Applied Network Science

REVIEW Open Access

Security through block vault in a blockchain enabled federated cloud framework

Olumide Malomo*, Danda Rawat† and Moses Garuba†

*Correspondence:
olumide.malomo@howard.edu
†Danda Rawat and Moses Garuba contributed equally to this work.
Department of Electrical Engineering and Computer Science College of Engineering and Architecture Howard University, Washington, DC, USA

Abstract
The survivability of any organization in the event of disaster or attack could greatly depend on its offsite data recovery. But which provider could offer secure and resilient cyber security protection to offsite data is a major problem for individuals, businesses and organizations. In recent times, cyber-attacks are strategic and tactical. Adversaries are advanced with capability to access sensitive business digital assets. Their nefarious actions have exposed devastating flaws in many access controls at different levels, and are now security risk to systems, data storages and resources. In this paper, we present a Blockchain enabled federated cloud computing framework, to secure storage for offsite digital assets. Our framework is designed with a great degree of efficiency, privacy, scalability and restricted access control. It addresses authentication flaws and improve early detection of data breaches, by continuously evaluating subject's access control and interaction with resources, using operation cost for monitoring and accountability. Evaluation showed proof of concept that our design and approach outperforms the traditional approaches.

Keywords: Federated cloud, Blockchain, Blockchain federated cloud computing, Cloud computing, Access control, Security and privacy, Offsite data, Cybersecurity, Federated cloud computing security

Introduction

Cloud based attacks are increasingly leveraging on poor security practices and vulnerabilities, partly from cloud users, developers and service providers. As cloud computing services broaden, a broad array of security issues and risks are presented for cybercriminals and malicious insiders to exploit. Cloud-based attacks, such as data threats, weak cryptography, shared technology vulnerabilities, cloud Application Programming Interface (API) vulnerabilities and vulnerable cloud services are making it difficult for security experts to determine what to control or defend against (Cai et al. 2018; Malomo et al. 2018; Rawat et al. 2017). Among several cloud based cyber-attacks, data threats are proving difficult and challenging to defend against by security experts. The reason being that the increasing trend in the importance of data to businesses for decision making process; exposing variability and optimizing profitability; is strongly reflecting more on the popularity of data storage infrastructure technologies and replication techniques. As a result, cybercriminals and malicious insiders' interests have now shifted to data storage (Cai et al. 2018; Malomo et al. 2018).

The general concerns are that cyber security risks on private and public cloud infrastructure, data storage and end user devices continue to be on the rise without abating (Sabillon et al. 2016; Saridakis et al. 2016). In recent times, cyber-attacks by fraudsters and hackers against individuals, businesses, and organizations are advanced, persistent, sophisticated and proliferating causing unrest (Conteh and Royer 2016; Course 2016). They are nefariously after corporate sensitive data and trade secrets, higher institutions cutting edge research and development, and individuals' sensitive identity and personal information (Ettredge et al. 2018; Sabillon et al. 2016; Yu 2015). These, among many external threats and the biggest, insider threats, coupled with access controls flaws, complications and challenges are some reasons not allowing some businesses and organizations to transition their sensitive data to cloud storage. In 2017, transportation and healthcare industries globally were attacked by ransomware WannaCry, which ripped through several control systems and computing devices on the Internet and encrypted their files (Ehrenfeld 2017; Tabone 2017). This attack sent a global shock wave to the entire world that created public awareness. It was an awareness that waken cybercriminals consciousness to the ease of such a lucrative attack. Since then, the threat to end users and businesses (large and small) across the globe is rapidly increasing. Because of the risen popularity of bitcoin serving as vehicle for ransom payments, criminals are attracted to this cyber extortion business; making ransomware crime the biggest threats of the 21st century to digital security (Kshetri and Voas 2017). In fact, cybercriminals have grown incredibly bold that they now have crimeware packages, in ransomware-as-a-service (RaaS) business model put up for sale on their dark websites.

There are reports on the magnitude of ransonware attacks, showing over 4,000 attacks happen on daily basis, and close to 50,000 computing devices are infected monthly. The financial implications are depressing, and ransom payments is forecast to hit well over a billion dollars at the end of this year (Jarvis 2017). Although the financial impact is huge, what is more worrisome is the business impact. Unfortunately, when exploit such as this is discovered, the awareness to the public can pose a serious major threat to business reputation, profitability and survivability. Several studies have shown that losses caused by cyber attack always have huge impact both tangible and non-tangible on victims, and in some cases apart from operational disruption depending on the level of impact, lives could be at risks. Thus, the attack calls into question, the several different layers of security solutions in existence, that should have prevented these cyber and data security breaches. Bearing in mind, there is a possibility that any of the security solutions may likely fail to mitigate damage from future cyber-attacks. This attack and other similar advanced exploits pose some serious questions that need urgent answers: What was the main target of interest? What could businesses do without paying the ransomware demand?

We know that the main target of ransomware criminals, and like any other cybercrimes is the organization's sensitive data. Critical business data is the crown jewels and life blood of any organizations. offsite data storage (data vault) without syncing would have saved businesses from paying the ransomware demand. Avoiding syncing eliminates the vulnerability of any file sharing engine synchronizing local data storage to the cloud storage, thereby corrupting the backup copy on cloud, which is an offensive tactical strategy cybercriminals are using effectively to advance their asymmetric advantage. The backup rule of 3-2-1 for disaster recovery plan suffice here, which means make three copies of backup, in two formats or store internally, and ensure that one copy is stored in an external storage device or offsite, like in the cloud for easy access in the event of natural disaster. However, offsite data storage has some concerns, such as unsatisfactory protection of sensitive data gives cause for concern. Obviously, the cloud is not short of storage providers, but which provider could offer secure and resilient cyber security protection to offsite data is a major problem for individual, businesses and organizations.

Survey report has shown that there are on-going studies to address some of the major concerns of data storage/vault, such as loss of control over sensitive data and management of the cryptographic keys (Malomo et al. 2018). These concerns have raised fresh questions that we can address: How can the vault security be built to protect against man-made internal and external threats, by implementing cybersecurity critical controls such as multi-authentication factor, identification fingerprint, separation of duties, and split knowledge, etc.? What will be the safe machine hardware component metrics for system identification fingerprint? In this paper, we design, develop and evaluate block vault in a Blockchain-enabled Federated Cloud Computing (BFC^2) framework to secure storage for offsite digital assets and reduce data breaches. Figure 1 shows a typical Blockchain enabled Federated-Cloud Framework where participating organizations in a federated cloud environment use their block managers for sharing the information with other participants in the consortium (Malomo et al. 2018).

This paper is an extension of the work we presented in Malomo et al. (2018). Our main contributions of this paper are:

- How Blockchain and federated cloud computing technologies are explored and integrated to create an intelligence community of technology systems and security experts that can provide unique services that secure cyber infrastructure and vault storage for offsite digital assets.

