CONTINUOUS SECURITY IN IOT USING BLOCKCHAIN

Rahul Agrawal, Pratik Verma, Rahul Sonanis, Umang Goel, Dr. Aloknath De,
Sai Anirudh Kondaveeti, Suman Shekhar

Samsung Research Institute Bangalore
{rah.agrawal, pratik.verma, sonanis.r, umang.goel, aloknath.de}@samsung.com

ABSTRACT

The two major roadblocks for state of the art Internet of Things (IoT) infrastructure like smart buildings, smart cities, etc. are lack of trust between various entities of system and single point of failure which is a vulnerability causing extreme damage to the whole system. This paper proposes a blockchain based IoT security solution where, trust is established through the immutable and decentralized nature of blockchain. The distributed nature of blockchain makes the system more robust and immune to single point of failure. We propose a mechanism to establish continuous security in the system by evaluating legitimate presence of user in valid IoT-Zone continuously without user intervention. Every user interaction in an IoT environment is stored in blockchain as a transaction and series of these transactions represent a user's IoT-trail. A unique digital crypto-token is required for a user interaction to be legitimate. This token is used as an access control mechanism to prevent any unauthorized access to the system. Tokens are pre-generated using a prediction model based on user's IoT-trail in the blockchain. By using blockchain as an underlying framework in IoT environment and through the method of continuous security, we made the system more secure, robust and interoperable.

Index Terms— Internet of Things, blockchain, continuous security, digital crypto-token

1. INTRODUCTION

Internet of Things (IoT) system is a complex environment where multiple entities and devices interact with each other [1]. The current system is a trust-dependent, centralized cloud based model which limits the interoperability due to varying configurations of multiple clouds and devices. Also the security mechanism in current infrastructure is only limited to discrete security leaving many vulnerable entry/exit points in IoT systems [2]. Due to involvement of multiple stakeholders in IoT, it becomes essential to build trust-less, preventive, decentralized and interoperable systems to make IoT systems practical. Hence, the need to exploit blockchain's inherent characteristics is evident in future of IoT Infrastructure.

The immutable and distributed nature of blockchain [3] brings legitimacy to any interaction in the IoT system. All these interactions are stored as transactions which will further be used in preventing any suspicious interactions. In this paper we propose a framework which uses a crypto-token [4] based blockchain to provide continuous security. A transaction in blockchain can represent various interactions like movement of user between IoT-Zones (e.g. home, office, etc.), secure transfer of data between devices and users, user-device activity (e.g. access to office) in a smart city/building, multiple organizations working in cohesion to provide improved service to user (health data securely shared between hospital and diagnostic centre), device to device activity etc [5].

In this paper, we focus on the security of the user-device interactions. The existing solutions for user authentication primarily use One Time Password (OTP), passwords or static security questions but are limited to one-time validity of credentials [6]. However legitimacy of path followed (transactions done) prior to providing credentials can be used to establish much more confidence in user's identity. This paper is structured as follows. In section 2, the key concepts of our solution are briefly introduced. In section 3, the methodology for continuous security which contain details about zone-identification, token generation and token validation is described. Section 4 presents the simulation results and the performance of the developed system and concludes with the impact of our solution on IoT systems.

2. KEY CONCEPTS

2.1. Blockchain Infrastructure

An application has been built on top of Hyperledger Fabric [7] framework to store every user-device interaction on blockchain through a consensus of nodes. These interactions are stored as transactions. A sequence of these transactions define a user's trail of user-device activities in the system. The IoT system will not be compromised in the event of single point failure [8] due to de-centralized nature of blockchain. The IoT system may contain constrained devices which don't have enough computational power or the storage to be full

nodes in blockchain network. Hence it is assumed that such devices will be interacting through an IoT-hub which, being an unconstrained device will act as a node in Blockchain network.

2.2. Continuous Security

Along with PINs/Passwords wherever required, every interaction is also mediated via crypto-token which can only be used by the legitimate user. The generation of crypto-token depends on the current user state and user's possible actions. Token will not be generated in case of suspicious actions as shown in Figure 1. This facilitates secure and seamless transitions between IoT-zones.

