Permission and Usage Control for Virtual Tourism using Blockchain-based Smart Contracts

Muhammad Shoaib Siddiqui, Toqeer Ali Syed, Adnan Nadeem, Waqas Nawaz, Ahmad Alkhodre
Faculty of Computer and Information Systems
Islamic University of Madinah, Madinah
Kingdom of Saudi Arabia

***Abstract*—Virtual Tourism (VT) is a booming business with potential perspectives in the entertainment and financial industry. Due to travel restrictions, safety concerns, and expensive travelling the younger generation is showing interest in virtual tourism instead of traditional tourism. However, virtual tourism does not financially benefit the service providers as compared to traditional tourism stakeholders. An online system is essential to provide a central point of access to various tourism sites along with usage, permission, and payment control. In this paper, a secure blockchain-based broker service for users and content providers is proposed, which allows tourism sites to announce their virtual tours and provide accessibility and accountability. Meanwhile, it enables users to register, subscribe, access, and be billed according to their usage. The permission control module ensures authentication and authorization, while the usage control provides accountability to the predefined service level agreement. The transactions are stored on the blockchain to ensure the integrity of data and smart contracts are used to ensure automatic usage and permission control. An implementation on Hyperledger Fabric is provided as a proof of concept with performance measurements as a case study.**

***Keywords*—Virtual tourism; permission control; usage control; access control; blockchain**

I. Introduction

The imaginative ability of the human mind can be perceived and experienced by the latest Virtual Reality Technologies [1]. People can visualize and interact with virtual objects in a realistic or enhanced environment. They can build and experience their imaginative environment, which allows them to live, roam, and interact in an ideal, purpose-built world. In general, it is not easy to define the limits of the human imagination, but with the cutting-edge technology of computer-based systems [2], it is possible to create human-inspired imaginary worlds. With the development of technology, the imagination of the virtual world can be diversified from the real world to provide a more realistic experience.

Industry 4.0's arsenal accompanies one of the most promising technologies, Virtual Reality, which is one of the modern technologies in the current generation that yielded a rising market for VR application development. The extensive advancements in VR and other mixed realities have considerably changed experiences in different areas of interest. The implications and uses of these technologies are broad and are being actively used to boost the level of experiences of clients. The massive adoption of these technologies is found in healthcare departments [3] [4], education sectors [5] [6], manufacturing and logistics [7], smart cities [8], Museum [9] [10], social media [11], etc. In retail, virtual reality enables products to be visualized in 3D for consumers [13] [12] that ultimately leads to customer satisfaction and maximum sales.

One of the breakthrough applications of VR in recent times is Virtual Tourism [16], which enables a tourist to explore nature, attractions, destinations, ruins, buildings, and other travel destinations without physical interactions. The immersive experience is provided by the emerging technologies of VR and remote sensing. Virtual tours may include 360-degree view, guided tour of the virtual and augmented world, and video tours of remote destinations, which provides the ability to interact with and experience the art, nature, culture, or ancient ruins [14]. Worldwide, the VT industry was worth USD five Billion in 2021 and it is anticipated to reach USD 24 Billion in 2022 [15], as shown in Fig. 1.

VT provides a cheap replicate, motivation, and a marketing pretense for the tourism industry [16]. Due to the recent coronavirus pandemic (global pandemic), social distancing was recommended that made people's daily lives a bit boring with monotonous experiences. VT enables tourism experiences within the boundaries and guidelines of epidemic restrictions. Users can use specialized equipment from the comfort of their homes to create immersive experiences in virtual or real environments. Another benefit of VT is that it preserves cultural and natural heritage in environments vulnerable to tourist visitation and urbanization [17].

VT is enabled by VR/AR/MR technologies and applications. Without the implementation and support of the right tools, this promising industry cannot develop and thrive [16]. At one end, the virtual and augmented content is created using 3D cameras, scene modelling and composition, and image rendering using computer graphics; while at the extreme end, display technologies with embedded tracking technologies are used to provide an immersive experience to the user. The realistic experience of the user is the selling point of these virtual tours. The availability of tours as online or computer-based services requires accountability for permission, rights, access, and usage to ensure justified financial benefits for the service and content providers. Without this accountability and billing model, the services are not financially feasible.

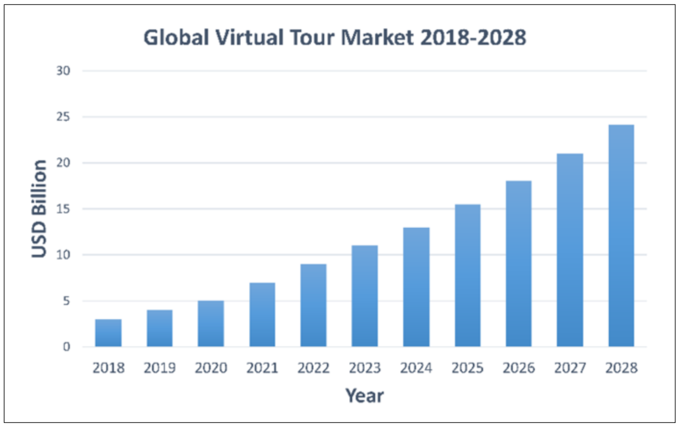

Fig. 1. Market Growth of Virtual Tourism [15].