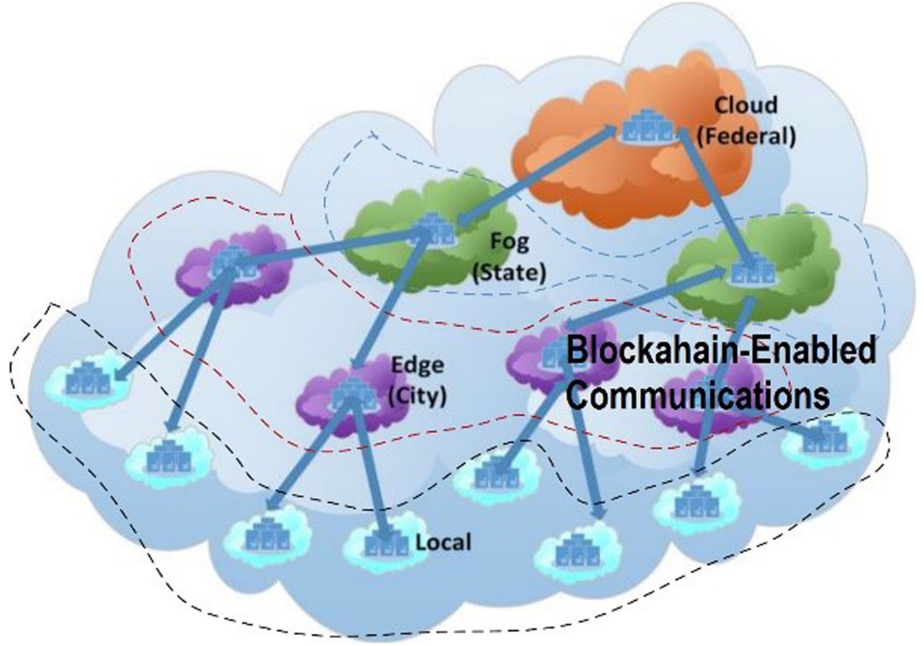

Fig. 1 Blockchain within Federated Cloud Architecture (Malomo et al. 2018)

- We designed and developed a framework to secure storage for offsite digital assets with a great degree of efficiency, privacy, scalability and restricted access control that improve authentication and early detection of breaches.
- We proposed vault security access control approach that protect against man-made internal and external threats implementing cybersecurity critical controls such as multi-authentication factor, identification fingerprint, separation of duties, and split knowledge, etc.
- We identified the safe machine hardware component metrics for system identification fingerprint.
- Introduced an approach in the model for continuous monitoring and accountability of subject's interaction with resources using amortized analysis concept and shrewd accounting principles to guarantee no data and cyber breaches occur.
- Evaluation and proof of concept that our design and approach can securely and intelligently control access to systems and resources.

The rest of this paper is organized as follows: "Background and related work" section describes background, related works and presents briefly, BFC^2 Block Vault and Block Generator interconnectivity. In "The proposed block vault in BFC^2" section, we present BFC^2 Block Vault strong and resilient proposed approach to secure storage for off-site digital assets with a great degree of efficiency, privacy, scalability and restricted cybersecurity access control. In "Performance evaluation and discussion" section, we present the performance evaluation and discussion. Finally, our conclusion is presented in "Conclusion" section.

Background and related work

The security threats continue to evolve from infrastructure networks and systems to sensitive business critical data. Today's threats are against business processes and components that make-up the entire business process. Nothing is sacred to attackers, including off-site data storage and physical security. There has been a growing and scaring trend in the sophistication of these threats as technology advances. In fact, the threats landscape is changing, with still many unknown crimewares in the wild; managing risk continues to prove difficult due to lack of information, which involves the understanding of threats, vulnerabilities and potential attacks. Studies have shown most security solutions have been either single security mechanism or technology, deployed to manage some specific risks (Malomo et al. 2018). It has been observed that some introduced complexity and other risk exposures (Malomo et al. 2018).

The risk to business process must be effectively and efficiently managed. Therefore, the security defense to deploy must have an asymmetric advantage over known and unknown advanced persistent tactics and strategies used by nefarious criminals. The defense mechanism must have an approach to constant monitoring and analyzing threats; information and knowledge gathering and sharing. Built as an adaptive, scalable and resilient cyber defense capable to protect business process and its components. A holistic approach is to strategically have a mix of proven technology innovations such as federated cloud computing and Blockchain technologies, combine with intelligent processes (Malomo et al. 2018; Sayeed and Marco-Gisbert 2019). The Federated cloud computing will expand the scope of cloud services, by pooling together resources from related cloud infrastructures.

While the Blockchain technology security power will provide the trust and transparency of data for accountability required among cloud infrastructure providers, by providing assured data provenance in the federated cloud. Hence, an adaptive cyber defense will be a blockchain enabled federated cloud computing mechanism built around security experts and business professionals, that can span across continents.

BFC^2 framework is a proposed architecture providing security and privacy, scalability and restricted access control. BFC^2 contains three sub-systems: Block Generator, Block Vault, and Threatroscope as shown in Fig. 2. BFC^2 focuses mainly on two aspects: Block vault enhances secure storage security and access control to restricted offsite digital assets, reinforce with Blockchain technology components to secure storage for offsite digital assets, which is the main focus of this paper; Threatroscope has been presented in an earlier publication (Malomo et al. 2018); addresses and improves Breach Detection Gap (BDG) by continuously monitoring, and analyzing systems and networks traffics, against data breaches and cyber-attacks; using Dempster Shafer Theory (DST) as basis to build evidences of probably attacks in federated cloud computing environment (Malomo et al. 2018).

The block generator is indispensable to the BFC^2 design. It is responsible for handling essential functions of BFC^2 operations (Malomo et al. 2018). It is built on the original purpose of federated blockchain, architecture concept that allows digital documents for private business purposes to be digitally signed and orderly added in both distributed and centralized ledgers (Underwood 2016). The architecture is different from Public Blockchain such as Bitcoin, but management is similar to Ethereum (Crosby et al. 2016; Underwood 2016). It is also different from Private Blockchain because of the number of cloud center (node) service providers controlling the federated network. The block generator connects the other two systems: block vault and threatroscope, and handles processes for accepting transactions validated for ledger update using Federated-Proof-of Stake (FPoS) protocol, which is different from the bitcoin miners and Ethereum Proof-of-Stake (Baliga 2017; Malomo et al. 2018; Swanson 2015).

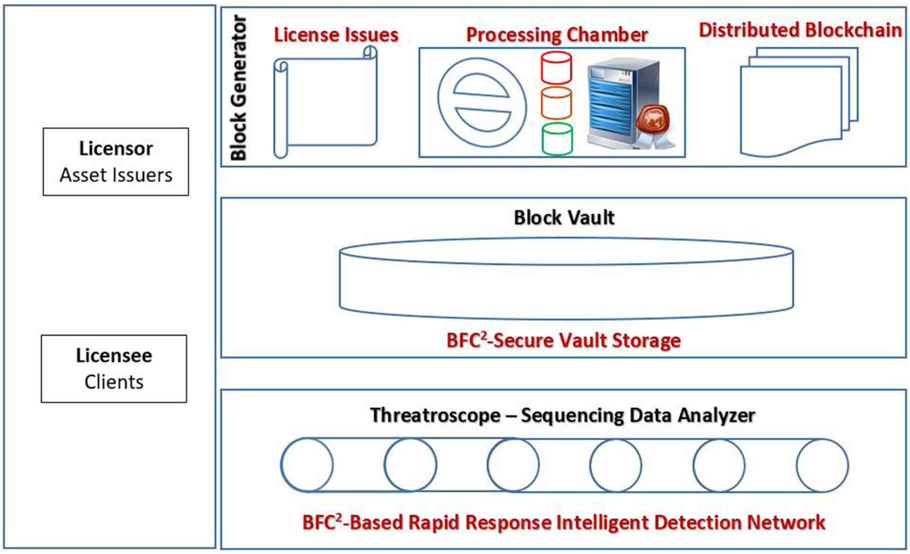
Licensor
Asset Issuers
Licensee
Clients
Block Generator
License Issues
Processing Chamber
Distributed Blockchain
Block Vault
BFC²-Secure Vault Storage
Threatroscope – Sequencing Data Analyzer
BFC²-Based Rapid Response Intelligent Detection Network

Fig. 2 BFC^2 architecture (Malomo et al. 2018)

We have introduced Block Generator and Threatroscope and their utmost importance to the whole design in an earlier publication “next generation cybersecurity through a blockchain enabled federated cloud framework” (Malomo et al. 2018). In that paper, we described our proposed framework BFC^2 with reference to Blockchain technology. We touched on their design essence, how the block generator works and areas of difference with traditional blockchain technology. We described Threatroscope technical analysis with evidence of effectiveness and efficiency. How information exchange performance is affected by DST and the consensus protocol. Also, we touched on smart contract and its purpose for client on-boarding process into the Blockchain system, which is very important and requires that client joining the network has to go through the process. We expressed our concerns on possible ways the system could be attacked and we outlined our recommendations.

