Journal of
Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.7684191

A STUDY TO ANALYSE DETECTING SECURITY ATTACKS ON BLOCKCHAIN USING ANOMALY DETECTION SYSTEM

Sony Kumari[1], Dr. Manoj Eknath Patil[2]
Research Scholar[1], Research Guide [2]
[1,2]Department of Computer Science & Engineering, Dr.A.P.J.Abdul Kalam University,
Indore(M.P)
sony.nayan@gmail.com[1],mepatil@gmail.com[2]

ABSTRACT

Due to the variety of applications it can be used for and the potential for disruption, block chain technology has attracted a lot of attention. One of the main reasons block chain is so popular is due to its distinctive structure: it is described as a computerized log file and kept as a series of connected groups, or blocks, and employs peer-to-peer networks and registers to keep transactions. In this paper, we define an encoder-decoder deep learning model-based anomaly detection system that is trained using aggregate data acquired by observing block chain activity. The effectiveness of our methodology in identifying attacks that have been disclosed publicly has been demonstrated through experiments on the whole history logs of the Ethereum Classic network. As far as we are aware, our strategy is the first to offer a complete and workable method to monitor the security of block chain transactions.

Keywords: Block chain, Attacks, Cyber security, Vulnerability, Anomaly detection

I. INTRODUCTION

Organizations' increased reliance on technology and the internet has opened up new revenue sources, but it has also left those businesses vulnerable to cyber-attacks. As malware evolves into more sophisticated tools, and as the danger of professional cyber groups grows, cyber-attacks have grown more targeted and intricate. Cybercriminals are resorting to highly lucrative strategies such as monetizing data access through the use of advanced ransom ware techniques or by disrupting overall business operations through Distributed Denial of Service (DDoS) attacks in an effort to steal valuable data such as intellectual property (IP), personally identifiable information (PII), health records, and financial data.

By virtue of its immutability, transparency, auditability, data encryption, and operational resilience, block chains may be able to contribute to enhanced cyber defense. This would allow for the platform to secure data, prevent fraudulent activities via consensus mechanisms, and detect data tampering (including no single point of failure).

A growing number of people see block chain technology as the future answer to many thorny record-keeping and financial transaction issues. Many people have difficulty grasping the implementation particulars and possible hazards associated with this cutting-edge technology. Businesses who are thinking in implementing block chain technology and their advisors should:

- Understand basic block chain technology concepts.
- Assess how its cyber risks may apply to them.

- Make reasonable implementation decisions as the technology and its applications mature.

II. BLOCKCHAIN CYBER VULNERABILITIES

Block chain Code Vulnerabilities

Software code flaws can present cyber threats to block chain applications in the same way they might to any other computer system. Errors in the code may be more common in network protocols that use unique or uncommon features whose security implications are yet poorly understood.

Cryptographic algorithms are also crucial to the functioning of block chain technology. The most standard forms of encryption have been thoroughly tested and are universally trusted. Yet as computational methods advance, they could become easier to exploit. Especially with the advent of quantum computing, which uses the peculiar characteristics of quantum particles to execute computations quickly and effectively, present encryption methods may become significantly vulnerable.

Blockchain Platform Vulnerabilities

Block chain applications often operate on general-purpose operating systems and platforms, making them vulnerable to the same hardware and software flaws that affect those systems. Even block chain solutions designed for a specific use case typically make use of commodity components.

These environments need to be protected in the same way as other mission-critical IT systems in an organization. One of the most important parts of any effective cyber security plan is locating and fixing previously discovered flaws.

End-User Vulnerabilities

The point of interaction between consumers and a block chain is typically the weakest link and the target of attackers. To steal crypto currency, hackers often target security flaws in interconnected infrastructure. Even if an individual digital wallet is compromised or a private key is taken and used to drain funds, the block chain will continue to function normally.

Due to security flaws in the application, hackers may be able to get access to and compromise private block chains by posing as legitimate users.

III. METHODOLOGY

Here, we outline a machine learning strategy for tracking out strange patterns of behavior in Block chain networks. Due to the scarcity of labeled data, our proposed method can be considered unsupervised; after all, successful assaults are exceptional occurrences that don't follow any particular patterns. Therefore, supervised models have subpar detecting abilities.

A neural encoder-decoder model is constructed in the proposed anomaly detection system, which can summaries data on the ledger's current state into a latent space and subsequently reconstruct the original data. The fundamental notion is that essential data attributes are maintained throughout the encoding/decoding process if the current state is consistent. Anomalies, on the other hand, have values that are inconsistent and lead to a failed reconstruction. This may occur, for instance, if the supply of ether (the ETC block chain's native coin) is disproportionately large relative to the other variables. To an encoder-decoder, this quantity is just noise, thus it won't be used in the reconstruction. Therefore, the alarm would be triggered because the disparity between the original and reconstructed values would draw attention to the out-of-the-ordinary circumstance.