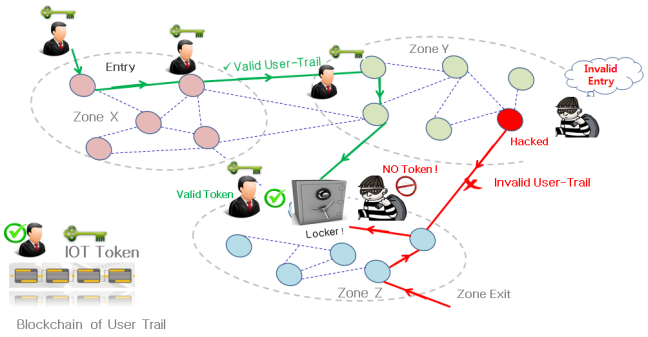

Fig. 1. High Level System Overview

3. METHOD FOR CONTINUOUS SECURITY

Continuous security is achieved primarily through IoT-Zone identification, IoT-Token generation for next valid zones and Token validation which are described in detail in following sub-sections. A high level flow of these three ideas is captured in Figure 2. The details of user's trail which is used in these ideas can be obtained by querying the blockchain.

3.1. IoT-Zone Identification

IoT-Zone identification needs active monitoring of user IoT-trails. Initial topology of IoT-zone is established based on physical connections (like swipe gates) between multiple zones and set of rules as shown in Figure 3. Initially the state transition probabilities are established through rule based solution.

A learning based path prediction model is used to increase user experience. In this case user's transition in zones is modeled as a random variable. The state-transitions of random variable can be represented via a directed graph with edges having state transition probabilities. For smart-building context, the most probable route of user could be predicted (e.g. entry (Z_1) $\rightarrow$ reception (Z_2) $\rightarrow$ business center (Z_5)). This probability distribution could be learned over a period of time

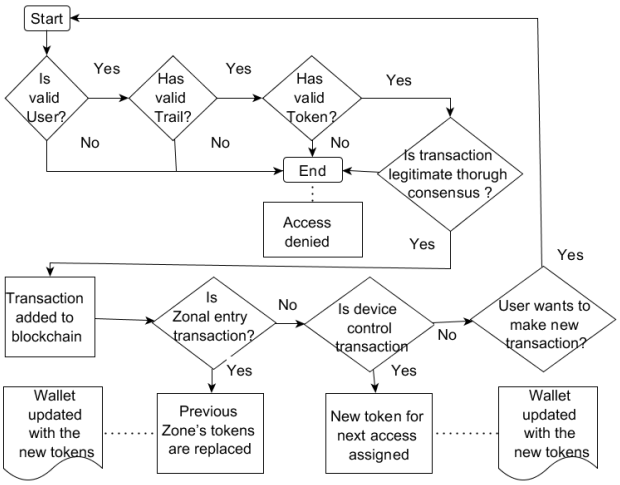

Fig. 2. Abstract Overview of Idea

based on user behavior derived from number of visits. This pattern is not limited to individual characteristic, however it could represent the crowd behavior in general (e.g. employees in smart-building). We use two approaches for learning which are described as follows.

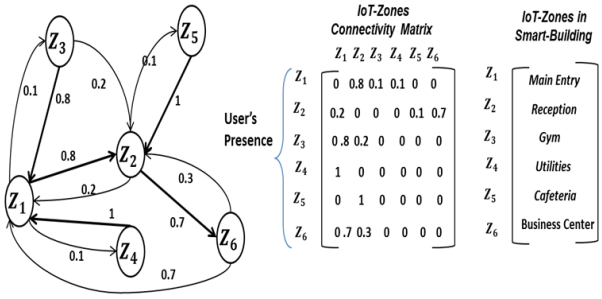

Fig. 3. Rule-Based IoT-Zone Connectivity

3.1.1. *Variable Order Markov Model*

In the base case, time invariant random variable has been modeled. Time dependence may be included via temporal variables, which control the probability distributions. We assume that the next chosen state by the user depends only on the past few states. A threshold Markov model of order n is chosen. Let the next zone (i.e. state) be denoted by Z_{n+1}. Let $\mathbf{z} \equiv Z_1 ... Z_n$ be the previous sequence of up to n zones. Let $N(\mathbf{z}Z_{n+1})$ denote the number of occurrences of the subsequence of zones encountered in the training sequence of zones. Let

$$\sum_s = \{\sigma : N(\mathbf{z}\sigma) > 0\}$$

then, the conditional probability estimator is given by

$$\widehat{P_n}(Z_{n+1}|\mathbf{z}) = \frac{N(\mathbf{z}Z_{n+1})}{\sum_{Z \in \sum_s} N(\mathbf{z}Z)}$$

For implementing the n bounded Markov model, data-structure "*trie*" $\boldsymbol{T}$ is used [9] as shown in Figure 4. Each node of $\boldsymbol{T}$ represents a zone and has a counter for number of visits made so far. Initially during training phase, the "*trie*" is initialized with valid nodes based on history of user's trail.