BFC^2 vault and generator interconnectivity

BFC^2 system model is a permissioned Federated Blockchain (Malomo et al. 2018) for privacy and restricted access control, which provides the requirements for enterprise that wants rigid security and environment that is fraud free (Malomo et al. 2018). The cloud centers (nodes) managed by service providers are known as Licensors/Validators V_L on the block generator, they run and control the federated network (Malomo et al. 2018). Introduction of new client/customer C_L (referred to as anonymous/Licensee) to the federated network is by client’s service provider (Licensor/Validator V_L) initiating smart contracts for client to on-board on the blockchain system (Malomo et al. 2018). On the block vault system, service provider that introduced/initiated client to the network is referred to as Client-Intro-Manager (CIM), and is client’s representative and contact on the network. It is required because, in block vault, the access control requires block generator to coordinate request information between the clients and their respective Client-Intro-Managers (CIMs) and other Licensors (Malomo et al. 2018). The information in the client smart contract is important as they contain the terms of agreement, rules and parameters that determine how client operate and interact with resources within the network (Malomo et al. 2018). Consensus protocol is Federated-Proof-of-Stake (FPoS) which is based on thresholds of number of Block Signers (BS) and the number of signatures (REQ) that is required to reach a consensus on several critical and important decisions pertaining to the security and operation of the federation is discussed in Malomo et al. (2018). Amongst what the block generator does is to assign new client consensusly accepted to the network a unique federated access identification $C_{L\text{-}ID}$, finalizes other client’s on-boarding requirements and notify client’s CIM (Malomo et al. 2018).

For better comprehension and without limiting the overall scope of this paper or resorting to cumbersome reference to the part publication (Malomo et al. 2018), below are some notations/terms (keys, credentials) used to give an explicit understanding and information on the BFC^2 block vault operations:

- ***Assets:*** An asset (digital asset) is the client’s electronic offsite data either to be stored/upload (after compressed and encrypted) or retrieved from block vault.
- ***Smart Contract:*** It’s the binding contract for a license agreement between the licensor and licensee on the system. It contains information and parameters that determine client’s requirements and interaction with resources (e.g. if client requires

multiple signing for authentication or notification at any time before access is granted, etc)

- ***Participants:*** These are service providers referred to as Licensor, Validator (V_L) or Client Intro Manager (CIM). Each provider has a private key ($VKey_{PR}$), public key ($VKey_{PUB}$), and CIM has a share key $CIMKey_{SHARE}$ used in the cryptographic functions: $Sign_{FUNC}()$, $Verify_{FUNC}()$ and $Hash_{FUNC}()$. Also, CIM has a special key (CIM_{SpKey}) used for key stretching encryption with Generator to encrypt highly sensitive blockchain records.
- ***Clients:*** Individuals, Businesses and Organizations with service agreements with service providers. They are referred to as Licensee. Each member must have unique federated access identification ($C_{L\text{-}ID}$) after the on-board smart contract procedure. Each client has private key ($CKey_{PR}$) and public key ($CKey_{PUB}$) for cryptography.
- ***Generator:*** The block generator or simple Generator is the brain of the system and coordinates major activities. Has its private key ($GenKey_{PR}$), public key ($GenKey_{PUB}$), share key ($GenKey_{SHARE}$) and special key (Gen_{SpKey}) for key stretching encryption with CIM to encrypt certain sensitive blockchain records.
- ***Buffers:*** These are tables in the system that are frequently read and rarely updated.
- ***Auditors or Regulators:*** The auditor oversees audit and corporate governance of the system. A provision is made for a regulator should there be government directives or legislation that will require access. Both the auditors and regulators are good options for maintaining governance and compliance. However, they cannot conduct any transactions.

The proposed block vault in BFC^2

The block vault is for clients that want secure storage for their offsite digital assets. The security is built to protect against man-made internal and external threats. First level of security is that clients' offsite data in storage are encrypted by themselves using their personal encryption/password key, anything outside the federation's cryptographic key management system. However, access controls is multi-factor authentication; designed from a general view of separation of duties. From the client side, access is viewed as external request from an authorized user, and therefore requires dual control; the involvement of the CIM access credentials among other metrics before access is granted. From the federated side where there may be occasional need for performing maintenance by internal staff, access requires split knowledge; at least THREE Licensors (Validators) randomly selected (and rotational) are involved in adding their access credentials before internal staff is granted access. Over and above, in both cases, the Generator input is required. However, security can further be harden by having the Generator's crypto-key pairs updated daily. But for now, the cost benefit analysis of the feasibility as against the crypto-key pairs daily update computational time and complexity on operations does not justify the need. It would be an interesting subject to be considered in a follow-up paper, to add another layer of security and encourage a daily access token. However, a periodic update in accordance to the federation's security rules and policies is acceptable for now. Figure 3 gives graphical representation of access into block vault.

First, let's discuss block vault approach in BFC^2 to gain an insight into federated cloud computing environment, and what it offers that allows cloud service providers

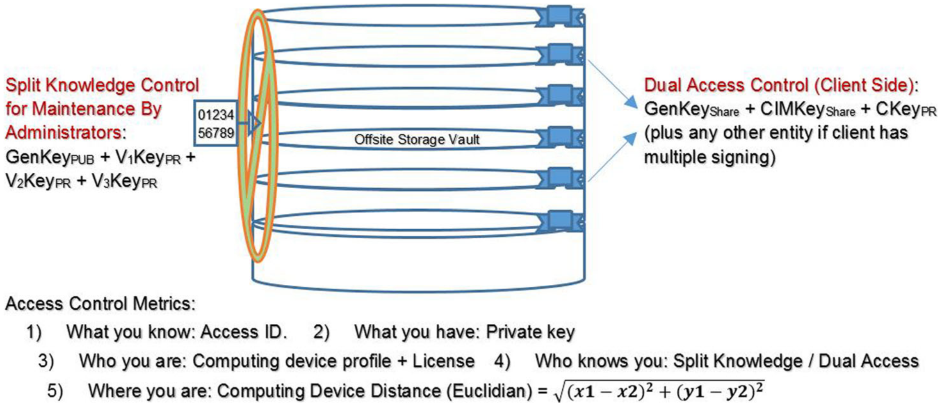
Split Knowledge Control for Maintenance By Administrators:
GenKey$_{PUB}$ + V$_1$Key$_{PR}$ + V$_2$Key$_{PR}$ + V$_3$Key$_{PR}$
01234
56789
Offsite Storage Vault
Dual Access Control (Client Side):
GenKey$_{Share}$ + CIMKey$_{Share}$ + CKey$_{PR}$
(plus any other entity if client has multiple signing)
Access Control Metrics:
1) What you know: Access ID. 2) What you have: Private key
3) Who you are: Computing device profile + License 4) Who knows you: Split Knowledge / Dual Access
5) Where you are: Computing Device Distance (Euclidian) = $\sqrt{(x1-x2)^2+(y1-y2)^2}$

Fig. 3 Block vault access metrics

to pool resources together to efficiently deliver quality vault service using locations; different hub bases that are scalable and efficient for digital assets storage within the federation.

