-values. This is accomplished by partitioning the separation among minimum and maximum RSSI values into L zones, each of the height , which is given by,

$$\Delta = \frac{P_{r(max)} - P_{r(min)}}{L} \qquad (3)$$

Pr(max) and Pr(min) are the maximum and minimum acquired powers, individually. The midpoint of each zone is appointed a value from 0 to L - 1. Each example falling in a zone is approximated to the value of the midpoint. Each zone is then alloted a 8 bit of word-length. This 8-bit word-length is represents to the link fingerprint (LF). The link fingerprint (every 8-bit parallel stream speaking to RSSI esteem) is then encoded with a 8-bit secret key i.e., K1 for IoT node 1, K2 for IoT node 2 and K3 for IoT node 3.

$$LF_{encoded\ (1\to n)} = LF_{1\to n} \oplus K_i \qquad (4)$$

In 4, speaks to logical exclusive-OR operation, though LFencoded is the encoded link fingerprint. Each IoT node sends LFencoded to the server and keeps a replica of the equivalent with itself. The link fingerprint and the secret-key won't be imparted to some other IoT node. The server is expected as profoundly verified and the information is accumulated after the verification is done. Despite the fact that in one case, it is viewed as that antagonistic node can send its data to the server by supplanting IoT node.

K1, K2 and K3 are available at the server, which are thought to be completely secured. The server unravels all the received encrypted link fingerprints of each IoT node utilizing key related to the concerned IoT node as,

$$LF_{1\to n} = K_i \oplus LF_{encoded\ (1\to n)} \qquad (5)$$

The binary coded link fingerprints are changed over to the individual decimal qualities in dBm and correlation procedure is accomplished by analyzing the Pearson relationship coefficient (). On the off chance that the value is somewhere in the range of 0.8 and 1, at that point it is considered as profoundly corresponded in a multi-hop system. Numerically,

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} \qquad (6)$$

where, cov is the covariance and gives the standard deviation. A computed equation can be written as;

$$r = \frac{\sum_{i=1}^{n}(X_i - \hat{X})\sum_{i=1}^{n}(Y_i - \hat{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \hat{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \hat{Y})^2}} \qquad (7)$$

where X_i and Y_i are the RSSI estimations of the ith bundle got at imparting IoT nodes and X and Y are the separate mean RSSI estimations of a grouping of n packets. The relationship coefficient r restores a value in [-1:1] where 1 shows perfect correlation, 0 demonstrates no correlation, and - 1 demonstrates anti-correlation.

Algorithm 1: Link Fingerprint and Encoding at IoT Node. $i = 1 \to n\ and\ j = 1 \to number\ \ of\ IoT\ Nodes.$

1. Initialize the IoT node
2. Read the RSSI values from adjacent IoT node
3. $RSSI_{new\,[i]} \leftarrow RSSI[i] + gain$
4. $Quantize\ RSSI_{new\,[i]}$
5. $LinkFingerprint[i] \leftarrow$ Assign binary code-word to $Quantize\ RSSI_{new\,[i]}$
6. $RSSI_{en\,[i]} \leftarrow XOR(LinkFingerprint[i], Key_{node\,(j)})$
7. $RSSI_{en\,[i]}$ bundled up with session identifiers
8. $Keep\ a\ copy\ at\ the$ IoT Node
9. $Send\ a\ copy\ to\ the$ Server

Algorithm 2: Adversarial node's detection at the server. ρ is the Pearson correlation coefficient having values between -1 and 1.

1. $LinkFingerprint[i] \leftarrow XOR\ (RSSI_{en\,[i]}, Key_{node\,(a)})$
2. $RSSI_{new\,[i]}$ ← bin-dec conversion $(LinkFingerprint[i])$
3. $LinkFingerprint[i] \leftarrow XOR(RSSI_{en\,[i]}, Key_{node\,(a)})$
4. $RSSI_{new\,[j]} \leftarrow$ bin-dec conversion $(LinkFingerprint[j]))$
5. $\rho(RSSI_{new\,[i]}, RSSI_{new\,[j]})$
6. if $0.9 < \rho \le 1$ then
7. $retrun$ No adversarial node is present
8. $else\ if\ \rho = -1\ to\ 0.9\ then$
9. $return$ Adversarial node is present
10. $else$
11. $return$ the $RSSI$ values are not correctly measured
12. $end\ if$

