BSFR-SH: Blockchain-Enabled Security Framework Against Ransomware Attacks for Smart Healthcare

Mohammad Wazid, *Senior Member, IEEE*, Ashok Kumar Das, *Senior Member, IEEE*, and Sachin Shetty, *Senior Member, IEEE*

***Abstract*—Ransomware is a type of malicious program or software that encrypts the contents on a hard disc and prevents the users from accessing them unless they pay an amount (called a ransom). Most of the organizations, such as financial institutes and healthcare sectors (i.e., smart healthcare) are targeted by ransomware attacks. Ransomware assaults are among the most frightening types of cyber-attacks, and they are not confined to a specific sector or the countries. Blockchain is a tamper-proof technology, which is more secure, robust and decentralized in nature. Features of blockchain can add more security for detection and mitigation of ransomware more effectively. In this paper, we propose a new blockchain-enabled security framework to detect and defend the ransomware attacks for smart healthcare (in short, BSFR-SH). The conducted security analysis proves the security of the proposed BSFR-SH against the ransomware attacks. The performance of BSFR-SH is significantly better than the other similar existing mechanisms as it achieves better accuracy and F1-score than other compared mechanisms. Furthermore, the practical demonstration of BSFR-SH is provided to estimate the impact on important performance parameters.**

***Index Terms*—Ransomware, smart healthcare, intrusion detection, machine learning, blockchain.**

I. Introduction

RANSOMWARE is a type of malware attack in which the attacker encrypts and locks the victim's data and crucial files, and then demands money (ransom) in exchange for the data to be unlocked and decrypted. Cybercriminals utilise ransomware as a sort of malware (malicious software). If a ransomware attack happens on a device, it either limits access or encrypts the data (files) of the device. Cybercriminals demand ransom (some amount of money) from their victims in exchange for releasing the data (i.e., to provide the decryption key). A close eye and security software are recommended to protect against ransomware outbreak. After being infected with ransomware, victims have only few options, such as a) pay the money (ransom), b) try to remove the ransomware, and c) reset the device. Extortion malware commonly exploits remote systems via phishing emails and other software flaws. A ransomware attack can affect both the individual users and the companies of commerce. Ransomware is one of the most effective ways to attack businesses, key infrastructure (i.e., smart healthcare system, power grid, nuclear power plant) and the associated people. This type of malware infects computers and prevents users or external software from accessing devices until ransom demands are paid [1].

The healthcare digital experience can become like the other digital interactions. For example, we use smartphone applications for ordering of some foods online which is just done with the help of few clicks. Where we order, pay, and even customize how our foods should be cooked, we can tell them when and where it should be delivered. Consumer electronics and technology have led to the personalized, integrated, and seamless experiences in the consumer-facing industries. This can be followed in the healthcare domain (i.e., smart healthcare). For instance, we can have elderly people at home (i.e., smart home). To monitor their health or to support their day-to-day activities, we can deploy different consumer electronics devices and products (i.e., smart coffee makers, smart air conditioners and wearable healthcare devices), which can do this task with the help of installed software and Internet connectivity [2]. The smart healthcare system should provide a more seamless and customised experience to fulfil rising customer expectations as we transit to a more consumer-centric future of health. Therefore, smart healthcare system becomes a part of consumer electronics. It can be considered as the consumer healthcare technology, where consumer healthcare electronics devices (i.e., wearable healthcare devices) support the people in various ways. Consumers (i.e., patients) expect healthcare technologies should function like those of other industries. In this direction, a better technology and digital experience can improve their satisfaction if it helps to increase the efficiency of the system and provide more convenience. It may include virtual visits to hospitals/clinics, registration, online scheduling of appointment and bill payments are among those being adopted by hospitals so that they can interact directly with the consumers for accessing, communicating and delivering better healthcare and bill payments.

Smart healthcare operates through new generation of information technologies, like "Internet of Things (IoT)", "Big data", "cloud computing", and "Artificial Intelligence (AI)".

Manuscript received 20 May 2022; revised 10 July 2022, 25 August 2022, and 11 September 2022; accepted 20 September 2022. Date of publication 22 September 2022; date of current version 26 January 2023. *(Corresponding author: Mohammad Wazid.)*

Mohammad Wazid is with the Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun 248002, India (e-mail: wazidkec2005@gmail.com).

Ashok Kumar Das is with the Center for Security, Theory and Algorithmic Research, International Institute of Information Technology Hyderabad, Hyderabad 500032, India, and also with the Virginia Modeling, Analysis and Simulation Center, Old Dominion University, Suffolk, VA 23435 USA (e-mail: ashok.das@iiit.ac.in).

Sachin Shetty is with the Virginia Modeling, Analysis and Simulation Center, Department of Computational Modeling and Simulation Engineering, Old Dominion University, Suffolk, VA 23435 USA (e-mail: sshetty@odu.edu).

Digital Object Identifier 10.1109/TCE.2022.3208795

They have transformed the current medical system, making it more efficient, simple, and personalised [3]. Cyber security makes the use of certain tools and techniques that secure the devices, servers, networks and the associated data from the malicious invasions (i.e., security breaches). Cyber attacks often try to get access to the important systems and devices. Then, the important data can be updated or deleted. They sometimes extort money, disrupt day-to-day operations or make the system unavailable. Because we have now more devices than people, cyber attackers are becoming more sophisticated day by day. Therefore, installing effective cyber security measures has become critical [4], [5].

Blockchain is a data storage mechanism, which is difficult or impossible to alter, hack, or cheat. A blockchain is a distributed digital database of transactions that is duplicated and distributed across the blockchain's network of computer systems (i.e., a Peer-to-Peer (P2P) cloud servers network). Each block in the chain comprises a number of transactions, and each participant's ledger is updated whenever a new transaction occurs on the blockchain (i.e., a distributed ledger). The distributed ledger technology (DLT) is a kind of decentralised database, which is administered by multiple parties, called as the miner nodes. In blockchain, the transactions are recorded using a hash, which is a non-changeable cryptographic signature. It can track the assets, record transactions and build trust between parties. The important properties of blockchain are "programmable, secure, anonymous, distributed, time-stamped, immutable and unanimous" [6], [7].

Features of the blockchain technology can add more security for detection and mitigation of ransomware more effectively. Therefore, we propose a blockchain-enabled security framework to detect and defend the ransomware attacks for smart healthcare. In our proposed framework, the data related to various ransomware attacks is collected through different sources (i.e., honeypot systems can be deployed). From the collected data, the digital signatures are built for the malicious ransomware programs and legitimate programs. Apart from this, some features are also built for the malicious ransomware programs, which are helpful to perform the feature-based ransomware detection along with the signature based ransomware detection. After performing this process, the generated information of ransomware (i.e., signatures and features) is stored in the form of blocks in the blockchain. Such type of information is helpful to detect the ransomware attacks, which can happen in the future.

A. Motivation

Since the cyber attacks may cause assault on the intellectual property, there may be theft of personal information, financial data, deletion of sensitive and compromised data. This further causes a high damage to the reputation of the organisation. Ransomware attacks are among the most frightening types of cyber attacks and they are not confined to specific countries. Ransomware attacks have also caused significant loss all around the globe. This scenario becomes very serious when they target the smart healthcare system as it is the backbone of the healthcare of modern era. The operations and services of the healthcare system should be expected to be available in 24×7. However, the patients, their relatives and healthcare staffs do not get the services of the healthcare system under the influence of the ransomware attacks. Hence, detection and mitigation of ransomware attacks become essential. As a result, we need a robust security mechanism to mitigate the ransomware attacks. In addition, features of the blockchain can add more security to such a mechanism [7], [8]. Therefore, a blockchain based security framework is then needed for the mitigation of ransomware attacks in smart healthcare.