Block vault approach in BFC^2

For better understanding of the BFC^2 block vault. Let us consider an image (.ISO) of an entire kali server that we want to move to the cloud block vault. First, the .ISO file runs through a compression program that splits the file based on the file size and the split volume/byte choice. For our discussion, we shall assume file will be split into three parts; say: xyz.c.001, xyx.c.002, and xyz.c.003. (Please note that simple ARJ or 7z compression works fine. It is better to compress and split files first to remove redundant character strings before encryption). The split compressed files are encrypted as first layer of security protection, using client personal password as the encryption key to prepare files (compressed, split, and encrypted) for offsite data upload. At this point, these files are referred to as digital assets. Last thing in this phase, client's digital assets with file-tag name and meaningful description.

The decision on how many files to split the compressed file to, is the choice of the federation. It is important, however, the distribution of the assets would impact on the number of vault locations required for upload. Figure 4 depicts the concepts of the locations of the vaults for assets storage.

We decided to have it split into three. A unique conventional name is assigned by the Generator to a restricted folder representing the vault on the file server at the edge centers, where file would be loaded and can be retrieved when needed. About the vaulting of the digital assets, the security is to keep the first file xyz.c.001 with the CIM edge cloud center, say Edge$_{\text{HU-1}}$, then xyz.c.002 goes to CIM state cloud center State$_{\text{HU-1}}$, and the third file xyz.c.003 location is determined by the block Generator, using greedy algorithm by selecting the cloud center under another state in a rational order and proximity that will be Edge$_{\text{HU-6}}$. The question is how do we track each files and link them? The Generator collate into a file, which we shall refer to as *BFC-BV-Tag-Txn-No*, the detailed information such as client's federated identification C$_{\text{L-ID}}$, filename and description client gave, path to file-level and share-level security location in each cloud center where the files are stored, and the summary transaction, which we shall discuss later. After all digital assets

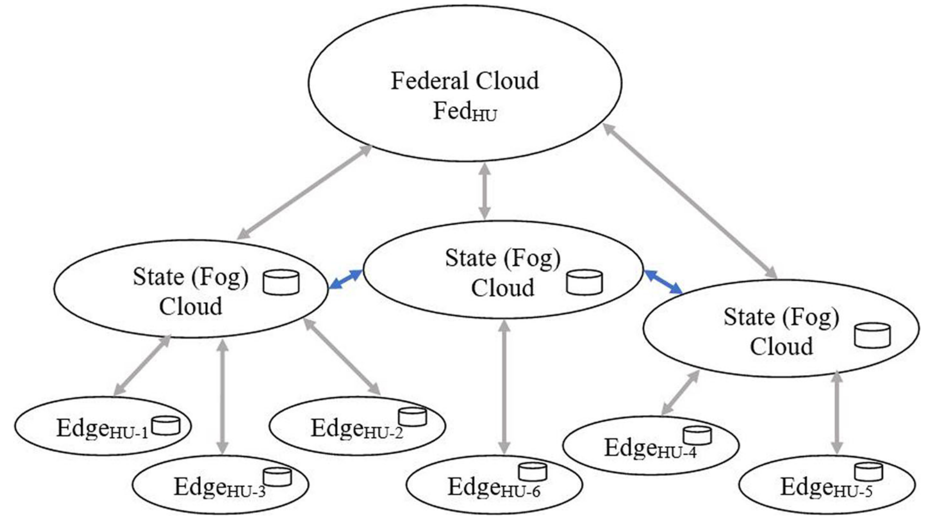
Federal Cloud
Fed_{HU}
State (Fog) Cloud
State (Fog) Cloud
State (Fog) Cloud
$\text{Edge}_{\text{HU-1}}$
$\text{Edge}_{\text{HU-2}}$
$\text{Edge}_{\text{HU-3}}$
$\text{Edge}_{\text{HU-4}}$
$\text{Edge}_{\text{HU-5}}$
$\text{Edge}_{\text{HU-6}}$

Fig. 4 BFC^2 block vault locations (Malomo et al. 2018)

have been vaulted, the detailed file *BFC-BV-Tag-Txn-No* will go through key stretching of encryption, first it is encrypted with Generator's special key $\text{Gen}_{\text{SpKey}}$:

$\text{BFC-GEN} = \text{Encrypt}_{\text{FUNC}}(\text{Gen}_{\text{SpKey}}, \text{BFC-BV-Tag-Txn-No})$

after passed through a special channel for the client's CIM to encrypt with CIM's special key $\text{CIM}_{\text{Spkey}}$:

$\text{BFC-GEN-CIM} = \text{Encrypt}_{\text{FUNC}}(\text{CIM}_{\text{SpKey}}, \text{BFC-GEN})$

This key stretching of encryption is very important and is another dual control; after CIM's encryption, for the Generator to have access to the content of the detailed file would need file to be decrypted, using the CIM special key $\text{CIM}_{\text{SpKey}}$ and vice versa, the CIM would need the Generator special $\text{Gen}_{\text{SpKey}}$ key to read the content. This mark the end of the transactions, and only the block Generator will do the next crucial task of adding the cipher-text of the detailed file to the blockchain ledger. This is the most important part and there is need to digress from the core blockchain concept of shared and distributed ledgers, but asserts the federated blockchain concept. The cipher-text of the detailed file *BFC-BV-Tag-Txn-No* cannot be shared and distributed to all participants because each block contains among many sensitive information, the digital assets location. However, the summary transaction becomes a blockchain record, shared and distributed to all participants for control access accounting/auditing usage. But for BFC-BV-Tag-Txn-No, the security risk is too high for all participants to share, which is the reason for the key stretching and dual control that involves CIM. Therefore, it is important that the level of security and privacy be raised to its highest level and have one restricted platform where participants share this platform. The Generator maintains a federated blockchain central ledger, completely separate from the cloud center for vaults, where all the detailed transaction files are chained. When files are required to be retrieve from vault, first CIMs decrypt for Generator to have access:

$\text{BFC-GEN} = \text{Decrypt}_{\text{FUNC}}(\text{CIM}_{\text{SpKey}}, \text{BFC-GEN-CIM})$

then, Generator runs:

$\text{BFC-BV-Tag-Txn-No} = \text{Decrypt}_{\text{FUNC}}(\text{Gen}_{\text{SpKey}}, \text{BFC-GEN})$

to have access to the detailed information and prepare digital assets for client to retrieve assets.

BFC^2 block vault access control model

There are many access control models, techniques and technologies, administration, and methods that have been serving as the first lines of defense in the war on unauthorized access to computing devices, sensitive data, systems and technology infrastructure resources (Cai et al. 2018; Garcia et al. 2015; Malomo et al. 2018). In recent times, cyber-attacks are strategic and tactical, adversaries are advanced with capability to access sensitive business digital assets, actions which have exposed devastating flaws in many access controls at different levels, some that were at one-time very effective, their weaknesses are now known to adversaries, potentially their purpose as first line of defense are now security risk to systems, data storages and resources (Page et al. 2017). Our approach introduced continuous monitoring and accountability, which are extremely important to access control: authentication and authorization of subjects' interactions with systems and resources. Most secure safes and vaults are impenetrable because every action is controlled and evaluated based on cost, monitoring and accountability.