The server connects the LFs of nearby IoT nodes. They are exceptionally related if there is no inclusion of any antagonistic node in the IoT network. In the event that any antagonistic node separates IoT node 1 and IoT node 2 then the link fingerprint gotten by IoT node 1 is not quite the same as link fingerprint acquired by IoT node 2. An exceptionally uncorrelated Pearson correlation coefficient is analyzed. The decryption is done at the server utilizing the keys officially present at the server. Algorithm 1 and 2 provides the recognition of adversarial node's existence in IoT System.

ii. Data Provenance

For data provenance, header data is utilized to arrive at the cause from which the information is started. As talked earlier, each IoT node sends the duplicate of the link fingerprints to the server, so all the header data will as of now be available at the server. On the off chance that the data is obtained at IoT node 3 from IoT node 1 by means of IoT node 2, the link -fingerprints of header are contrasted at the server in succession and duplicates of link fingerprints beforehand sent by the IoT nodes. From whichever IoT node the last header data coordinates, the information is originated from that IoT node. Size of header relies upon the choice of packet size. Here, the header size is 16 bytes. Algorithm 3 depicts the data provenance where the IoT nodes are associated with one another. Each IoT node connects the encoded link-fingerprint

Algorithm 3: Data Provenance

1. $for\ Header_i, i = n \rightarrow 1\ do$
2. //n is the last IoT node the packet is received at $LinkFingerprintHeader_i =$ $XOR\ (Header_i, Key_i)$
3. Correlate $LinkFingerprintHeader_i$ with copy of link fingerprintd received from IoT $node\ [i]$
4. if Correlation > 95% then
5. $return\ i \leftarrow i - 1$
6. $else$
7. Data forged between IoT $node\ [i]$ and IoT $node\ [i-1]$
8. $end\ if$
9. $end\ for$
10. The origin of the packet is IoT $node\ [i]$

as header to the packet it gets and advances it to the following IoT node. Toward the end, the concerned node after getting the packet includes its link fingerprint as header and simply like some other IoT node, it sends it to the server. The server knows the size of header that each IoT node connects and the nearby IoT nodes of each IoT node. So as to check the evolution from which the information is generated, server deciphers the header with the keys present at the server and relates the connection unique mark with the effectively present link- fingerprints got from that node. When the link-fingerprints coordinate, a similar procedure is rehashed for the neighbouring IoT node(s). The procedure proceeds until;

1) Profoundly coordinated link-fingerprints are watched and all the header information is depleted. The source is the last IoT node from which the header information is coordinated.

2) Disparity occurs in link fingerprints displaying that the data has been tempered at that node.

While finding the source of information, if adversarial node is available between any two IoT nodes and the packet moves via pugnacious node then the server will yet get high connected outcome by contrasting the link- fingerprints. The link fingerprints will coordinate the link-fingerprints present at the server got from the IoT node. The reason is that in the event that we consider the referenced circumstance in Fig 1, the adversarial-node is between IoT node 1 and IoT node 2, the IoT node 1 includes the link-fingerprint at the header which is of the connection between IoT node 1 and pugnacious node. So also, IoT node 2 includes the link-fingerprint of the connection between antagonistic node and IoT node 2 to the packet header obtained from pugnacious node and advances it. The last IoT node on acquiring it, includes its link-fingerprint. The server validates the header for the source and gets high related value in the wake of decrypting the header embedded by IoT node 2. The source can at present be estimated regardless of whether the antagonistic node is available between them. In spite of the fact that the link-fingerprints of IoT node 1 and IoT node 2 will be exceptionally uncorrelated.

As IoT nodes will be enormous in number, the physical security won't be workable for a majority of the nodes. The information can be effectively replicated or tempered. On the off chance that the information is tempered at IoT node 2 and sent to IoT node 3 a short time later, the data provenance can't be accomplished rather the adversarial-node's inclusion can be identified. The procedure can tell precisely between which link the information has been produced.

This is a helpful data in information criminology. The extremely uncorrelated outcome is accomplished when contrasting the link- fingerprints in the header and the ones present at the server. Algorithm 3 represents the accomplishment of data provenance.