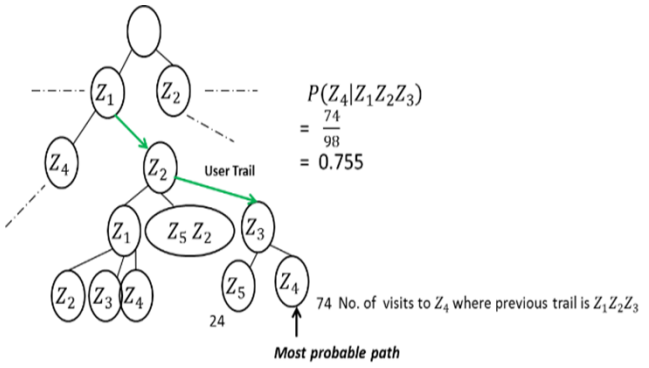

Fig. 4. A "*trie*" corresponding to a sample third order model

3.1.2. Recurrent Neural Network

The second approach for prediction is based on Long Short-Term Memory (LSTM, a recurrent neural network architecture). A LSTM cell [10] enables the network to remember previous trends for predicting future output. This approach first trains a neural network for each user based on his/her previous trail. Currently the previous trail is used on the feature vector. The prediction problem for the next state has been handled using a multiclass prediction model. Each of the next possible states is considered a class. Given the previous sequence window, the classifier predicts a class which corresponds to the predicted next state. Depending on the problem, the LSTM may skew towards long term or short term memory as needed.

3.2. IoT-token Generation

Every user in the IoT system has to register with Enrollment Certificate Authority (ECA) which provides an Enrollment Certificate (ECert) to the user. User's public key is used to gather transactions from blockchain establishing his/her trail. GPS system and locations of surrounding IoT-devices in the network along with mined user trail helps in establishing his current zone. User's attributes like permission level, organization etc. is checked with ECA to verify his/her authenticity. Next possible zones are identified using trail. Based on the transition probabilities, tokens are generated for high probability zone devices whereas for low probability zones, a 2^{nd} step of authentication (fingerprint scan, PIN/PASSWORD) is required. The transition probabilities are learnt using Variable Order Markov Model (VMM) and stored in "*trie*" data structure. Next probable transition to IoT-Zones is predicted using refined zone connectivity matrix and n^{th} order prediction (lookback n steps). For some specific situations like an emergency, corresponding tokens are assigned to all users at the time of on-boarding. These tokens are enabled in the system only when the corresponding situation occurs. For example emergency exit tokens will be enabled in case of fire.


```
{
      "timestamp": "1572042",
      "userName": "Charlie",
      "currentLocation": "Floor 4",
      "userTrail": ["Parking",
                    "Entrance",
                    "Cafe"]
      "designation": "employee",
      "validityPeriod": "28192",
      "ECert": "MIIBrjCC....CQ"
}
```


Fig. 5. Crypto token JSON object

Tokens generated for next possible zones/devices are stored in a digital repository also known as wallet [11]. Wallet also stores the private key of the user which is required to sign a transaction. The generated tokens are stored as JSON objects (Figure 5) in transaction which are later verified to execute a transaction.

3.3. IoT-token Validation

Once IoT-token has been generated, an action is triggered which is being analyzed by a nearby IoT-hub. IoT-hub queries the particular token from digital wallet of nearby user devices via API and verifies it with the help of blockchain network. Each user has unique private-public key pair. Token is signed using RSA Digital Signature algorithm [12] by user's private key. Digital Signature is verified first on blockchain network using user's public key. This ensures that token is not used by another user in the network in case of token theft. If IoT-Tokens are authentic, the user is granted access to control associated devices in IoT-Zones. After validation, the token is included as part of a transaction which gets added in the blockchain network using Practical Byzantine Fault Tolerance (PBFT) consensus [13]. If transaction addition was successful, then current tokens get disabled and user's wallet is updated with new IoT-tokens for next possible zones. The prediction model parameters are updated accordingly.