Our proposed solution breaks down client's request to interact with the BFC^2 vault into sequence of operations with each operation carrying a cost (amortized cost), an operational amount charge to client on every operation that make-up the request (Thomas 2009). We are using the amortized analysis concept to guarantee no data and cyber breaches occur. Also, to ensure that proper processes are in place to detect breaches early should there be any and deploy cyber defense accordingly (Malomo et al.; Sayeed and Marco-Gisbert 2019). To achieve these and in accordance with some flavor of blockchain basis and accounting principles, we employed the accounting method of amortized analysis (Thomas 2009). It is very important that the amortized cost and actual cost assigned for each sequence of operations are carefully and qualitatively defined and agreed upon by the participants in the federation:

Amortized cost for an operation (Amort$_{\text{Cost}}$)≥Actual cost for that operation (Act$_{\text{Cost}}$) Implies that the i$^{\text{th}}$ operation for all sequences of k operations to complete a client's request is:

$$\sum_{i=1}^{k}(Amort_{cost-i}) \geq \sum_{i=1}^{k}(Act_{cost-i}) \tag{1}$$

The Amortized cost for an operation ($Amort_{Cost}$), which may differ from operation to operation, is the ceiling and applies to the entire federation. The Actual cost for that operation (Act_{Cost}) to be allocated to client is determined by each client's CIM and cannot exceed the Amortized cost ($Amort_{Cost}$). The Actual cost for operation is setup for client by CIM during on-boarding client to the network. It can differ from client to client based on the average time required to complete a sequence of operations. There is $Credit_{Score}$ in the smart contract if is set to -2 implies actual cost of operation for that client is 2 less the amortization cost. Table 1 shows the actual cost allocation for digital assets cloud upload operations for three clients to buttress this point: You will observe that based on the credit score $Credit_{Score}$, the actual cost Act_{Cost} for an operation differs for the three. Also, observe that transfer completion notification is same. Otherwise, computation would result to a violation of having negative actual cost, which cannot be zero or negative. Therefore, in

Table 1 Actual cost allocation based on client's credit score

Operation Description	Federation Amortization $Amort_{Cost}$	Client – ABC $Credit_{Score}$ = -2 Act_{Cost}	Client – MNO $Credit_{Score}$ = 0 Act_{Cost}	Client – TUV $Credit_{Score}$ = -1 Act_{Cost}
Split file: 1^{st} Digital Asset to vault	5	3	5	4
Split file: 2^{nd} Digital Asset to vault	5	3	5	4
Split file: 3^{rd} Digital Asset to vault	5	3	5	4
Transfer completion notification	1	1	1	1

such case, actual cost automatically equals set amortization cost ($Act_{Cost} = Amort_{Cost}$). The importance of setting actual cost, support information security principle of least authority. Also, it helps CIM to learn over time, how to service their clients better without exposing the federation to abuser of cloud services.

It's worthy to note that our approach has put into consideration that some technologies, and security mechanisms that are significantly resourceful in the cybersecurity space today, may become weak and vulnerable in the near future. For example, Quantum computing may not take decade from now, before its application potentially becomes a major threat to break cryptographic keys; compromising other security mechanisms that protects sensitive data. Therefore, what makes our BFC^2 vault novel are: how we explored and integrated blockchain and federated cloud computing technologies - to create a federated intelligence community of service providers; to provision unique services that secure cyber infrastructure and vault storage for offsite digital assets; the controls that grant access to only authorized users, and our access control model can perfectly be deployed in system applications development. Also, the ability to split offsite data into different block sizes and store each in different cloud centers within the federation, and blockchain providing additional platform for consistency, trust, monitoring and accountability.

We shall now explain the block vault access control identification, authentication, authorization, and accountability processes in phases. Highlight important concepts employed to protect against man-made internal and external threats in the course of our discussion. We shall break our discussion into external operation - client side access control and internal operation administrator side access control.

Client side authentication, authorization and accountability (Dual access control)

The phases in this section evaluate our access control framework (AAA) and shows proof of concept that our design and approach can securely and intelligently control access (Authentication and Authorization) to systems and resources; can keep track of subject's interaction to generate information necessary for accounting/auditing usage, adaptable to enforce standard best practice security policies, verifiable and flexible to defining access rights authorization level after subject is successfully cleared by our reliable authentication procedure. Figure 5 shows the sequence of events.

- ***Phase 1 - Computing Device Fingerprint:*** The first computation that is required for access, is the hash of the metrics that can uniquely profile a computing device. The purpose for this, is to create an identification fingerprint (IF) that is unique (similar to biometric: *who you are*) for every computing device forensic that will access the block vault to ensure that the authorized system is granted access. Our metrics are the machine hardware components which are: Physical machine address MAC_{Addr}, IDE

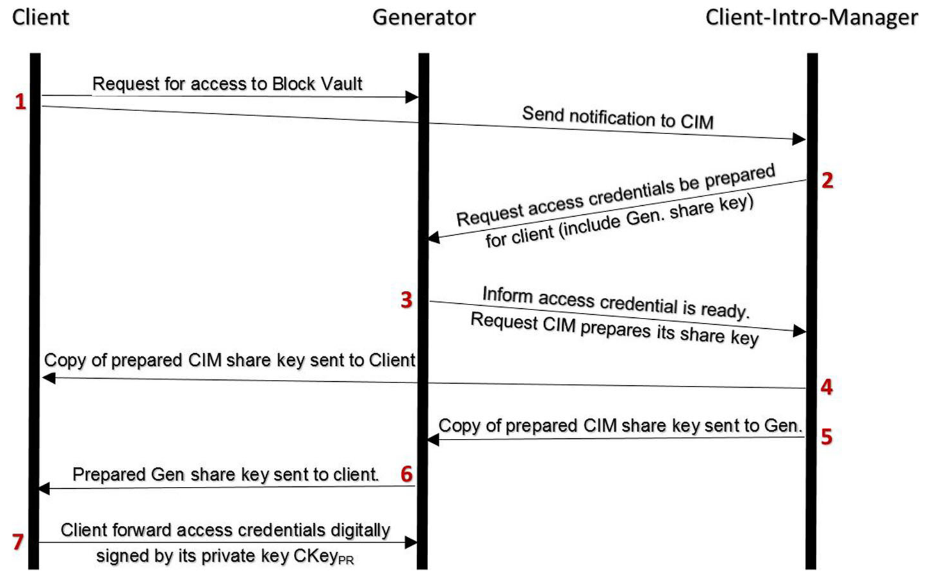
Client
Generator
Client-Intro-Manager
1
Request for access to Block Vault
Send notification to CIM
2
Request access credentials be prepared for client (include Gen. share key)
3
Inform access credential is ready. Request CIM prepares its share key
Copy of prepared CIM share key sent to Client
4
Copy of prepared CIM share key sent to Gen.
5
Prepared Gen share key sent to client.
6
7
Client forward access credentials digitally signed by its private key CKey_PR

Fig. 5 Client access to vault (dual control) – sequence of events

integrated controller model IDE_{Mod}, CPU model and speed CPU_{MS}, installed physical memory size RAM_{Size}, and System drive model and size HDD_{MSz}. To reduce the possibility of a collision, the unique smart contract license transaction Identification $LICENSE_{SC}$ is concatenated, together with all the metrics and passed as parameter to a hash function $Hash_{FUNC()}$ that generates device identification fingerprint $Device_{IF}$.
$Device_{IF} =$
$Hash_{FUNC}(MAC_{Addr} + IDE_{Mod} + CPU_{MS} + RAM_{Size} + HDD_{MSz} + LICENSE_{SC})$