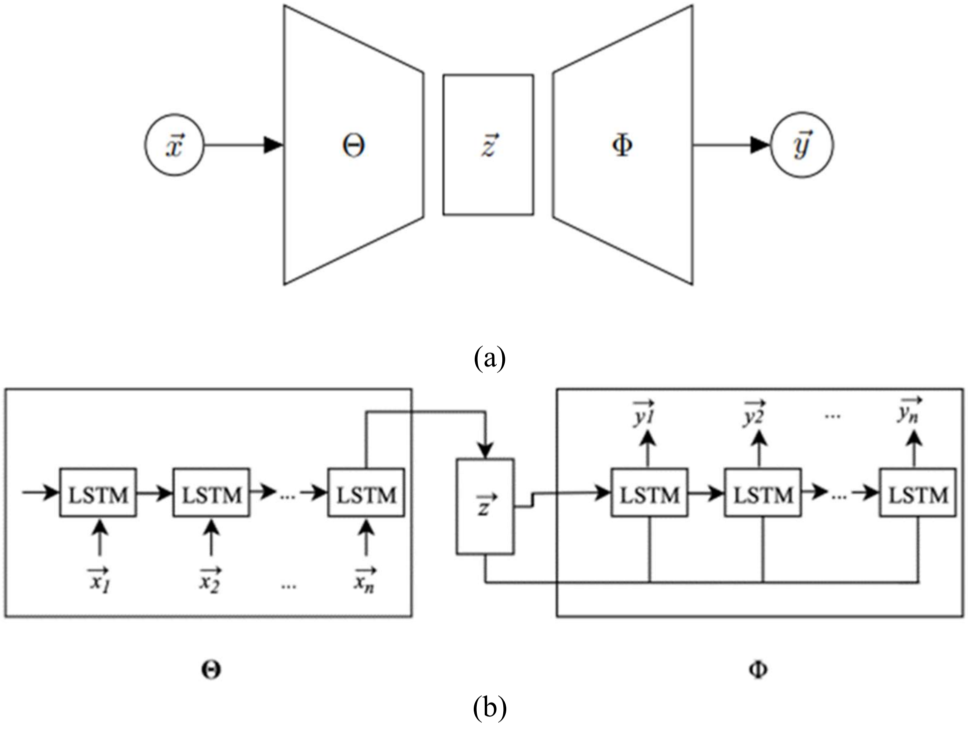
The model is applicable to time-ordered lists of occurrences. We assume, formally, that the information is sequential in nature. X = $\{\bar{x}_{1,} \dots, \bar{x}_N\}$ relative to a period of observation, where $\bar{x}_t$is a vector of features describing the t-th event in chronological order within X. An anomaly is an unexpected event in X, i.e. a vector $\bar{x}_t$ significantly different from its neighbors $\bar{x}_{t-w}$, . . ., $\bar{x}_{t-1}$, $\bar{x}_{t+1}$, . . ., $\bar{x}_{t+v}$ (for a given window [t − w, t + v] to be determined). Thus, the model should be capable of summarizing all the regularities in the data that characterize the sequence $\bar{x}_{t-w}, \dots, \bar{x}_{t-v}$.

Autoencoders provide a powerful answer to the unsupervised problem of learning a compact representation of the latent characteristics characterizing a piece of information. A neural architecture that can be taught to output an exact copy of its input is called an autoencoder. As can be seen in Figure 1a, it consists of two parts.

- The encoder Θ, a neural network whose goal is to map an input $\bar{x}$ to a latent compact representation Θ ($\bar{x}$) = $\bar{z} \in$ R^K. This mapping produces an embedding of the original input into a latent vector of size K.

- A decoder Φ, another neural networks that, given a K-dimensional vector $\bar{z}$, aims at producing an output Φ ($\bar{z}$) = $\bar{y}$ that is as close as possible to the original input.

In the above description, the components Θ, Φ, $\bar{x}$ and $\bar{y}$ are unspecified. We model the autoencoder's input and output x, y as time-stamped events within our framework. This means that recurrent networks may be defined for both Θ and Φ (RNNs). Since an RNN keeps (partial) memory of each event while it iterates through the sequence, it is well-suited to sequential data. In our code, the RNNs are instantiated as LSTMs.