4. RESULTS AND CONCLUSION

4.1. Dataset

We have used dataset of multiple users transitioning between zones in a smart building. For every user on an average, 1000 data sequences were used. Zones are mapped to floors of the building. Inter-floor transitions were considered for simplicity of the problem. Intra-floor transitions are ignored as transitions within a floor are considered inconsistent.

4.2. Test Setup

The Blockchain framework used is Hyperledger Fabric v0.6 by IBM. The Blockchain network comprises of 4 validating nodes and 1 Certificate Authority node. The use-case logic is implemented in Smart Contract over Hyperledger fabric.

4.3. Zone Connectivity Matrix

Zone connectivity matrix is generated (Figure 6) for the rule based method as discussed in section 3.1. Simpler zone connections can be obtained by setting a threshold probability. For the dataset, we have generated zone connectivity graph as shown in Figure 7 for threshold probability of 0.1.

	Z_0	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8	Z_9	Z_{10}	Z_{11}	Z_{12}	Z_{13}
Z_0	0	0.13	0.10	0.02	0.10	0.12	0.12	0.05	0.01	0.09	0.09	0.09	0.04	0.04
Z_1	0.47	0	0.12	0.22	0.03	0.02	0.02	0.03	0.02	0.01	0.02	0.03	0.01	0.01
Z_2	0.59	0.12	0	0.08	0.07	0.02	0	0.03	0.03	0.01	0.02	0.02	0.01	0.01
Z_3	0.44	0.18	0.14	0	0.07	0.01	0.02	0.04	0.03	0.02	0.01	0.01	0.01	0.01
Z_4	0.65	0.07	0.08	0.06	0	0.01	0.01	0.07	0.02	0	0	0.01	0.01	0.01
Z_5	0.67	0.04	0.02	0	0.01	0	0.01	0.04	0.01	0	0.08	0.09	0.01	0.02
Z_6	0.72	0.02	0	0	0.01	0	0	0.02	0.01	0.12	0.01	0.01	0.02	0.06
Z_7	0.55	0.02	0.04	0.02	0.11	0.06	0.01	0	0.03	0.03	0.03	0.02	0.02	0.03
Z_8	0.90	0.01	0	0	0	0	0	0	0	0	0.01	0.03	0.05	0
Z_9	0.71	0.04	0.01	0	0	0.01	0.10	0.06	0.01	0	0	0	0.02	0.03
Z_{10}	0.64	0.03	0.02	0	0.01	0.07	0.01	0.03	0.01	0	0	0.03	0.01	0.14
Z_{11}	0.74	0.05	0.01	0	0.01	0.12	0.01	0	0.01	0	0.03	0	0.01	0.01
Z_{12}	0.92	0	0	0	0	0	0	0	0.07	0.01	0	0.00	0	0
Z_{13}	0.41	0.06	0.02	0.01	0.04	0.04	0.19	0.02	0.05	0.04	0.10	0.02	0.01	0

Fig. 6. Zone Connectivity Matrix

4.4. Discussion

To train the models, 60% of the data were used and the rest 40% were used for calculating the accuracy of the prediction. A comparison of the Markov models in presented in Table 1. It is observed that decreasing the order improves accuracy with a majority of the users, suggesting that for the user zone movement prediction, the immediate history has more correlation with the next step. The number of labels represent the different zones which are frequently visited by the user. As the number of labels increase, the accuracy decreases due to more degrees of freedom for each user. In the general case, this suggests that an application oriented order be used.

To learn more complex relationships, LSTM model with 32 hidden units within 1 layer and learning rate of 0.1 was used whose results are shown in Table 1. In our model, Gradient Descent algorithm is used for optimization. It is observed that, on an average, LSTM model gives less prediction accuracy than Markov model, owing to lesser training data and the inherent first order Markov behavior present in the training set. However, RNN based model will potentially perform better with more diverse datasets with availability of more data. Also parameters like training steps, batch size, learning rate, hidden units, etc. were tweaked for better accuracy.