- ***Phase 2 - Client Place a Request:*** The second step is for the client to request for access to storage, by placing two requests on special channels to the Generator and CIM, with each request digitally signed. The content of the request message is simply device identification $Device_{IF}$ combined with client's federated access identification C_{L-ID} (*what you know*), and digital assets split count DA_{CNT} (3 in our previous example), both concatenated and passed as parameter with client private key $CKey_{PR}$ (*what you have*) to a signature function $Sign_{FUNC}$ which uses the $CKey_{PR}$ to encrypt the concatenated parameter and produces the client's digitally signed request $CREQ_{SIGN}$ which is sent to block Generator and CIM.
$CREQ_{SIGN} = Sign_{FUNC}(CKey_{PR}, Device_{IF} + C_{L-ID} + DA_{CNT})$
- ***Phase 3 - Generator/Client-Intro-Manager Response:*** On receipt of the request, both will verify the request using the identity verification function $Verify_{FUNC}$ to verify and identify that the entity is valid. The $Verify_{FUNC}$ decrypt $CREQ_{SIGN}$ using the client's public key $Ckey_{PUB}$. $Verify_{FUNC}$ returns a logical value of TRUE or FALSE to determine the next action required $Action_{REQ}$:
$Action_{REQ} = Verify_{FUNC}(Ckey_{PUB}, CREQ_{SIGN})$
  - **If $\mathbf{Action_{REQ}}$ is FALSE** on either or both sides, which could be as a result of client, might have changed any of the hardware components used as metrics to profile the device, the federated Auditor/Regulator is notified for further investigation. The CIM is involved in the investigation as well. However, if

there is a breach, by an intruder that has clone client computing device or an attempt by hacker, the entire federation is put on an alert, sensitive assets become moving target and all mechanisms and resources are deployed to defend against such threat (Malomo et al.).

- **If Action$_{\mathbf{REQ}}$ is TRUE**, that is result matches CIM's/Generator's computation and expectation, then entity is identified. An estimated location (*where you are*) of the device (x_1, y_1) and cloud center (x_2, y_2) is calculated using the Euclidean distance estimated by Global Positioning System (GPS):

$$Device_{DLOC} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{2}$$

There may be slight difference to whatever Generator/CIM has at their ends from the latest information gathered, any difference in value that is reasonably close will be accepted. The CIM might further investigate the client authenticity. After, there has to be handshake between the CIM and Generator. On behalf of client, CIM request the Generator to prepare client access credential (include symmetric encryption key: share key between Generator and client). Generator will notify CIM when credential is ready and request for CIM share key with client (symmetric encryption key) $CIMKey_{Share}$ to be generated for the client access. CIM forward the key to client and a copy to the Generator. Also, the Generator sends its own share key $GenKey_{Share}$ to client. It is important to note that for multiple signatures same steps and rigor are applied to other signatories. Furthermore, all exchange is done by cryptographic key pairs (public and private).

It is important to note that part of what the CIM request for, on behalf of the client is that a credit line is made available for the client by the Generator. To expatiate on this, we will now explain the summary transaction. Please let's revisit our previous example and once again assume the digital assets: xyz.c.001, xyz.c.002, and xyz.c.003 are prepared (compressed, split, and encrypted) and ready for offsite data upload. Also we shall use one of our clients from Table 1, Client ABC, with credit score: $Credit_{Score}$ = -2. The expected summary transaction is shown in Table 2.

The Generator opens credit line available for the client in the amount of 10 checks (this is a fictional currency, checks like tokens or chips, which we use for monitoring and accounting for vault interactions by client.). Thus, in terms of accounting ledger transactions: the request from CIM to Generator in accounting representation in the federated system is: Generator's Account is

Table 2 Summary transaction for digital asset to/from BFC^2 vault

	CIM's Account		Client's Account		Generator's Account	
Operation Description	Debit	Credit	Debit	Credit	Debit	Credit
CIM request credit line for client		10			10	
Credit available to client	10			10		
Split file: 1st Digital Asset to vault			3			3
Split file: 2nd Digital Asset to vault			3			3
Split file: 3rd Digital Asset to vault			3			3
Transfer completion notification			1			1

debited 10 checks, and the CIM account is credited with 10 checks. Implies that the general transaction ledger in accounting term is balance (row 1 in table 2). However, the amount has to be moved into client's account, therefore the CIM is again debited 10 checks and client's account is credited with exact amount (Note row 2, that at CIM that the account is balance and more importantly these are transactions showing dual control and become part of transaction summary/history that will become chain record). Assuming client is authenticated for now, which will not actually happen until phase 4 but just to quickly touch on authorization.

Authorization of operation depends on, if there is available balance to service operation, bearing in mind that in any-case the balance in client's account will always be drawn down either uploading to, or retrieving from vault digital assets. However, for each sequence of operation performed, client is charged 3 checks and appears as a debit in client's account and credit in Generator's account. At the completion of the transaction to mark the end of session, the balance 1 check is passed as credit to Generator's account, and debit to client's account. At this point the accounting transaction ledger is completely balance at zero amount. CIM is notified in two ways, zero balance on client's credit and Generator request for CIM to encrypt the BFC-BV-Tag-Txn-No for Generator to add to the federated blockchain record. Only the summary transaction blockchain record is shared and distributed to all participants, for control access accounting/auditing usage such as to track subject's amount of data uploaded and session time interacting with the systems and resources. Information like these will guide the federation to enforce informed policy and best practice. Also, note that clients do not have access to the manipulation of actual cost. The balance is on a dashboard for cloud center operators, CIMs and Generator, and for early detection of potential data breaches monitored at the federated cloud center. The dashboard information provides to every node/cloud center, complete visibility of the federation full environment by having a blockchain single view of active clients' operation costs, balances, file transfer status, session times and the amortized costs. These are indicators to identify potential breaches. This way everyone is involved in the federation access control and security monitoring. After the sequence of operations are performed or session exceeds time frame, the credit balance is wipe out to zero. Procedure for authentication would have to be initiated again to resume operation.

- ***Phase 4 - Client Access with Credential:*** This is the last step and shows that more than one entity is involved in the access authentication process by reflecting digital signing with dual control for client access.

  Dual Control: After the client has received from the Generator $GenKey_{Share}$ and from the CIM $CIMKey_{Share}$, the client will use them to perform a key stretching of encryption on its client identification and digitally sign using its private key: The key stretching means that signature function $Sign_{FUNC}$ will be called three times with different parameters each time. First, client Generator share key $GenKey_{Share}$ and client's unique federated access Identification C_{L-ID} are passed as parameters,

$GenKey_{Share}$ is used to encrypt C_{L-ID}, the second time Client-Info-Manager share key $CIMKey_{Share}$ will be used to encrypt the previous outcome and finally, client's private key $CKey_{PR}$ encrypt the last result as the digital signature. These are important to the block vault security because these keys have a window time frame they have to be used. Having the keys is a response to *"who knows you"*. There is a secure channel for getting the share keys from the Generator and client-intro-manager CIM.