B. Research Contributions

The main research contributions made in the paper are given below.

- We propose a new blockchain enabled security framework to detect and defend the ransomware attacks for smart healthcare applications (in short, we call it as BSFR-SH).
- The conducted security analysis proves the security of the proposed BSFR-SH against the various potential attacks along with the ransomware attacks.
- The performance of the BSFR-SH is better than the other similar mechanisms of the domain as it has achieved better accuracy and F1-score than the other compared mechanisms.
- The practical demonstration of BSFR-SH is also provided to get the estimation of some important performance parameters, such as accuracy, F1-score, computational time, and transactions per second.

C. Paper Outline

The remaining part of the paper is organised as follows. The ransomware attacks and security requirements are presented in Section II. The summary of various related schemes is given in Section III. The different phases of the proposed blockchain enabled generic ransomware defense framework (BSFR-SH) are given in Section IV. The security analysis of BSFR-SH is provided in Section V. The comparison of various relevant schemes is given in Section VI. Next, a practical implementation of BSFR-SH is shown in Section VII. Finally, the paper is concluded in Section VIII with some future works.

II. Attacks and Working Mechanisms of Ransomware

In this section, we discuss about the various incidents and types of ransomware attacks, and the working mechanisms of ransomware.

A. Ransomware Attacks

Crypto and locker ransomware are the two most common types of ransomware. Threat actors have recently become interested in double extortion and ransomware as a service (RaaS).

- *Locker ransomware:* The Locker ransomware prevents the users from accessing their computers. To enter the systems, this version employs social engineering tactics and compromised credentials. Threat actors get access to

the system and prevent users from using it until a ransom is paid. A pop-up message may show on the victim's screen (i.e., "Your computer was used to view websites with unlawful content. You must pay a fine of 100 USD to unlock your computer or a virus has been installed on your computer. To fix the problem, go here").

- *Crypto ransomware:* Crypto ransomware that encrypts files is more frequent and ubiquitous than locker ransomware. It encrypts all or partial files on a computer and demands a ransom in exchange for the decryption key from the victim. Some of the more recent varieties infect shared, networked, and cloud storage as well. Malicious emails, websites, and downloads are all ways that crypto ransomware spreads.
- *Double extortion ransomware:* To blackmail victims into paying a ransom, it encrypts files and exports data. Attackers, using double extortion ransomware, threaten to reveal stolen data if their demands are not satisfied. This means that the attacker still has control over the victim, even if they can recover their data from backup.
- *Ransomware as a service (RaaS):* RaaS entails criminals renting access to a ransomware strain from the author of the ransomware, who sells it as a pay-per-use service. RaaS developers put their ransomware on dark Web sites and sell it as a subscription to criminals, similar to the "software as a service (SaaS)" model. Once the member infects the machines then he/she has to pay some money to the RaaA creator as per their agreement.

Some well-known ransomware attacks include locky, WannaCry, bad rabbit, Ryuk, shade/troldesh, jigsaw, CryptoLocker, Petya, and GoldenEye [4], [5].

B. Distribution Mechanisms of Ransomware

The various distribution mechanisms of ransomware are given below [5], [9], [10].

- *Phishing email:* A ransomware infection is spread by clicking a link in an email that leads to a rogue website.
- *Email attachments:* A ransomware infection can be injected by enabling malicious macros in an email attachment; or downloading a document containing a "remote access trojan (RAT), or downloading a ZIP file containing a malicious JavaScript or windows script host (WSH) file".
- *Use of social media:* On facebook, twitter, social media posts, instant messenger dialogues, and other comparable sites, by clicking a malicious link a ransomware infection may spread.
- *Malvertising:* A ransomware infection is also spread by going to a legitimate advertising website that has been infected with malware.
- *Virus-infected software:* By installing a program or application that is infected with malware, a ransomware infection may occur.
- *Infections acquired by accident:* A ransomware infection can happen when someone visits a dangerous, suspicious, or fraudulent website or when someone opens or closes a pop-up window. It is worth noting that a genuine Web page can be hacked if malicious JavaScript code is introduced into the content. This can be used to distribute the malware further.
- *Traffic management system (TDS):* By clicking a link on a legitimate gateway Web page, the user is redirected to a malicious site based on the user's geo-location, browser, and operating system. In this way, a ransomware infection may also spread.
- *Self-propagation:* A ransomware infection can happen by using network and USB drives to spread the malicious code to other devices.

C. Working Mechanisms of Ransomware

The ransomware assault follows the steps below after a device is infected with malicious code. Ransomware can remain dormant on a device until it is at its most vulnerable, at which point it will attack [5], [10].

- *Infection:* The devices (i.e., smartphone, laptop, desktop and server) are inadvertently infected with ransomware.
- *Execution and identification of resources:* Ransomware checks and maps the locations of specific file types, such as "locally stored data and network-accessible systems, which are mapped and unmapped". In some attacks, the back-up files and folders are also encrypted or deleted.
- *Encryption of resources:* The encryption key is used to encrypt all files detected during the execution stage after a key exchange with the command and control server happens. Then, all the files and resources are encrypted with a selected key. Thus, it restricts access to the files and resources to the victims.
- *Notification to users:* Ransomware adds pay-for-decryption instruction files to the system, and then utilises those files to show a ransom note to the user.
- *Cleaning process:* Typically, ransomware terminates and deletes itself, leaving only the payment instructions files behind.
- *Payment process:* The victim goes to a Web page with more information on how to pay the ransom by clicking on a link in the payment instructions. To evade detection by network traffic monitoring, "The Onion Router (TOR)" based hidden services are used to encapsulate the communication that obfuscates the conversations. Note that TOR is a "free and open-source software for enabling anonymous communication".
- *Decryption of resources:* The victim may acquire the decryption key after paying the ransom, which is usually done via the attacker's bitcoin address. There is no guarantee, however, that the decryption key will arrive on time or it will work correctly.

The working mechanisms of ransomware attacks are depicted in Fig. 1.

III. Prior Related Work

In this section, we provide the summary of other similar schemes of the domain.

Almashhadani *et al.* [11] used Locky, one of the most hazardous ransomware families, as a case study to present

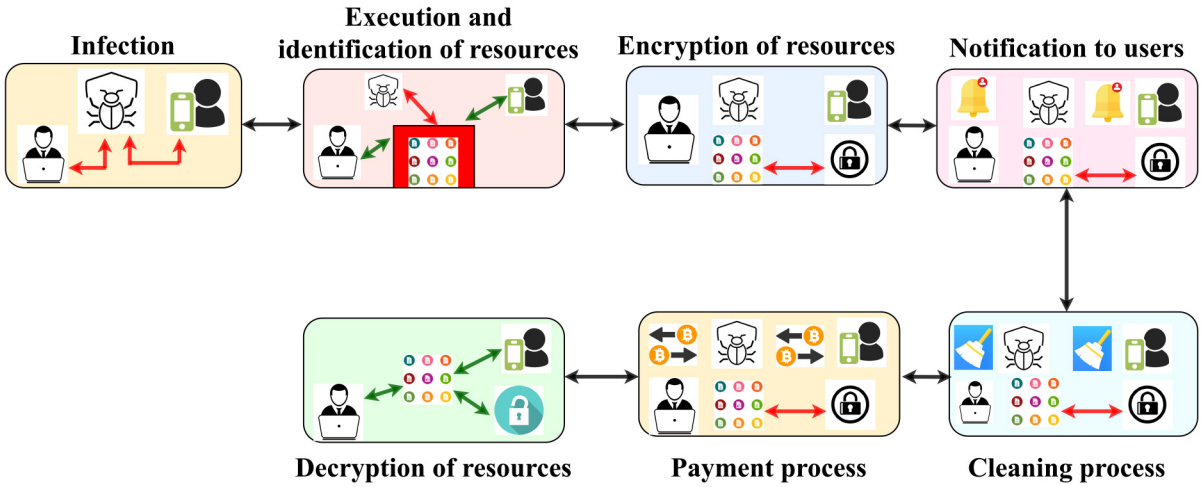


Fig. 1. Working mechanisms of ransomware (Source: [10]).

a comprehensive behavioural analysis of crypto ransomware network operations. A dedicated testbed was built, and a variety of relevant and instructional network properties were gathered and organised into numerous categories.