IV. RESULT ANALYSIS

A) Adversarial Node Detection

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

Significant improvements in results are seen by filtering out the RSSI variations. The comparative results are shown in Table 1.

Table 1: Pearson correlation coefficient (r) calculated for various cases

Scenario	IoT node 1 and IoT node 2		IoT node 2 and IoT node 3		Confidence Interval (CI)
	r	filtered r	r	filtered r	
Case 1	0.9270	0.9614	0.8420	0.9713	95%
Case 2	-0.0038	0.0287	0.9280	0.9515	95%
Case 3	0.8913	0.9628	0.0628	0.2056	95%
Case 4	-0.0063	-0.3693	-0.1740	-0.5125	95%
Case 5	-0.2753	-0.3384	0.8369	0.9520	95%
Case 6	0.8382	0.8590	0.5269	0.7643	75%

B) Data provenance

Data provenance has been accomplished utilizing similar information obtained at the base station. Simulation is performed for two cases. They are as under,

Case 1: No forging of data

The primary case is the point at which the packet is moved from IoT node 1 to IoT node 3 by means of IoT node 2, IoT node 1 articulates the encrypted connection link fingerprint to the header and transfers it to IoT node 2. IoT node 2 joins two encrypted link fingerprints to the header.

At the point when data provenance must be accomplished, the packet header is decrypted in succession at the server. At first, the last embedded-packet is decrypted with the key related with IoT node 3 and link -fingerprints are contrasted and all the accessible link fingerprints obtained from IoT node 3. The simulations have demonstrated that the match is 100% with a part of all the accessible link fingerprints of IoT node 3. At that point the contiguous nodes are checked. As the adjoining node is IoT node 2, so the following sequence of packet is decrypted with K2 and 100% match is identified at some piece of all accessible link-fingerprints from IoT node 2. Presently the contiguous nodes are checked once more. IoT node 2 associated with IoT node 1 and IoT node 3 associated with IoT node 2 are in the nearness list. Both are checked and 100% match is found with a part of all link-fingerprints present at the server received from IoT node 2 connected with IoT node 1. Presently a similar procedure is accomplished for the following in grouping of header. A 100% match in link fingerprints from the header with part of IoT node 1's link-fingerprints is accomplished. At this point, all the header-sequences are checked and no header information is left to discover a match for. The last header is the first embedded header from ioT node 1 which is obtained at IoT node 3 at last. Table 2 demonstrates the outcomes obtained.

Case 2: Packet is forged at the node level

This case represents a circumstance when packet is replicated at IoT node 1 and is obtained at IoT node 3 via IoT node 2.

Table 2: Data Provenance

Scenario	Correlation of IoT node header with all available LFs at the server				Remarks
Case 1	100%	100%	100%	100%	The origin is IoT node 1
Case 2	100%	100%	100%	40.7407%	The data is tempered at IoT node 1

C) Time Complexity

The time -complexity correlation is accomplished via figuring the computational delay at node and server level. As appeared in Fig 13, the time stays steady if we increment the quantity of RSSI tests to be quantized at node level and for connection at the server. This demonstrates as the quantities of bits are heightened the computational time isn't influenced. The time complexity of our framework is O(1) which is as better contrasted with other best in class cryptographic arrangements alluded at [21]. Table 3 demonstrates the differentiation of time intricacy of our model with other accessible data security algorithms.

Table 3: Time complexity comparison of our system with various state of the art algorithms for data security.

AES	DES	3DES	RC4	BlowFish	Our System
O(N)	O(N)	O(N)	O(N)	O(N)	O(1)