(a)

(b)

Figure 2: (a) General autoencoder structure (b) Recurrent Autoencoder (RAE)

As can be seen in Figure 1b, this design yields a sequence-to-sequence recurrent autoencoder (RAE) model. In a nutshell, RAE's goal is: (i) to read the whole series of events, (ii) extract a compact representation of all the events inside the sequence, and (iii) use this representation to build a new sequence that is identical to (or very similar to) the input sequence. Mathematically, this can be specified as follows: given an input sequence I = {$\overrightarrow{x_1}$, . . . ,$\vec{x}_n$}, we compute an output sequence O = {$\bar{y}_1$, . . . , $\bar{y}_n$} where:

$$\begin{aligned}
\vec{h}_t^{(e)} &= \mathtt{LSTM}_\theta(\vec{x}_t, \vec{h}_{t-1}^{(e)}) \\
\vec{z} &= \mathtt{mlp}_\vartheta(\vec{h}_n^{(e)}) \\
\vec{h}_t^{(d)} &= \mathtt{LSTM}_\phi(\vec{z}, \vec{h}_{t-1}^{(d)}) \\
\vec{y}_t &= \mathtt{mlp}_\varphi(\vec{h}_t^{(d)})
\end{aligned}$$

Where LSTMθ and mlpϑ represent the encoder, with internal state h ⃗_t^((e))given the t-th event; symmetrically, LSTMφ and mlpϕ represent the decoder, with inner stateh ⃗_t^((d)). Further, mlpϑ and mlpϕ represent multilayer networks parameterized by θ and φ, respectively. Since the main purpose of RAE is to reconstruct the input from a compact representation, the model can be trained by considering a reconstruction loss:

$$\ell(\mathcal{I}, \mathcal{O}) = \frac{1}{n}\sum_{t=1}^{n}\|\vec{x}_t - \vec{y}_t\|_2$$

Input subsequences are obtained from X through a sliding window mechanism. Each time step within X is associated with a subsequence W$_t$ =$\vec{x}_{t-m+1}, \dots, \dots, \vec{x}_t$ where m is the window size. Thus, RAE is trained on the set {W$_m$, . . . , W$_N$ } of subsequences that can obtained from X, as illustrated in Figure 3, by learning to reconstruct them in a way that minimizes the specified loss.

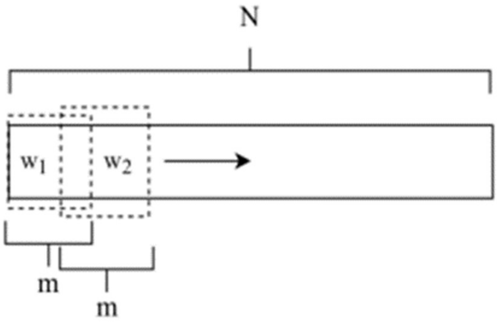


Figure 3: Sliding window mechanism.

By translating input data to compact representations that can contain all the necessary elements characterizing normal behavior, RAE may learn to mimic such behavior. If anything strange happens, we won't be able to rely on this skill to rebuild the past. This means that the analysis sequence's outlieriness may be scored based on the distance between input and output. Given that a given event may fall inside more than one overlapping time frame, the final score for that event is determined by averaging the outlier scores of those windows:

$$o(\vec{x}_t) = \frac{1}{m} \sum_{i=1}^{m} \|W_i - \tilde{W}_i\|_2,$$

where $\widetilde{W}_i$ = dec(enc(W$_i$)) is the output of the RAE with input Wi.

IV. ANALYSIS OF THE ETHEREUM CLASSIC NETWORK

The preceding section's proposed model is an example of a generic strategy for anomaly identification in sequential data. The purpose of this section is to demonstrate how our model may be used to detect anomalies in the ETC network that may indicate the presence of an attack.

4.1 Data source and preprocessing

Our tests have been performed on a four-year-old portion of the ETC block chain (July 2015 - July 2019). Seven tables make up the Kaggle2 dataset: blocks, transactions, contracts, logs, token transfers, tokens, and traces. The tables below detail the network's use, its operational state, and its blocks.

Based on our early investigation, it appears that just two of these tables, blocks and transactions, provide data that is useful to us.

The following steps were taken to clean the raw data: (i) feature engineering, or choosing the most pertinent attribute, doing correlation analysis, filtering, and aggregating; (ii) normalization; (iii) generating data views using the sliding window approach.

When you combine the results of steps one and two, you get the following collection of useful attributes that are calculated every day:

(i) the typical block size, measured in bytes; (ii) the typical amount of supplied gas required to process a transaction. (iii) the typical level of complexity (in terms of work) involved in validating a block; (iv) the average number of transactions in a block; (v) the overall quantity of gas consumed; and (vi) the total number of transactions across all blocks.