Hwang *et al.* [12] demonstrated a two-stage hybrid ransomware detection model, which included both a Markov and a Random Forest model. They then created a Random Forest machine learning model to control both "false positive (FPR) and false negative (FNR) error rates" using the remaining data.

Sharmeen *et al.* [13] overcomes the limitations of typical supervised detection engines. They introduced a semi-supervised framework that uses deep learning algorithms to compute the inherent latent sources of various patterns in fresh variations in an unsupervised manner.

Bae *et al.* [14] demonstrated a method for identifying ransomware that can tell the difference between ransomware and benign files, as well as malware and ransomware. According to the results of their experiments, their approach found ransomware among malicious and benign files.

In a ransomware family, Akcora *et al.* [15] offered a tractable data analytics system for automatically detecting new hazardous addresses. Min *et al.* [16] presented "Amoeba", a solid-state drive (SSD) solution that allows for automated backup for overcome the ransomware attacks.

Zhang *et al.* [17] proposed a detection framework, called as TGAN-IDS. It is based on the dual generative adversarial networks. In this framework, a deep convolutional generative adversarial network (DCGAN) was adopted to train a generator, which has good performance to generate adversarial sample.

Chakkaravarthy *et al.* [1] proposed an intrusion detection honeypot (IDH) to address the security problems of the systems. Their scheme contains important components, like honeyfolder, audit watch and complex event processing (CEP). Baek *et al.* [18] then proposed a ransomware defense scheme, named as "SSD-Insider++," which mitigates the users' files from being damaged by the ransomware attacks. Their proposed SSD-Insider++ has facility to embed into an solid-state drive (SSD) controller in the form of a firmware.

Min *et al.* [19] proposed a technique, which is a device-level backup solution. It does not require any additional storage for backup. It is equipped with important segments, such as: 1) a hardware accelerator to run "content-based detection algorithms for ransomware detection" and 2) a fine-grained backup control mechanism for minimizing "space overhead for data backup". Almashhadani *et al.* [11] also demonstrated a comprehensive behavioral analysis of crypto ransomware (locky). A dedicated testbed was built, and a set of valuable and informative network features were extracted. Furthermore, the features were classified into multiple types.

Poudyal and Dasgupta [20] created a hybrid "Artificial Intelligence (AI)"-powered technique to overcome recent ransomware detection hurdles. They presented a deep inspection approach for multilevel crypto-ransomware profiling that captured different information at the Dynamic link library, function call, and assembly levels. Bajpai and Enbody [21] also introduced the ransomware kill chain to show their attackers the steps they must take to achieve their mitigation goal.

IV. BSFR-SH: Proposed Blockchain-Enabled Generic Ransomware Defense System for Smart Healthcare

In the proposed scheme, the blockchain enabled generic ransomware defense system works through the following phases: (i) creations of backups of healthcare data through blockchain, (ii) data collection and signature generation through blockchain, (iii) ransomware detection and analysis through machine learning, (iv) mitigation of ransomware, and (v) data recovery through blockchain. The scenario of all phases is depicted in Fig. 2. The details of various notations used in the paper are given in Table I.

A. Creation of Backups of Healthcare Data Through Blockchain

In this phase, the backups of healthcare data of the systems is created through blockchain in which data is stored in the form of certain encrypted transactions. The tasks of this phase

TABLE I
USED NOTATIONS

Symbol	Description
$P2PCS$	Peer-to-peer cloud servers network
CS_l	l^{th} cloud server over $P2PCS$
DT_{BU}	Data backup unit
SYS_i	i^{th} system under ransomware attack
RW	Ransomware
$\mathcal{A}$	Attacker (creator) of ransomware RW
SK_{CS_l,SYS_i}	Session key between CS_l and SYS_i
KU_{CS_l}	Public key of CS_l
$E_{KU_{CS_l}}(Tx_j)$	Encrypted transaction used in DT_{BU}
$BC_{DT_{BU}}$	Blockchain of data backups
β_j	j^{th} block in $BC_{DT_{BU}}$
β_{Ver_j}	Version of β_j
TS_{DT_j}	Timestamp used in β_j
RN_{DT_j}	Random nonce used in β_j
MTR_{DT_j}	Merkle tree root used in β_j
OID	Owner's identity for β_j
OKU	Owner's public key
$HC\beta_j$	Hash value of current block for β_j
$HP\beta_{j-1}$	Hash of previous block for β_j
Sig_{β_j}	Signature of block β_j
L	Leader node selected from P2PCS
$pBFT$	Practical Byzantine fault tolerance algorithm
$BC_{Sig_{RW}}$	Blockchain of RW
HP_{RW}	Deployed honeypot for RW
DT_{RW}	Data unit of RW
$SK_{CS_l,HP_{RW}}$	Session key between CS_l and HP_{RW}
Sig_{RW}	Signature of RW extracted from DT_{RW}
FT_{RW}	Features of RW extracted from DT_{RW}
$E_{KU_{CS_l}}(Tx_i)$	Encrypted transaction used for Sig_{RW}, FT_{RW}
β_i	i^{th} block in $BC_{Sig_{RW}}$
β_{Ver_i}	Version of β_i
TS	Timestamp used in β_i
RN	Random nonce used in β_i
MTR	Merkle tree root used in β_i
$HC\beta_i$	Hash value of current block in β_i
$HP\beta_{i-1}$	Hash of previous block in β_i
Sig_{β_i}	Signature of block β_i
DM_{CS_l}	Ransomware detection module of CS_l
ML	Machine learning
$RFor$	Random forest algorithm
$LReg$	Logistic regression algorithm
$DTree$	Decision tree algorithm
KNN	k-nearest neighbors algorithm
$NProf$	Normal profile (absence of ransomware attack)
$AProf$	Abnormal profile (presence of ransomware attack)
$AMsg$	Alert message for SYS_i for RW
Inf_{SYS_i}	Infected files of SYS_i with RW
RW_{amt}	Desired ransomware amount by creator of RW
DT-SYS$_{i-amt}$	Actual data price of SYS_i
K_d	Decryption key provided by $\mathcal{A}$

are executed over the "peer-to-peer cloud server (P2PCS)" network. As they are resource rich devices of the network and have high computation, storage and communication capabilities [16]. For the blockchain implementation and consensus process, we have used practical Byzantine fault tolerance method.

In this work, we consider only the private blockchain because the healthcare data that we store may be sensitive in nature. Because the private blockchain is limited in size, it can be very fast and can process transactions much more quickly as compared to that for a public blockchain. In case of a private blockchain, the enterprises have the ability to communicate without a third party or the necessity for a central authority [22]. However, in case of a public blockchain, the data is accessible to anyone who wishes to use or just have a look at them. Moreover, whether it is private or public, they are proof against any alteration as they are considered as transparent and safe. In future, we have plan to work with hybrid blockchain.