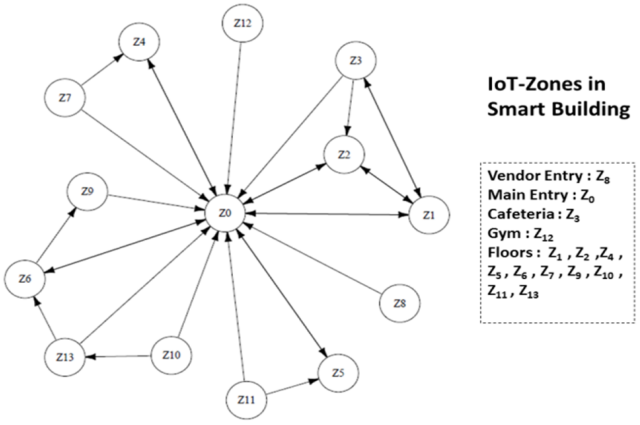

Fig. 7. Zone Connectivity Graph

	Markov			LSTM	
User Id	3^{rd} order	2^{nd} order	1^{st} order	No of labels	Accuracy
1	98.75	99.12	99.30	4	77.77
2	89.65	89.65	93.10	4	71.80
3	43.75	59.37	62.50	5	64.40
4	85.18	88.89	96.30	6	61.50
5	97.88	83.10	66.90	7	60.00
6	53.33	63.33	73.33	5	52.68
7	54.76	64.29	73.81	7	52.08
8	61.19	64.18	65.67	7	46.85

Table 1. Accuracy of Markov and LSTM model for 8 users

4.5. Conclusions

Continuous security is established in the system with seamless user authentication using crypto-tokens. This innovative approach enhances the security of the system without any user intervention as crypto-tokens are pre-generated using various prediction models. Further crypto-token generation can be improvised by using an ensemble learning approach which uses the best models specific to the input data or a weighted combination thereof. Better accuracy is achievable using bigger datasets in LSTM model. This being the first solution of its kind, by incorporating more diverse and big datasets better results could be achieved for commercialization. Moreover, the intangible and unquantifiable security benefits of blockchain will enhance its potential for commercial application in future.

5. REFERENCES

[1] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645 – 1660, 2013, Including Special sections: Cyber-enabled Distributed Computing for Ubiquitous Cloud and Network Services and Cloud Computing and Scientific Applications - Big Data, Scalable Analytics, and Beyond.

[2] S. Sicari, A. Rizzardi, L.A. Grieco, and A. Coen-Porisini, "Security, privacy and trust in Internet of Things: The road ahead," *Computer Networks*, vol. 76, no. Supplement C, pp. 146 – 164, 2015.

[3] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," `https://bitcoin.org/bitcoin.pdf`, 2009.

[4] M. Rosenfeld, "Overview of Colored Coins," `https://bitcoil.co.il/BitcoinX.pdf`, 2012.

[5] K. Christidis and M. Devetsikiotis, "Blockchains and Smart Contracts for the Internet of Things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016.

[6] Cheng Xiao-rong, Feng Qi-yuan, Dong Chao, and Zhang Ming-quan, "Research and realization of authentication technique based on OTP and Kerberos," in *Eighth International Conference on High-Performance Computing in Asia-Pacific Region (HPCASIA'05)*, July 2005, pp. 5 pp.–416.

[7] "Hyperledger Fabric docs v0.6 - whitepaper," `https://media.readthedocs.org/pdf/hyperledger-fabric/v0.6/hyperledger-fabric.pdf`, 2017.

[8] Rodrigo Roman, Jianying Zhou, and Javier Lopez, "On the features and challenges of security and privacy in distributed internet of things," *Computer Networks*, vol. 57, no. 10, pp. 2266 – 2279, 2013, Towards a Science of Cyber Security Security and Identity Architecture for the Future Internet.

[9] Ron Begleiter, Ran El-Yaniv, and Golan Yona, "On Prediction Using Variable Order Markov Models," *Journal of Artificial Intelligence Research*, vol. 22, no. 1, pp. 385–421, Dec. 2004.

[10] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A Search Space Odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, Oct 2017.

[11] "Bitcoin Wallet," `http://www.investopedia.com/terms/b/bitcoin-wallet.asp`.

[12] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978.

[13] Miguel Castro and Barbara Liskov, "Practical Byzantine Fault Tolerance," in *Proceedings of the Third Symposium on Operating Systems Design and Implementation (OSDI '99)*, Berkeley, CA, USA, 1999, pp. 173–186, USENIX Association.