Normalization is a process that re-scales values by eliminating seasonal, cyclical, and trending changes in the data using a moving quintile ratio in order to mitigate the impacts of instability in neural modeling. Figure 4 displays how we divided the entire dataset into four time intervals. Forget about numbers before August 7, 2015; those dates correlate to the first week of the chain's inception, and they're missing a few key variables.

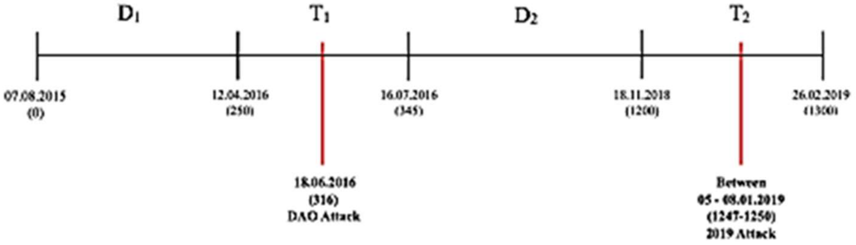


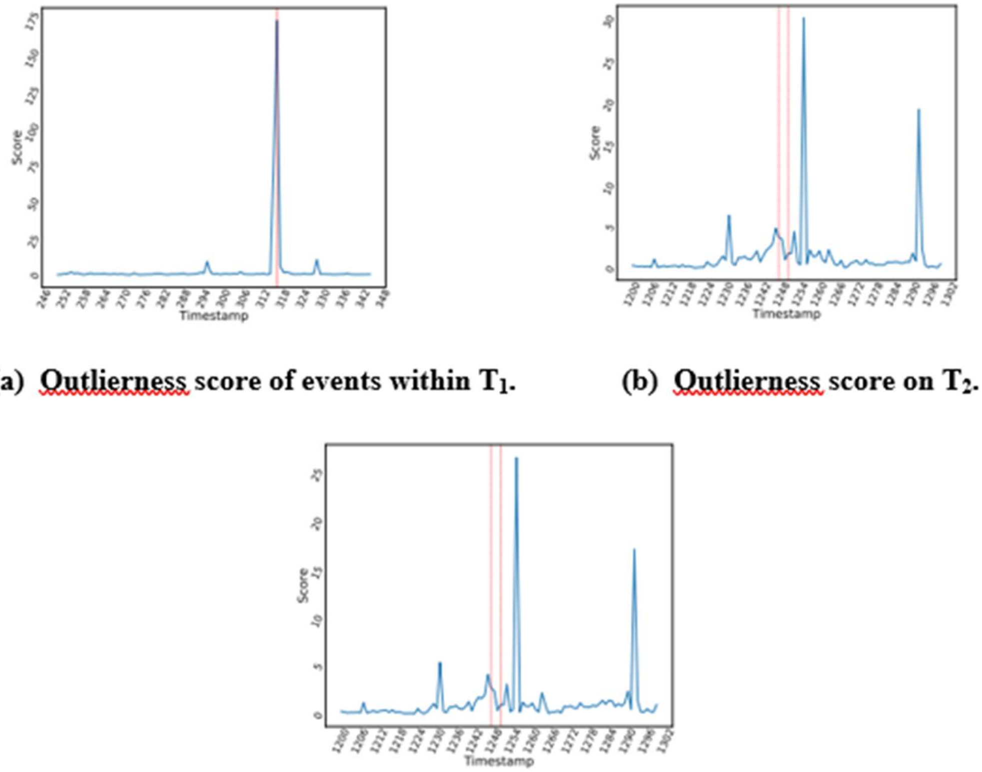
Figure 4: Data split according to the attacks.

4.2 Experimental Results

We did three distinct experiments by training two instances of RAE. The first two utilized D1 as training set and scored all the events within T1 and T2, respectively. In the third trial RAE is trained on D1 ∪ D2 and scores events in T2.

The RAE-calculated outlier scores for T1 and T2 are shown in Figures 5a and 5b. According to Figure 5a, RAE is capable of detecting the DAO attack with 100% accuracy. There is no notice from the network until day 295, when a slight increase becomes visible. On day 316, which coincides with the DAO attack, the outlierness score instead displays a difference of two orders of magnitude. Our results are in line with the literature, but RAE can pinpoint the specific day of the attack with pinpoint accuracy.