Algorithm 1 Creation of Backups of Data Through Blockchain

Output: Blockchain of data backup $BC_{DT_{BU}}$

All CS_l's collect DT_{BU} securely via the session key SK_{CS_l,SYS_i}
CS_l creates encrypted transactions via DT_{BU}, that is, $E_{KU_{CS_l}}(Tx_m) = E_{KU_{CS_l}}(DT_{BU})$ where $m = 1, 2, \cdots Nd_{Tx}$ and Nd_{Tx} is the number of transactions in a block β_j
CS_l creates a block β_j by putting $\{\beta_{Ver_j}, TS_{DT_j}, RN_{DT_j}, MTR_{DT_j}, OID, OKU, \{E_{KU_{CS_l}}(Tx_m)| m = 1, 2, \cdots, Nd_{Tx}\}, HC\beta_j, HP\beta_{j-1}, Sig_{\beta_j}\}$
CS_l broadcasts β_j to the P2PCS network
A leader L selected from the P2PCS network calls the consensus process using pBFT for addition of β_j in $BC_{DT_{BU}}$
if a threshold fraction of miners commit on addition of β_i in $BC_{DT_{BU}}$ **then**
    β_j is added in $BC_{DT_{BU}}$
else
    Start the consensus process again
end if
if all the blocks are added **then**
    return $BC_{DT_{BU}}$
else
    Continue the process
end if

It is important to mention that all exchanged messages of information happen in a secure way after performing the steps of any standard mutual authentication and key establishment mechanism. For example, a session key SK_{E_A,E_B} can be used for the secure exchange between two communicating entities, say E_A and E_B. The procedure is also explained in Algorithm 1.

B. Data Collection, Signature Generation and Features Building Through Blockchain

In this phase, the data related to various ransomware attacks is collected through different sources (i.e., honeypot systems can be deployed) [1]. The tasks of this phase are executed over the peer-to-peer cloud server (P2PCS) network. As they are resource rich devices of the network and have high computation, storage and communication capabilities.

The collected data is pre-processed to remove the abnormalities. From the collected data signatures are built for the malicious ransomware programs and legitimate programs [1]. Apart from that some features are also built for the malicious ransomware programs, which is helpful to perform the feature based ransomware detection along with the signature based ransomware detection. Here, it is important to mention that the "Elliptic Curve Digital Signature Algorithm (ECDSA)" [23] has been used. After performing this process, the generated information of ransomware (i.e., signatures and

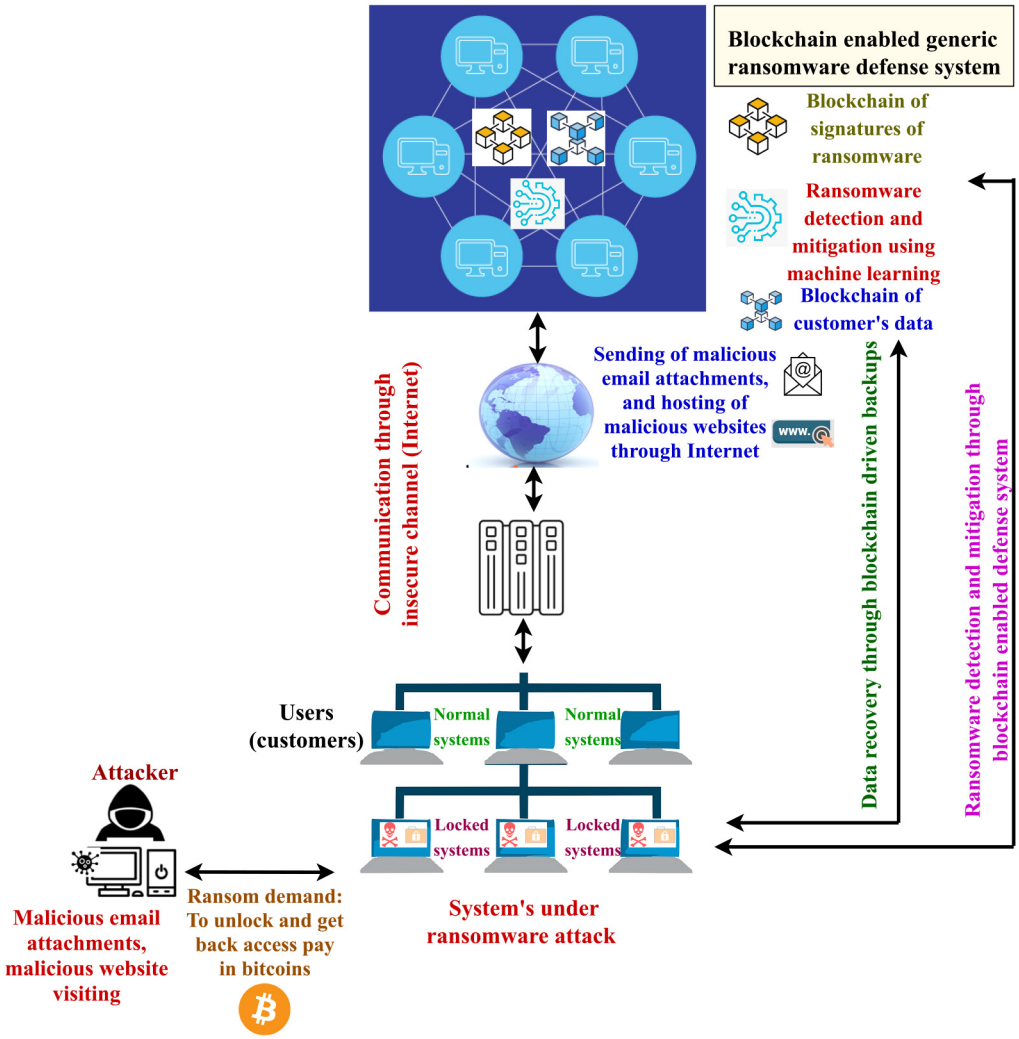


Fig. 2. Blockchain enabled generic ransomware defense system.

features) are stored in the form of blocks in the blockchain. This information is helpful to detect the ransomware attacks, which will be happened in the future. For the blockchain implementation and consensus process, we have used practical Byzantine fault tolerance consensus algorithm [24]. This procedure is explained in Algorithm 2.

C. Analysis for Ransomware Detection Through Machine Learning (ML)

In this phase, we do the analysis of all connected system of the enterprise through the machine learning algorithms and signature base of ransomware that we have maintained in $BC_{Sig_{RW}}$.

To perform the analysis, the authorised cloud server, say CS_l, gets data from the P2PCS network, which is stored in $BC_{Sig_{RW}}$. However, for that purpose the encrypted transactions should be decrypted first. For the ML based analysis, we use the state of art ML algorithms, like random forest, logistic regression, decision tree and k-nearest neighbors algorithm (KNN). CS_l does the analysis and tries to find out any sign of ransomware in the system. The ML algorithms are trained through the information available in $BC_{Sig_{RW}}$, such as Sig_{RW}, FT_{RW}, which is also called as the training phase of the ransomware detection module. Here, the normal and abnormal profiles *NProf* and *AProf* are created. *NProf* has the definition of normal files, which are not ransomware and *AProf* has the definition of abnormal files (i.e., ransomware). Next, the detection of ransomware is performed in the real-time mode. This work is conducted in the testing phase. Since the ransomware information (such as Sig_{RW}, FT_{RW}) is maintained in $BC_{Sig_{RW}}$ in the encrypted transactions, it is very difficult for an adversary to corrupt the detection module. Once this model detects any sign of ransomware in the system, it starts the procedure of "mitigation of ransomware" phase to mitigate the ransomware attacks. The entire procedure is explained in Algorithm 3.

D. Mitigation of Ransomware

In this phase, the mitigation of ransomware is done.