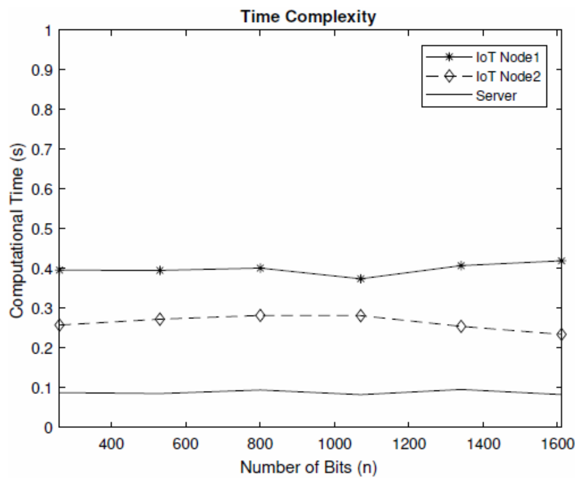

Fig. 13: Time complexity of system at node and server level

D) Energy Consumption

In this segment, power utilization is determined for the framework model introduced. The particulars of MICAz bits are as of now introduced. The standard qualities determined for MICAz bits are utilized for power estimations. Besides, the power benchmarks of MICAz bits utilized in the literature are connected to the displayed protocols. The energy utilization for AES-128 encryption (128 bits), SHA-1 Hash (64 bits), ECDSA-160 Sign and Transmit 1 part are 1.83 J, 154 J, 52 J and 0.6 J individually [7]. As the decrypting is completed at the server, the power figurings are not accomplished for the server. The server isn't power restricted. Two situations are introduced:

1) After every 5 minutes and 20 seconds, each IoT node transmits its corresponding quantized and encrypted RSSI values of 16 bytes for the server.

2) IoT nodes provide few bytes as headers to the payload which comprises of encrypted link -fingerprints.

Table 4 and 5 discloses the power exploitation at each node level when the packet is transmitted from IoT node 1 to IoT node 3 through IoT node 1 according to the hash data and session identifiers as part of protocols used conventionally in [7].

Table 4: Energy consumption at IoT node level for link fingerprints transmission to the server

IoT node (1,2,3)	Fingerprint (bytes)	Transmission Cost (μJ)	AES-128 (μJ)	SHA-1 (μJ)	ECDSA-160 (mJ)	Total (mJ)
1	16	76.8	1.83	308.0	52	52.386
2	32	153.6	3.66	616.0	52	52.773
3	16	76.8	1.83	308.0	52	52.386
Total Energy dissipated at node level						157.545

Table 5: Energy consumption at IoT node level for data provenance protocol

IoT node (1,2,3)	Fingerprint (bytes)	Transmission Cost (μJ)	AES-128 (μJ)	SHA-1 (μJ)	ECDSA-160 (mJ)	Total (mJ)
1	4	19.204	0.568	0.768	52	52.020
2	8	38.408	1.144	1.536	52	52.041
3	4	19.204	0.568	0.768	52	52.020
Total Energy dissipated at node level						156.082

By implementing several optimization-methods, the link fingerprint can be decreased.

V. CONCLUSION

The fingerprints created between any two associated IoT nodes are profoundly corresponded. Presenting an adversarial node gives exceptionally low correlation-coefficient. It implies that the detection of any adversarial-node in an IoT system should be possible for low power nodes. The data-forensics can likewise be connected by probing at the header of the last obtained data. The source of information is processed by extricating the header. The server is considered as exceptionally secured since it comprises the keys related with all the IoT nodes. The power calculations demonstrate that less power is devoured by applying the link fingerprint generation protocol, transmitting the packet to the server and to the contiguous IoT node. Time intricacy of the framework remains the same no matter how extended the code becomes.