$CREQ_{SIGN} =$
$Sign_{FUNC}(CKey_{PR}, Sign_{FUNC}(CIMKey_{Share}, Sign_{FUNC}(GenKey_{Share}, C_{L-ID})))$

Client Token Verification: Client present this token for final authentication and access to vault storage through a special channel to the Generator. The Generator through this channel authenticates using $Verify_{FUNC}$.

$Action_{REQ} =$
$Verify_{FUNC}(GenKey_{Share}, (Verify_{FUNC}(CIMKey_{Share}, Verify_{FUNC}(Ckey_{PUB}, CREQ_{SIGN})))$

Please note: We breakdown the client token verification to confirm the order of decryption and encryption for easy comprehension as follows:

First decryption by calling $Verify_{FUNC}$ with client's public key $Ckey_{PUB}$ and $CREQ_{SIGN}$, passed as parameters. Let us call the result XX and matches Generator's expectation:

XX = $Verify_{FUNC}(Ckey_{PUB}, CREQ_{SIGN})$

Secondly $Verify_{FUNC}$ is called to decrypt the result using the copy of Client-Info-Manager share key $CIMKey_{Share}$. Let us call the result YY

YY = $Verify_{FUNC}(CIMKey_{Share}$, XX)

Finally, $Verify_{FUNC}$ is called for the last time to decrypt the last result using the block Generator shared key $GenKey_{Share}$

$Action_{REQ}$ = $Verify_{FUNC}(GenKey_{Share}$, YY)

If Action$_{\mathbf{REQ}}$ is TRUE, authentication is established and access to vault storage is granted, otherwise, the breach process is activated.

Administrator side authentication steps (Split knowledge control)

The concern here is the internal threat. Internal risk can be intentional and unintentional. The unintentional are sometimes as a result of genuine mistakes which could be lack of training. In either case, assets must be protected and access must be to authorize user only. Although, it may seems daunting to protect against an internal personnel with malicious intent, which is one of the reasons the assets are split. For complete digital assets to be stolen and reconstructed to original file, would required at the minimum three internal personnels from different edge cloud centers to collaborate in the crime, need cryptographic keys to decrypt back to split compressed files and finally, need to know the client's encryption password key used to encrypt the split compressed files before the upload procedure. Furthermore, the blockchain record containing detailed transaction of digital assets locations is locked down by client's CIM, and the Generator would be required to collaborate in the crime too. Separation of duties is one of the effective control measures put in place to curb misuse of information, sabotage and some other security compromises. The split knowledge allows more than one person to be involved at any time in handling the administration of the vault. No single individual in the system can perform all tasks because key combinations to access are divided among three validators which are chosen in random. The important security to note here is that the client is represented by the CIM which is the eye for the clients during this process. Figure 6 depicts Split Knowledge to access any vault by an administrator.

- **Administrator** places a request for access, details what the task will entail and estimate time duration for the process. The Generator will respond to the request and generate access credentials broken into four parts for the three validators and the administrator.
- **Generator** will forward different public keys ($GenKey_{PUB}$) to the selected validators and the administrator.
- **Each individual** generate a hash value derived from the public key received and sent digitally signed by their private key (e.g. Administrator - $AdmKey_{PR}$, Validator-1 - V_1Key_{PR} etc.)
  $Adm_{sign} = Sign_{FUNC}(AdmKey_{PR}, Hash_{FUNC}\left[(GenKey_{ADM})\right)$
  $V_{1sign} = Sign_{FUNC}(V_1Key_{PR}, Hash_{FUNC}\right](GenKey_{PUBv1}))$
  $V_{2sign} = Sign_{FUNC}(V_2Key_{PR}, Hash_{FUNC}\left[(GenKey_{PUBv2})\right)$
  $V_{3sign} = Sign_{FUNC}(V_3Key_{PR}, Hash_{FUNC}\right](GenKey_{PUBv3}))$
- **Generator** receives the digitally signed data and will run checks to validate each, and combine the access in order to form a key combination that grant access to the administrator.
  $$\begin{aligned} Access_{REQ} = {} & Verify_{FUNC}(AdmKey_{PUB}, Adm_{sign})+ \\ & Verify_{FUNC}(V_1Key_{PUB}, V_{1sign})+ \\ & Verify_{FUNC}(V_2Key_{PUB}, V_{2sign})+ \\ & Verify_{FUNC}(V_3Key_{PUB}, V_{3sign}) \end{aligned}$$

Performance evaluation and discussion

Performance of our proposed framework is evaluated in a simulation as shown in Fig. 7, using amortized cost, actual cost of an operation and credit line balance available to client.

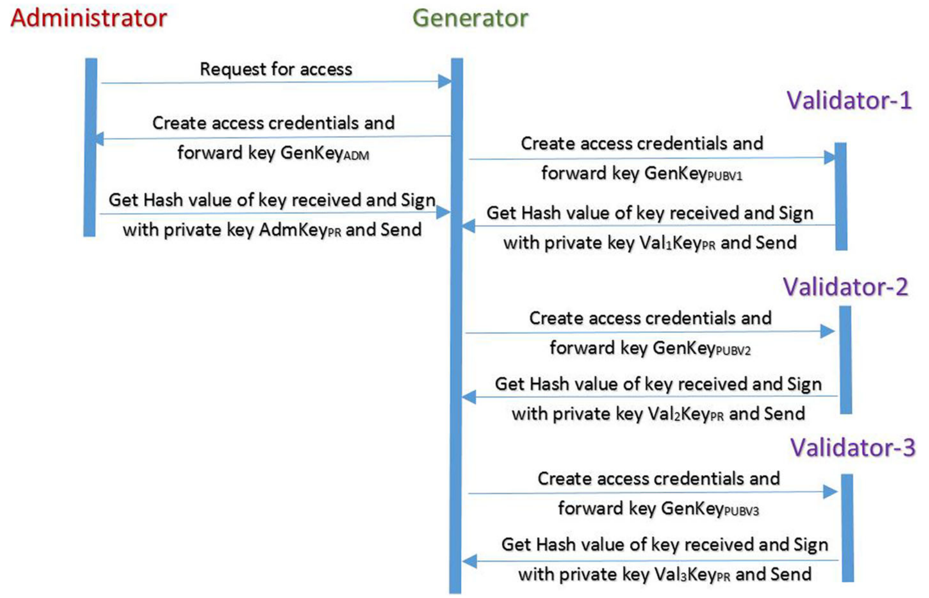
Administrator
Generator
Request for access
Create access credentials and forward key GenKey$_{ADM}$
Get Hash value of key received and Sign with private key AdmKey$_{PR}$ and Send
Validator-1
Create access credentials and forward key GenKey$_{PUBV1}$
Get Hash value of key received and Sign with private key Val$_1$Key$_{PR}$ and Send
Validator-2
Create access credentials and forward key GenKey$_{PUBV2}$
Get Hash value of key received and Sign with private key Val$_2$Key$_{PR}$ and Send
Validator-3
Create access credentials and forward key GenKey$_{PUBV3}$
Get Hash value of key received and Sign with private key Val$_3$Key$_{PR}$ and Send