In the previous phase, an authorised CS_l did the detection of any existence of ransomware in the system. If CS_l detects the existence of any ransomware in the system, it tries to mitigate the attack by raising the alert messages. First of all that infected system should be isolated so that other normal

Algorithm 2 Data Collection, Signature Generation and Features Building Through Blockchain

Output: Blockchain of signatures of ransomware $BC_{Sig_{RW}}$

Deploy HP_{RW}
All CS_l's collect DT_{RW} securely via their session keys $SK_{CS_l,HP_{RW}}$
CS_ls do pre-processing over DT_{RW}
CS_ls clean DT_{RW} to get DT_{RWC}
Each CS_l creates Sig_{RW} from DT_{RWC}
Each CS_l generates FT_{RW} from DT_{RWC}
Each CS_l generates $E_{KU_{CS_l}}(Tx_i) = E_{KU_{CS_l}}(Sig_{RW}, FT_{RW})$ where $i = 1, 2, \cdots N_{Tx}$ and N_{Tx} is the number of transactions in a block β_i
CS_l creates β_i by putting $\{\beta_{Ver_i}, TS, RN, MTR, OID, OKU, \{E_{KU_{CS_l}}(Tx_i)|i = 1, 2, \ldots N_{Tx}\}, HC\beta_i, HP\beta_{i-1}, Sig_{\beta_i}\}$
CS_l broadcasts β_i to P2PCS network
A leader L selected from the P2PCS network calls consensus using pBFT for addition of β_i in $BC_{Sig_{RW}}$
if a threshold fraction of miners commit on addition of β_i in $BC_{Sig_{RW}}$ **then**
    β_i is added in $BC_{Sig_{RW}}$
else
    Start the consensus process again
end if
if all blocks are added **then**
    return $BC_{Sig_{RW}}$
else
    Continue the process
end if

Algorithm 3 Analysis for Ransomware Detection Through ML

Output: Detection of *RW*

CS_l decrypts $BC_{Sig_{RW}}$ to retrieve Sig_{RW} and FT_{RW}
CS_l does the training of DM_{CS_l} through *RFor*, *LReg*, *DTree* and *KNN*
DM_{CS_l} creates *NProf* and *AProf* via Sig_{RW} and FT_{RW} using *RFor*, *LReg*, *DTree* and *KNN*
DM_{CS_l} starts detection via *NProf* and *AProf*
if DM_{CS_l} detects *RW* **then**
    Call "mitigation of ransomware" phase
else
    Initiate detection process again
end if

systems do not get the spreading of the ransomware. Next, the associated application of the detection module, which is installed in that infected system, tries to overcome the situation by erasing the infected files from the system. Under these circumstances, there are some possibilities, for instance, (i) if the ransomware is removed then system initiates its normal working again, (ii) the data and files, which are locked by the ransomware can be deleted if system is formatted and then data can be recovered from the "blockchain of customer's data", and (iii) if due to some technical problems backup of data is not available and amount of ransom (money) is less than the actual data price, which is available in the system then proposed mechanism initiates the secure payment of ransom to the ransomware generator (attacker). After the successful payment it tries to get decryption key K_d from $\mathcal{A}$. Then it needs to decrypt the files of SYS_i via K_d. It is important to mention that

Algorithm 4 Mitigation of Ransomware

Output: Mitigation of detected *RW*

DM_{CS_l} of CS_l detects *RW* through *NProf* and *AProf*
if DM_{CS_l} discovers *RW* in SYS_i **then**
    CS_l raises *AMsg* for SYS_i, and SYS_i is isolated
    DM_{CS_l} erases Inf_{SYS_i} using one of the following cases:
    Case-1: DM_{CS_l} erases *RW*
    Case-2: DM_{CS_l} formats SYS_i and recovers data via "data recovery through blockchain" phase
    Case-3: If DM_{CS_l} finds $RW_{amt} <$ DT-SYS$_{i-amt}$, it then pays RW_{amt} and gets the decryption key K_d from the adversary $\mathcal{A}$; otherwise, it follows Case-1 and Case-2
else
    Initiate mitigation process again
end if

Algorithm 5 Data Recovery Through Blockchain

Output: Data recovery DT_{BU} through blockchain $BC_{DT_{BU}}$

CS_l identifies SYS_i that needs for data recovery
CS_l starts the recovery of data DT_{BU} through $BC_{DT_{BU}}$
CS_l raises the request to decrypt the encrypted transactions $E_{KU_{CS_l}}(Tx_j)$ to get DT_{BU}
The concerned cloud server CS'_l decrypts $E_{KU_{CS_l}}(Tx_j)$ and provides DT_{BU} to CS_l securely using secret key $SK_{CS'_l,CS_l}$
CS_l provides DT_{BU} to SYS_i securely via the session secret key SK_{CS_l,SYS_i}
SYS_i stores DT_{BU} in its storage unit
if data recovery is done **then**
    Terminate the process
else
    Continue recovery of data
end if

for the recovery of data, the steps of "data recovery through blockchain" are utilized.

The entire procedure is explained in Algorithm 4.

E. Data Recovery Through Blockchain

In this phase, the data of the systems, which is under the attack of ransomware, is recovered.

Under a ransomware attack, the important data of the system is not available to the legitimate users of the system due to its encryption by the ransomware. As we have discussed earlier, the backups of the data DT_{BU} were created thorough the blockchain via the steps of "creations of backups of data through blockchain" phase. The authorised CS_l can initiate the data recovery process after the detection of the ransomware attack on a system. In this process, the data backups maintained over the blockchain $BC_{DT_{BU}}$ are utilized. CS_l searches for the required data in the $BC_{DT_{BU}}$ and then it raises the request to decrypt the encrypted transactions $E_{KU_{CS_l}}(Tx_j)$ as the data is in the encrypted form. CS_l provides the data to the associated system in a secure way. Here, it is important to mention that all exchanges of information happen in the secure way after performing the steps of any standard mutual authentication and key establishment mechanism.

This discussed procedure is explained in Algorithm 5. A sequence diagram of all the phases related to the proposed

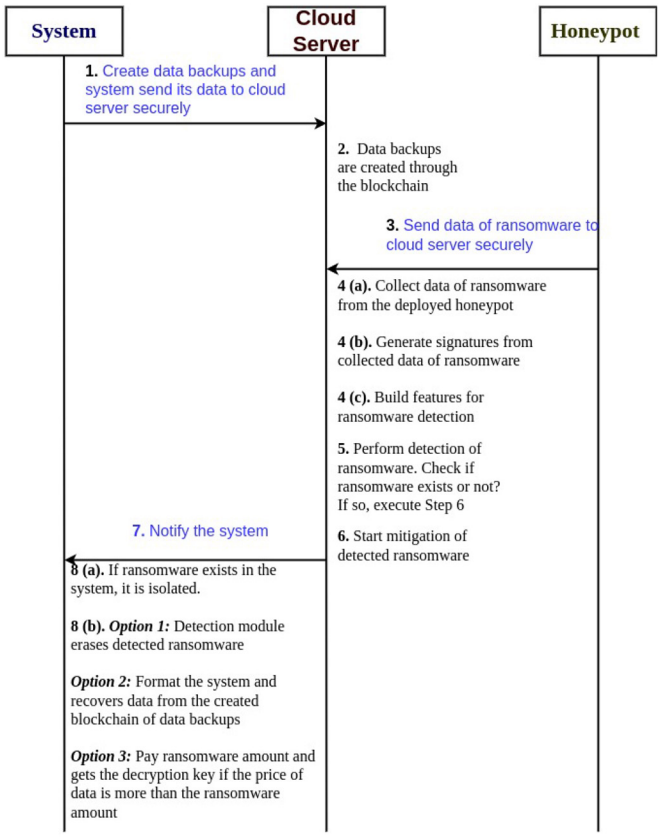


Fig. 3. Sequence diagram of the proposed BSFR-SH.