Within Figure 5b, the same model is utilized to score occurrences in T2. As the time of the impending attack draws near, a shift is visible in the outlierness score's typical behavior. However, the peak is translated of a few days. This scenario begs for some explanation. On day 1255, a few days after the actual attack, the network detects a significant abnormality. When looking at what happened during the 51% attack, it is clear that many businesses functioning on ETC were aware of the attack and froze their operations. Then, on the basis of our analysis, block size and related features seems not fully sufficient to early detect some types of attacks occurring on block chain, but we figure out that integrating data from other sources (e.g. the server operating system and the application) could improve the detection performances of our approach. As a consequence, the block chain failed to record the necessary minimum number of transactions and had to be reset in the days that followed. In actuality, it appears that all the frozen actions were logged on the network on day 1255 (where the anomaly score is greatest), marking the precise instant when the anomaly was inserted inside the block chain. This outcome is congruent with the findings of the third trial, when RAE is trained both on D1 and D2: Figure 4c displays a similar trend, therefore illustrating that the preprocessing processes make the RAE findings consistent even across various time periods. The lone false positive seems to be represented by the peak on day 1291, which appears in both models. Although there is no reporting of assaults in such periods, it would be fascinating to offer a close inspection to the activities during that era and try to explain such anomalies based on those.



(a) Outlierness score of events within T_1.

(b) Outlierness score on T_2.

(c) Outlierness score on T_2. RAE is trained on $D_1 \cup D_2$.

Figure 5: scores computed by RAE

V. CONCLUSION

This paper provides an encoder-decoder deep learning model for spotting irregularities in block chain system utilization. The rarity of the target events (attacks) is the greatest challenge in this case, necessitating the use of an unsupervised solution. In particular, the paper makes two kinds of original contributions: We deploy a sequence-to-sequence neural network model for anomaly detection in the block chain network, and we have I identified a meaningful collection of features calculated on block chain logs representing network state in defined time steps. Our model's capacity to accurately identify security breaches is demonstrated by experiments conducted on Ethereum Classic's entire history logs.

To acquire automated feature engineering and assess potential improvements, we propose to investigate the use of hybrid architectures based on the combination of RNNs with Convolutional Neural Networks in future research. However, we're keen on creating models that can foresee assaults in order to strengthen the safety of the network.

REFERENCES: -

Abdelwahed, Nagy Ramadan, Hesham Ahmed Hefny ,2020," Cybersecurity Risks of Blockchain Technology".

Xiaoqi Li, Peng Jiang, Ting Chen, Xiapu Luo, Qiaoyan Wen,2020," A Survey on the Security of Blockchain Systems".

M. Usman, M.A. Jan, X. He, and J. Chen. A survey on representation learning efforts in cybersecurity domain. ACM Comput. Surv., 52(6):111:1–111:28, 2019.

S. Mahdavifar and A.A. Ghorbani. Application of deep learning to cybersecurity: A survey. Neurocomputing, 347:149 – 176, 2019.

M. Signorini, M. Pontecorvi, W. Kanoun, and Di Pietro R. BAD: blockchain anomaly detection. CoRR, abs/1807.03833, 2018.

C. Ye, G. Li, H. Cai, Y. Gu, and A. Fukuda. Analysis of security in blockchain: Case study in 51%-attack detecting. In 2018 5th International Conference on Dependable Systems and Their Applications (DSA), pages 15–24, 2018.

N. Atzei, M. Bartoletti, and T. Cimoli. A survey of attacks on ethereum smart contracts (sok). In Principles of Security and Trust", pages 164–186. Springer, 2017.

A. Bogner. Seeing is understanding: Anomaly detection in blockchains with visualized features. In Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers, UbiComp '17, pages 5–8. ACM, 2017.

A. L. Buczak and E. Guven. A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys Tutorials, 18(2):1153–1176, 2016.

Kumar, Ram and Patil, Manoj, Improved the Image Enhancement Using Filtering and Wavelet Transformation Methodologies (July 22, 2022). Available at SSRN: https://ssrn.com/abstract=4182372

Ram Kumar, Manoj Eknath Patil ," Improved the Image Enhancement Using Filtering and Wavelet Transformation Methodologies", Turkish Journal of Computer and Mathematics Education ,Vol.13 No.3(2022), 987-993.

Ram Kumar, Jasvinder Pal Singh, Gaurav Srivastava, "A Survey Paper on Altered Fingerprint Identification & Classification" International Journal of Electronics Communication and Computer Engineering ,Volume 3, Issue 5, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209.

Chetna kwatra, Bukya Mohan Babu, M.Praveen, Dr T.Sampath Kumar, Ram Kumar Solanki ,Dr A V R Mayuri. (2023). Modified Cnn Based Heart Disease Detection Integrated With Iot. Journal of Pharmaceutical Negative Results, 993–1001. https://doi.org/10.47750/pnr.2023.14.S02.120