REFERENCES

1. Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, “A survey on security and privacy issues in internet-of-things,” IEEE Internet of Things Journal, vol. 4, no. 5, pp. 1250–1258, Oct 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7902207/
2. S. Satpathy, S. Mathew, V. Suresh, and R. Krishnamurthy, “Ultra-low energy security circuits for iot applications,” in IEEE 34th International Conference on Computer Design (ICCD), 2016, pp. 682–685.
3. M. N. Aman, K. C. Chua, and B. Sikdar, “Mutual authentication in iot systems using physical unclonable functions,” IEEE Internet of Things Journal, vol. 4, no. 5, pp. 1327–1340, Oct 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7924368/
4. T. Idriss, H. Idriss, and M. Bayoumi, “A puf-based paradigm for iot security,” in IEEE 3rdWorld Forum on Internet of Things (WF-IoT), 2016, pp. 700–705.
5. S.-R. Oh and Y.-G. Kim, “Security requirements analysis for the iot,” in International Conference on Platform Technology and Service (PlatCon), 2017, pp. 1–6.
6. M. I. M. Saad, K. A. Jalil, and M. Manaf, “Achieving trust in cloud computing using secure data provenance,” in IEEE Conference on Open Systems (ICOS), 2014, pp. 84–88.
7. S. T. Ali, V. Sivaraman, D. Ostry, G. Tsudik, and S. Jha, “Securing first-hop data provenance for bodyworn devices using wireless link fingerprints,” IEEE Transactions on Information Forensics and Security, vol. 9, no. 12, pp. 2193–2204, Dec 2014. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6898844
8. K. Zhang, X. Liang, R. Lu, and X. Shen, “Sybil attacks and their defenses in the internet of things,” IEEE Internet of Things Journal, vol. 1, no. 5, pp. 372–383, Oct 2014. [Online]. Available: http://ieeexplore.ieee.org/document/6868197/
9. G. Kecskemeti, G. Casale, D. N. Jha, J. Lyon, and R. Ranjan, “Modelling and simulation challenges in internet of things,” IEEE Cloud Computing, vol. 4, no. 1, pp. 62–69, Jan 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7879128/
10. J. Pacheco and S. Hariri, “Iot security framework for smart cyber infrastructures,” in IEEE International Workshops on Foundations and Applications of Self* Systems, 2016, pp. 242–247.
11. Z. Bohan, W. Xu, Z. Kaili, and Z. Xueyuan, “Encryption node design in internet of things based on fingerprint features and cc2530,” in IEEE International Conference on Green Computing and Communications (GreenCom), Internet of Things, and IEEE Cyber, Physical and Social Computing (iThings/CPSCom), 2013, pp. 1454–1457.
12. J.-Y. Lee, W.-C. Lin, and Y.-H. Huang, “A lightweight authentication protocol for internet of things,” in International Symposium on Next-Generation Electronics (ISNE), 2014, pp. 1–2.
13. M. N. Aman, K. C. Chua, and B. Sikdar, “Secure data provenance for the internet of things.” ACM Press, 2017, pp. 11–14. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3055245.3055255
14. J. Qian, H. Xu, and P. Li, “A novel secure architecture for the internet of things.” IEEE, Sep 2016, pp. 398–401. [Online]. Available: http://ieeexplore.ieee.org/document/7695208/
15. Y. Xie and D. Wang, “An item-level access control framework for intersystem security in the internet of things,” vol. 548-549, Apr 2014, pp. 1430–1432. [Online]. Available: https://www.scientific.net/AMM.548-549.1430
16. Q. Wang, D. Chen, N. Zhang, Z. Qin, and Z. Qin, “Lacs: A lightweight label-based access control scheme in iot-based 5g caching context,” IEEE Access, vol. 5, pp. 4018–4027, 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7872420/
17. Z. Zhou, C. Gao, C. Xu, Y. Zhang, S. Mumtaz, and J. Rodriguez, “Social big-data-based content dissemination in internet of vehicles,” IEEE Transactions on Industrial Informatics, vol. 14, no. 2, pp. 768–777, Feb 2018. [Online]. Available: http://ieeexplore.ieee.org/document/7995077/
18. Z. Zhou, K. Ota, M. Dong, and C. Xu, “Energy-efficient matching for resource allocation in d2d enabled cellular networks,” IEEE Transactions on Vehicular Technology, vol. 66, no. 6, pp. 5256–5268, Jun 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7585029/
19. Z. Zhou, H. Yu, C. Xu, Y. Zhang, S. Mumtaz, and J. Rodriguez, “Dependable content distribution in d2d-based cooperative vehicular networks: A big data-integrated coalition game approach,” IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 3, pp. 953–964, Mar 2018. [Online]. Available:

http://ieeexplore.ieee.org/document/8264740/

20. Micaz-Wireless Measurement System, Crossbow Technology, 4 2007.

AUTHORS PROFILE

Ch. Venkateswarlu has received his M.C.A from S.U.V, Tirupati and M.Tech degree in Computer science and Engineering from JNTU, Anantapur respectively. He is dedicated to teaching field from the last 9 years. He has guided 10 P.G and 20 U.G students. His research areas included Data Mining. At present he is working as Assistant Professor in Visvodaya Engineering College, kavali, Nellore(D.t), Andhra Pradesh, India.

P. Uma Maheswari has received her B.Tech degree in CSE from MERITS JNTU, Anantapur in 2012 and pursuing M.Tech degree in CSE from PBR VITS, affiliated JNTU, Anantapur in 2019.