BSFR-SH is also depicted in Fig. 3. It summarizes the flow of execution of various phases related to the proposed BSFR-SH.

V. Security Analysis

In this section we provide the details of security analysis of BSFR-SH.

1) We have used the existing procedure of mutual authentication and key establishment for the secure exchange of information among various entities. After mutual authentication, a session key SK_{E_A,E_B} can be used for the secure exchange between two communicating entities E_A and E_B. Thus, the entities, like CS_l, HP_{RW} and SYS_i can securely exchange the information among them. As a result, in the absence of authentication and key establishment procedure, the entities will not be able to communicate securely. A sender encrypts the data with the session key and then transmits the encrypted information to the receiver. The receiver then receives the encrypted information and decrypts it with the established session key. In each exchange of messages, there should be a provision of the use of freshly generated timestamps and random secret nonces. The timestamps are verified at the receiver's end, whereas the random nonces are closely associated with an entity, who generates them. The received messages are then verified at the receiver's end to achieve the mutual authentication. If an entity computes a session key then that session key is also computed at the receiver's end for the cross verification. It is also under the consideration that session key should be computed with the short term secret values (i.e., random nonce values) and long term secret values (i.e., secret keys and pseudo-identities). This process provides to achieve the distinct keys in each different session for different entities. Therefore, an adversary $\mathcal{A}$ can not estimate the session key. This process provides protection against the replay attack, man-in-the-middle attack, impersonation attack, illegal session key computation, etc.
2) After the registration of entities, the registration credentials have been deleted from the database of the trusted registration authority. Therefore, a privileged insider user of the registration authority does not have access to the secret registration information of the entities. Furthermore, the secret credentials of the entities are stored in the secured regions of various devices, such as cloud servers. Hence, $\mathcal{A}$ can not launch other attacks including privileged insider, secret credential guessing and stolen verifier attacks on the proposed BSFR-SH.
3) It is already revealed that the attacks like 51% attack and selfish mining are possible on the blockchain based networks. Such attacks may occur in case if $\mathcal{A}$ has a large amount of hashing power [25]. The 51% attack needs that $\mathcal{A}$ should carry more than half of the hashing power. Particularly, 51% attack may be conducted against the cryptocurrencies in which $\mathcal{A}$ performs unauthorised tasks like, double-spending. Furthermore, the selfish mining is another well discovered security flaw in the blockchain based networks, which can be executed by malicious miners to theft block rewards. Recent exploitation proves that Proof-of-Work (PoW) consensus algorithm is unsafe against the 51% attack. In BSFR-SH, the practical byzantine fault tolerance (PBFT) consensus algorithm has been used, which is secured against these attacks. In some of the cases, it is observed that PBFT is vulnerable to sybil attack. However, such kind of attack can also mitigated when we deploy PBFT along with other consensus algorithms, i.e., PoW. Moreover, deployment of permissioned version of PBFT is also helpful to overcome the security issues related to sybil attack. Thus, BSFR-SH is safe against 51% attack, selfish mining, sybil attack and other potential attacks of the domain.
4) BSFR-SH may be useful in defending against different forms of threats. The entities, like cloud servers, keep secret information in secured areas of their databases to thwart any hacking attempts. A blockchain-based technique is also used in the BSFR-SH, which makes it more secure and tamper-proof. As a result, it can fight against a wide range of assaults, including denial-of-service (DoS) and various types of data manipulation and leakage attacks [26].
5) BSFR-SH works through the five different phases as discussed earlier. Apart from that the blockchain based mechanism is utilized for both ransomware detection and data recovery. The two different blockchains are used: a) one is for ransomware detection and b) other one is

TABLE II
PERFORMANCE COMPARISON OF DIFFERENT MECHANISMS

Technique	Accuracy	F1-score
Almashhadani *et al.* [11]	97.08	0.971
Hwang *et al.* [12]	97.30	0.973
Sharmeen *et al.* [13]	95.96	0.960
Bae *et al.* [14]	98.65	0.987
Proposed BSFR-SH	98.98	0.990

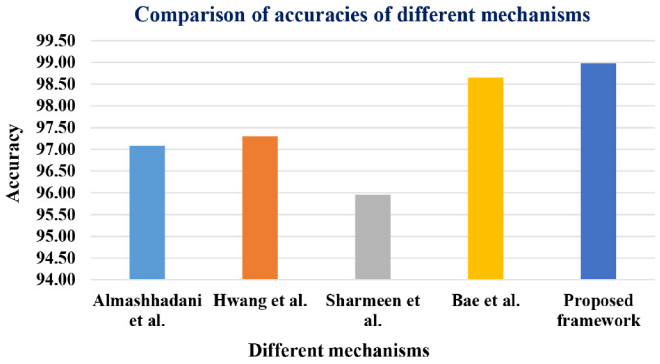


Fig. 4. Comparison of accuracies of different mechanisms.

for the data backups and recovery. Due the deployed blockchain based mechanism, data alteration and leakage are very difficult for $\mathcal{A}$. Hence, both "ransomware detection and mitigation" and "data backups and recovery" phases can be done effectively and securely.

VI. COMPARATIVE STUDY

In this section, we provide the details of the comparisons of the BSFR-SH and other similar frameworks. BSFR-SH is compared with the similar mechanisms, such as Almashhadani *et al.* [11], Hwang *et al.* [12], Sharmeen *et al.* [13], Bae *et al.* [14]. Here, we have considered two important parameters, such as accuracy and F1-score.

A. Comparison of Accuracy

The values of accuracy for Almashhadani *et al.* [11], Hwang *et al.* [12], Sharmeen *et al.* [13], Bae *et al.* [14] and BSFR-SH and are 97.08, 97.30, 95.96, 98.65, and 98.98, respectively. Here, it is important to mention that BSFR-SH performs well as compared to other similar mechanisms. The results are reported in Table II and Fig. 4.

B. Comparison of F1-Score

The values of F1-score values for Almashhadani *et al.* [11], Hwang *et al.* [12], Sharmeen *et al.* [13], Bae *et al.* [14] and BSFR-SH and are 0.971, 0.973, 0.960, 0.987 and 0.990, respectively. BSFR-SH performs well as compared to other similar mechanisms as it has achieved high F1-score values which are reported in Table II and Fig. 5.

VII. PRACTICAL IMPLEMENTATION

In this section, we provide the details of the practical implementation of BSFR-SH. The test was run on a Windows 11,

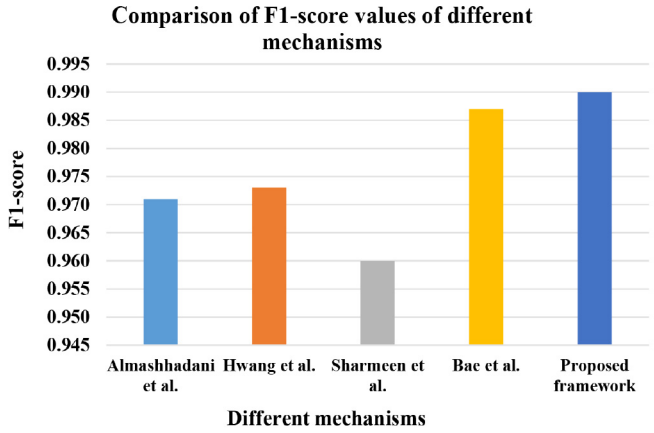


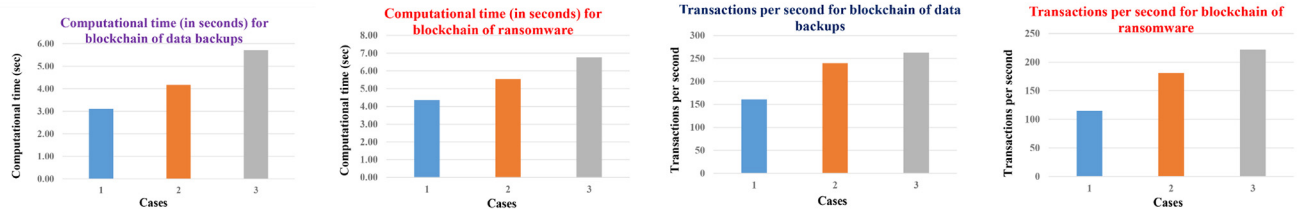
Fig. 5. Comparison of F1-score values of different mechanisms.