Fig. 6 Vault access sequence of events by split knowledge

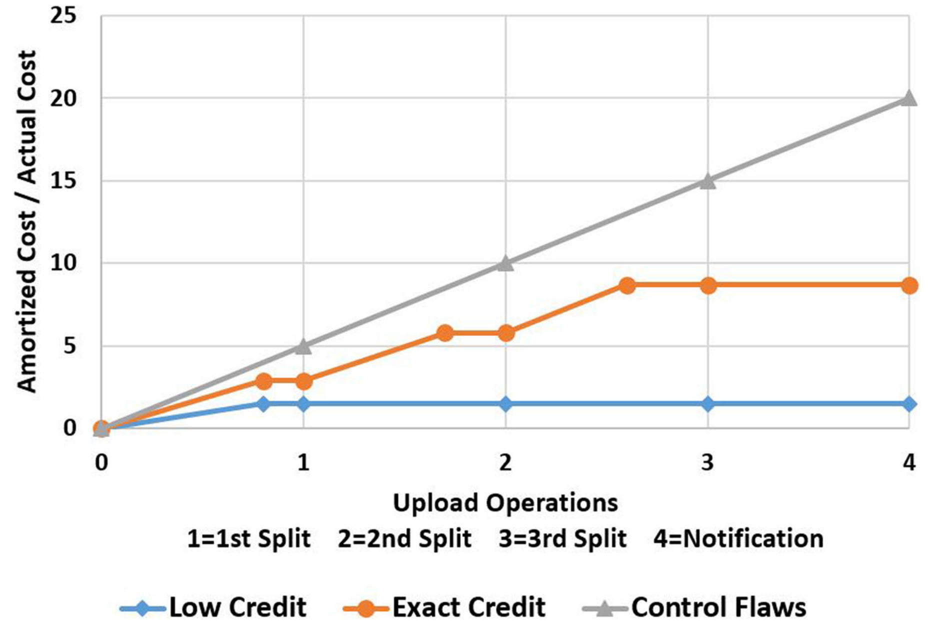

Fig. 7 Traditional approach vs proposed approach and breach detection

The blue line showed evaluation of client with only minimal credit (2 checks allocated by CIM), enough to upload first split file and afterwards there was no enough credit balance to proceed to operation 2, and as a result transaction was forced to terminate even though there was some tiny session time left out of the 2 checks. The orange line showed normal upload operations (e.g. client ABC in our previous example), with exact credit amount to perform all three upload digital assets transfer operations to block vault. At the end of each operation the actual cost of the operation (in this case 3 checks) is deducted from client's credit balance even though there are some saved session time. The rule is after end of the entire sequence of operations, the balance must be zero. The grey line buttressed the reason for our proposed framework, showed activities with control flaws where a subject can remain interacting with systems and resources, which open systems and resources to abuse and other security risks.

It may interest you to note that the blue (low credit) line that terminated is a breach. It is an incomplete transaction with one operation and would indicate on the monitoring and accountability dashboard. This would be of interest to the federation's auditor and those concern.

It is worthy to note that there would be a follow-up paper, on how we employed the complex concept of moving target defense, for cybersecurity vault storage and monitory data breaches. We would show additional layers of security for roaming computing devices; the ways their digital assets are securely stored, and how authorized access to vault works.

Conclusion

In this paper, we have demonstrated how to secure storage for offsite digital assets, with a great degree of efficiency, privacy, scalability and restricted access control that address, and improve early detection of data breaches, by continuously evaluating subject's access control and interaction with resources, using operation cost for monitoring and accountability. We presented a framework built to manage risk, revoke access control easily and protect against man-made internal and external threats, by implementing cybersecurity critical controls such as multi-authentication factor, system identification fingerprint, separation of duties, and split knowledge. Evaluation showed proof of concept that our design and approach can securely and intelligently control access to systems and resources.

Acknowledgments

Not applicable.

Authors' contributions

All authors have contributed equally to the paper. All authors read and approved the final manuscript.

Funding

This work was supported in part by the U.S. National Science Foundation (NSF) under grants CNS-1658972 and CNS1650831. However, any opinion, finding, and conclusions or recommendations expressed in this material are entirely those of the author and do not necessarily reflect the views of NSF.

Availability of data and materials

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Received: 30 August 2019 Accepted: 10 February 2020
Published online: 27 February 2020

References

Baliga A (2017) Understanding blockchain consensus models. Persistent 2017(4):1–14
Cai F, Zhu N, He J, Mu P, Li W, Yu Y (2018) Survey of access control models and technologies for cloud computing. Clust Comput:1–12. https://doi.org/10.1007/s10586-018-1850-7
Conteh NY, Royer MD (2016) The rise in cybercrime and the dynamics of exploiting the human vulnerability factor. Int J Comput (IJC) 20(1):1–12
Course ORMIA Specialized (2016) Social engineering
Crosby M, Pattanayak P, Verma S, Kalyanaraman V, et al. (2016) Blockchain technology: Beyond bitcoin. Appl Innov 2(6–10):71
Ehrenfeld JM (2017) Wannacry, cybersecurity and health information technology: A time to act. J Med Syst 41(7):104
Ettredge M, Guo F, Li Y (2018) Trade secrets and cyber security breaches. J Account Pub Policy 37(6):564–585
Garcia DJP, Ouye MM, Rossmann A, Crocker ST, Gilbertson E, Huang W, Humpich S, Vainstein K, Ryan NM (2015) Methods and systems for providing access control to secured data. Google Patents. US Patent 9,129,120
Jarvis J (2017) Ransomware: Are you paying attention? https://www.fortinet.com/blog/industrytrends/ransomware-are-you-payingattention.html. Accessed 14 May 2018
Kshetri N, Voas J (2017) Do crypto-currencies fuel ransomware? IT Prof 19(5):11–15
Malomo OO, Rawat DB, Garuba M (2018) Next-generation cybersecurity through a blockchain-enabled federated cloud framework. J Supercomput 74(10):5099–5126
Malomo O, Rawat DB, Garuba M A federated cloud computing framework for adaptive cyber defense and distributed computing. In: 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). pp 1–6. https://doi.org/10.1109/infcomw.2017.8376184
Malomo OO, Rawat DB, Garuba M (2018) A Survey on Recent Advances in Cloud Computing Security. The Journal of Next Generation Information Technology 9(1):32–48
Page J, Kaur M, Waters E (2017) Directors' liability survey: Cyber attacks and data loss—a growing concern. J Data Prot Priv 1(2):173–182
Rawat DB, Malomo O, Bajracharya C, Song M (2017) Evaluating physical-layer security for secondary users in cognitive radio systems with attackers. In: MILCOM 2017-2017 IEEE Military Communications Conference (MILCOM). pp 659–665. https://doi.org/10.1109/milcom.2017.8170855
Sabillon R, Cano J, Cavaller RV, Serra Ruiz J (2016) Cybercrime and cybercriminals: a comprehensive study. Int J Comput Netw Commun Secur 4(6):165–176
Saridakis G, Benson V, Ezingeard J-N, Tennakoon H (2016) Individual information security, user behaviour and cyber victimisation: An empirical study of social networking users. Technol Forecast Soc Chang 102:320–330
Sayeed S, Marco-Gisbert H (2019) Assessing Blockchain Consensus and Security Mechanisms against the 51% Attack. Appl Sci Multidiscip Digit Publ Inst 9(9):1788
Swanson T (2015) Consensus-as-a-service: a brief report on the emergence of permissioned, distributed ledger systems
Tabone SR (2017) Cyber Security: 51 Handy Things to Know about Cyber Attacks. RJT Lion Publishing
Thomas H (2009) Amortized Analysis. In: Cormen CE, Leiserson RL, Rivest CS (eds). Introduction to Algorithms. The MIT Press, Cambridge, Massachusetts London. pp 451–456
Underwood S (2016) Blockchain beyond bitcoin. Commun ACM 59(11):15–17
Yu PK (2015) Trade Secret Hacking, Online Data Breaches, and China's Cyberthreats. Cardozo L. Rev. De-Novo:130

Publisher's Note