64-bit system with an Intel (R) core i5 9th gen @2.40GHz and Nvidia 1650 4GB graphics. The random-access memory (RAM) was 8 GB in size. Eclipse IDE 2019-12 was used as the development platform, and the Java language was used with four miner nodes in each scenario (i.e., cloud server). Two blockchains were implemented: a) one for ransomware detection, and b) other one for data backups and recovery. For blockchain mining, a voting-based system was employed with PBFT [26]. We have taken three different cases: case-1, case-2 and case-3. In the considered cases, there were 5, 10, and 15 blocks were mined respectively and each block contains 100 transactions. For the machine learning part, we tried with the random forest, logistic regression, decision tree and k-nearest neighbor (KNN) algorithms. The mechanism of private blockchain was considered.

We have taken "BitcoinHeist Ransomware Address Dataset", which is available at UCI machine learning repository [15]. This dataset contains address features on the heterogeneous Bitcoin network to identify ransomware payments. It has features, like data Set characteristics: multivariate, time-Series, number of Instances: 2916697, and number of attributes: 10. It has attributes like address: string-bitcoin address, year: integer, day: integer, length: Integer, weight: float, count: integer, looped: integer, neighbors: integer, income: integer and label: category string, and the name of the ransomware family (e.g., Cryptxxx, cryptolocker etc.,) or white (i.e., not known to be ransomware). There are total of 2,916,697 transactions where 2,875,284 "legitimate" and the remaining 41,413 are "ransomware". For the performance comparison, we have taken 90% attackers (ransomware) and 10% benign samples. Note that it may cause more false positive rates (FPR). In future, we would like to take more variants of attackers (ransomware) and benign samples in this study.

A. Estimation of Accuracy

Accuracy or classification accuracy is important performance parameters of an ransomware detection system. It is the ratio of number of correct predictions to the total number of input samples. Generally, it lies between



(a) (b) (c) (d)

Fig. 6. (a) Computational cost for blockchain of data backups (b) Computational cost for blockchain of ransomware (c) Transaction per second (TPS) for blockchain of data backups (d) Transaction per second (TPS) for blockchain of ransomware.

$0-100\%$. It should be as high as possible. In the deployed mechanism, we have tried with four algorithms, i.e., random forest, logistic regression, decision tree and KNN algorithms. However, the best accuracy value that we achieved was 98.98% through the decision tree method.

B. Estimation of F1-Score

The F1- score (also known as the F1- measure or F measure) is a test accuracy metric that is defined as the weighted harmonic mean of the test's precision and recall. What proportion of positive identifications was actually correct is considered as precision. However, what proportion of actual positives was identified correctly is considered as recall. Generally, it lies between $0-1$. It should be as high as possible. In the deployed mechanism, we have tried with four algorithms, i.e., random forest, logistic regression, decision tree and KNN algorithms. However, the best accuracy value that we achieved was 0.990 through the decision tree method.

C. Estimation of Computational Time

For all scenarios investigated, the effect of increasing number of systems was calculated as the computation time (in seconds). For case-1, case-2, and case-3, the estimated computational time (in seconds) are 3.10, 4.17, and 5.71 respectively for "blockchain of data backups". Fig. 6(a) shows the computational time values of "blockchain of data backups". Moreover, for "blockchain of ransomware", the estimated computational time (in seconds) are 4.36, 5.54, and 6.76, in case-1, case-2, and case-3, respectively. Fig. 6(b) shows the computational time values of "blockchain of ransomware". It is worth noting that the "computational time" rises with the increasing "number of devices" from case-1 to case-2 and case-2 to case-3, owing to the "creation and adding (mining) of more number of blocks in the blockchain".

D. Estimation of Transactions per Second

The impact of the BSFR-SH on transaction per second (TPS) is also estimated for each of the scenarios. For case-1, case-2, and case-3, the transaction per second (TPS) values are 161, 240 and 263 respectively for "blockchain of data backups". Fig. 6(c) shows the transactions per second values of "blockchain of data backups". Moreover, for "blockchain of ransomware", the estimated transactions per second are 115, 181, and 222, in case-1, case-2, and case-3, respectively. Fig. 6(d) shows the transactions per second values of "blockchain of ransomware". It is worth noting that when the blockchain develops in size and number of smart healthcare devices and users, the value of TPS rises, because this results in the "creation and addition (mining) of more blocks in the blockchain".

VIII. Conclusion

A blockchain enabled security framework to detect and defend the ransomware attacks for smart healthcare was presented. The conducted security analysis proves the security of BSFR-SH against the ransomware attacks. The performance of the proposed BSFR-SH was better than the other similar mechanisms as it achieved better accuracy and F1-score than the other methods. Further, the practical demonstration of BSFR-SH was provided to measure the impact of BSFR-SH on the important parameters, like accuracy, F1-score, computational time and transactions per second.

In future, we would like to add more functionality features in the BSFR-SH. Moreover, we would also try to increase the accuracy of the presented framework without degrading the security of the scheme.

Acknowledgment

The authors would like to thank the anonymous reviewers and the Associate Editor for their valuable comments and suggestions which helped them to improve the presentation and quality of the paper.

References

[1] S. S. Chakkaravarthy, D. Sangeetha, M. V. Cruz, V. Vaidehi, and B. Raman, "Design of intrusion detection honeypot using social leopard algorithm to detect IoT ransomware attacks," *IEEE Access*, vol. 8, pp. 169944–169956, 2020.

[2] M. Wazid, A. K. Das, V. Odelu, N. Kumar, and W. Susilo, "Secure remote user authenticated key establishment protocol for smart home environment," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 2, pp. 391–406, Mar./Apr. 2020.

[3] S. Tian, W. Yang, J. M. L. Grange, P. Wang, W. Huang, and Z. Ye, "Smart healthcare: Making medical care more intelligent," *Global Health J.*, vol. 3, no. 3, pp. 62–65, 2019.

[4] E. Berrueta, D. Morato, E. Magana, and M. Izal, "A survey on detection techniques for cryptographic ransomware," *IEEE Access*, vol. 7, pp. 144925–144944, 2019.

[5] D. Farhat and M. S. Awan, "A brief survey on ransomware with the perspective of Internet security threat reports," in *Proc. 9th Int. Symp. Digit. Forensics Security (ISDFS)*, Elazig, Turkey, 2021, pp. 1–6.

[6] H.-N. Dai, Z. Zheng, and Y. Zhang, "Blockchain for Internet of Things: A survey," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8076–8094, Oct. 2019.
[7] M. A. Ferrag, M. Derdour, M. Mukherjee, A. Derhab, L. Maglaras, and H. Janicke, "Blockchain technologies for the Internet of Things: Research issues and challenges," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2188–2204, Apr. 2019.
[8] M. A. Ferrag and L. Shu, "The performance evaluation of blockchain-based security and privacy systems for the Internet of Things: A tutorial," *IEEE Internet Things J.*, vol. 8, no. 24, pp. 17236–17260, Dec. 2021.
[9] J. Zhao, R. Masood, and S. Seneviratne, "A review of computer vision methods in network security," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1838–1878, 3rd Quart., 2021.
[10] "Ransomware." Imperva. Accessed: Jan. 2022. [Online]. Available: https://www.imperva.com/learn/application-security/ransomware/
[11] A. O. Almashhadani, M. Kaiiali, S. Sezer, and P. O'Kane, "A multi-classifier network-based crypto ransomware detection system: A case study of locky ransomware," *IEEE Access*, vol. 7, pp. 47053–47067, 2019.
[12] J. Hwang, J. Kim, S. Lee, and K. Kim, "Two-stage ransomware detection using dynamic analysis and machine learning techniques," *Wireless Personal Commun.*, vol. 112, no. 4, pp. 2597–2609, 2020.
[13] S. Sharmeen, Y. A. Ahmed, S. Huda, B. S. Kocer, and M. M. Hassan, "Avoiding future digital extortion through robust protection against ransomware threats using deep learning based adaptive approaches," *IEEE Access*, vol. 8, pp. 24522–24534, 2020.
[14] S. I. Bae, G. B. Lee, and E. G. Im, "Ransomware detection using machine learning algorithms," *Concurrency Comput. Pract. Exp.*, vol. 32, no. 18, pp. 1–11, 2020.
[15] C. G. Akcora, Y. Li, Y. R. Gel, and M. Kantarcioglu, "BitcoinHeist: Topological data analysis for ransomware prediction on the bitcoin blockchain," in *Proc. 29th Int. Joint Conf. Artif. Intell. (IJCAI)*, Yokohama, Japan, 2021, pp. 4439–4445.
[16] D. Min *et al.*, "Amoeba: An autonomous backup and recovery ssd for ransomware attack defense," *IEEE Comput. Archit. Lett.*, vol. 17, no. 2, pp. 245–248, Jul.–Dec. 2018.
[17] X. Zhang, J. Wang, and S. Zhu, "Dual generative adversarial networks based unknown encryption ransomware attack detection," *IEEE Access*, vol. 10, pp. 900–913, 2022.
[18] S. Baek, Y. Jung, D. Mohaisen, S. Lee, and D. Nyang, "SSD-assisted ransomware detection and data recovery techniques," *IEEE Trans. Comput.*, vol. 70, no. 10, pp. 1762–1776, Oct. 2021.
[19] D. Min, Y. Ko, R. Walker, J. Lee, and Y. Kim, "A content-based ransomware detection and backup solid-state drive for ransomware defense," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 41, no. 7, pp. 2038–2051, Jul. 2022.
[20] S. Poudyal and D. Dasgupta, "Analysis of crypto-ransomware using ML-based multi-level profiling," *IEEE Access*, vol. 9, pp. 122532–122547, 2021.
[21] P. Bajpai and R. Enbody, "Attacking key management in ransomware," *IT Prof.*, vol. 22, no. 2, pp. 21–27, 2020.
[22] M. Shah. "Blockchains and NFTs in identification and security protocols." Accessed: Apr. 2022. [Online]. Available: https://www.forbes.com/sites/forbestechcouncil/2022/02/23/blockchains-and-nfts-in-identification-and-security-protocols/?sh=5bc80aad14fc
[23] D. Johnson, A. Menezes, and S. Vanstone, "The elliptic curve digital signature algorithm (ECDSA)," *Int. J. Inf. Security*, vol. 1, no. 1, pp. 36–63, 2001.
[24] M. Castro and B. Liskov, "Practical Byzantine fault tolerance and proactive recovery," *ACM Trans. Comput. Syst.*, vol. 20, no. 4, pp. 398–461, 2002.
[25] S. Sayeed and H. Marco-Gisbert, "Assessing blockchain consensus and security mechanisms against the % attack," *Appl. Sci.*, vol. 9, no. 9, pp. 1–17, 2019. [Online]. Available: https://www.mdpi.com/2076-3417/9/9/1788
[26] M. Wazid, A. K. Das, R. Hussain, N. Kumar, and S. Roy, "BUAKA-CS: Blockchain-enabled user authentication and key agreement scheme for crowdsourcing system," *J. Syst. Archit.*, vol. 123, pp. 1–16, Feb. 2022.

Mohammad Wazid (Senior Member, IEEE) received the Master of Technology degree in computer network engineering from Graphic Era University, Dehradun, India, and the Ph.D. degree in computer science and engineering from the International Institute of Information Technology Hyderabad, Hyderabad, India. He is currently working as an Associate Professor with the Department of Computer Science and Engineering, Graphic Era University, Dehradun, where he is the Head of the Cybersecurity and IIoT Research Group. Prior to this, he was an Assistant Professor with the Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal, India. He was also a Postdoctoral Researcher with the Cyber Security and Networks Lab, Innopolis University, Innopolis, Russia. His current research interests include security, remote user authentication, the Internet of Things (IoT), and cloud computing. He has published more than 100 papers in international journals and conferences in the above areas. He was a recipient of the University Gold Medal and the Young Scientist Award from UCOST, the Department of Science and Technology, Government of Uttarakhand, India.

Ashok Kumar Das (Senior Member, IEEE) received the M.Sc. degree in mathematics, the M.Tech. degree in computer science and data processing, and the Ph.D. degree in computer science and engineering from the Indian Institute of Technology Kharagpur, India. He is currently an Associate Professor with the Center for Security, Theory and Algorithmic Research, Indian Institute of Technology Hyderabad, Hyderabad, India. He was also a Visiting Faculty with the Virginia Modeling, Analysis and Simulation Center, Old Dominion University, Suffolk, VA, USA. His Google Scholar H-index is 70 and i10-index is 202 with over 13 800 citations. His research interests include cryptography, system and network security, blockchain, security in Internet of Things, Internet of Vehicles, Internet of Drones, smart grids, smart city, cloud/fog computing, intrusion detection, and AI/ML security. He has authored over 320 papers in international journals and conferences in the above areas, including over 275 reputed journal papers. He was a recipient of the Institute Silver Medal from the Indian Institutes of Technology Kharagpur. He is/was on the editorial board of IEEE SYSTEMS JOURNAL, *Journal of Network and Computer Applications* (Elsevier), *Computer Communications* (Elsevier), *Journal of Cloud Computing* (Springer), *Cyber Security and Applications* (Elsevier), *IET Communications*, *KSII Transactions on Internet and Information Systems*, and *International Journal of Internet Technology and Secured Transactions* (Inderscience). He also severed as one of the Technical Program Committee Chairs for the first International Congress on Blockchain and Applications (BLOCKCHAIN'19), Avila, Spain, June 2019, International Conference on Applied Soft Computing and Communication Networks (ACN'20), Chennai, India, October 2020, and second International Congress on Blockchain and Applications (BLOCKCHAIN'20), L'Aquila, Italy, October 2020.

Sachin Shetty (Senior Member, IEEE) received the Ph.D. degree in modeling and simulation from Old Dominion University in 2007. He was an Associate Professor with the Electrical and Computer Engineering Department, Tennessee State University, USA. He is currently an Associate Professor with the Virginia Modeling, Analysis and Simulation Center, Old Dominion University. He holds a joint appointment with the Department of Modeling, Simulation and Visualization Engineering and the Center for Cybersecurity Education and Research. He has authored and coauthored over 125 research articles in journals and conference proceedings and two books. His research interests lie at the intersection of computer networking, network security, and machine learning. He was a recipient of the DHS Scientific Leadership Award. He has served on the Technical Program Committee of ACM CCS, IEEE INFOCOM, IEEE ICDCN, and IEEE ICCCN.