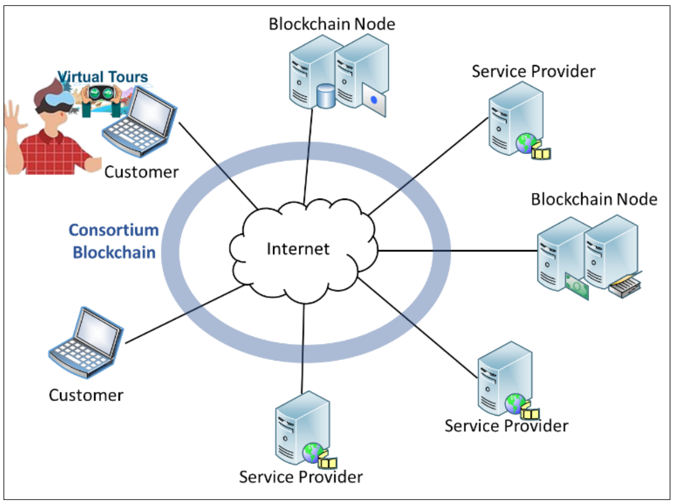

Fig. 2. The Blockchain based Content Distribution System using a Consortium of Blockchain Nodes, and Registered Service Providers and Customers.

To facilitate the virtual tourism industry and its customers, this paper presents an online system that provides a consortium blockchain-based decentralized access to various tourism sites and content providers, as shown in Fig. 2. Along with the permanent nodes of the blockchain, service providers and customers can become members of the blockchain, which maintains usage, permission, and payment control. It allows the tourism sites to announce their virtual tours and provides accessibility and accountability, at the same time, allows users to register, subscribe, access, and be billed according to their usage. The permission control module ensures authentication and authorization, while the usage control provides accountability and billing. The transactions are stored on the blockchain to ensure the integrity of data and smart contracts are used to ensure automatic usage and permission control.

The rest of the paper is structured as follows. Section II discusses the literature review. Section III presents the proposed blockchain-based permission and usage control system. Section IV presents the implementation details on Hyperledger fabric and evaluation of the system and Section V concludes the paper and provides future directions.

II. Related Work

VT is used by various industries from museums to ancient ruins and from universities to stadiums. VT is an efficient marketing edge that any business can rely on for expanding clientage and market penetration. Tourism agencies use virtual tours for marketing their destinations and attractions. According to a survey in 2017 by TIG Global, 50% of tourists base their selection of a destination on virtual tours [18]. A study by Google in 2015, identified a 50% increase in online interest and a 40% increase in booking reservations [19]. According to Panomatics' statistics, 75% of potential buyers consider virtual tours as a major factor in their purchasing decision [20]. Another study found that among the students of culture, 80% of people go to virtual exhibits, museums, and classes for further education [21]. The online event industry, which is a subcategory of virtual tourism, is identified to grow by $2.3 trillion in 2026 [22].

Many websites and portals on the Internet provide virtual tours to various destinations, such as museums [9], architectures [23] [26], [28], underwater explores [24], archaeological sites [25] [27], monuments [29], Holy sites [30] [31], and remote locations [32] [33]. From the couch of your living room, you can party in Ibiza, be alone in the clouds in a hot air balloon, dive with fishes in the Georgia Aquarium, fly over the skies of Paris; and much more [34].

However, most of these sites are using VT for marketing purposes only. As the boom of VT expands, more and more people will prefer virtual tourism over traditional tourism [16]. For making the system financially feasible, a content distribution system is required, which provides all the tours at a single portal, to allow authentication, authorization, accountability, and payment facilities along with security and trust. Such a content distribution system for VT is missing; however, research has already been done for permission, access, and usage control.

In [35], the authors have reviewed various access control systems and identified the challenges and opportunities. They have recommended the use of blockchain for developing an access control system with decentralized trust. Kishigami et al. have proposed a decentralized blockchain-based digital content distribution system along with a prototype application [36]. It uses a digital currency-based payment system; however, a usage control system is missing. Maesa et al. have proposed an access control system, which uses blockchain to publish the policies expressing the right to access a resource and to allow the distributed transfer of such rights among users. The system provides distributed auditing and prevents fraudulent transactions. It is deployed on the Bitcoin blockchain [37]. In [38], the same team has proposed the use of blockchain technologies to implement the traditional access control system to achieve better auditability.

Khan et al. have proposed an extended usage control model known as DistU (Distributed Usage Control), which considers all possible access control models required by a business for permissioned blockchain frameworks [39]. The detailed mechanism of DistU is provided in [40]. Khan et al. also presented an extension to the UCON model in [42]. Their model is an extension of the UCON model developed by Park and Sandhu [41]. UCON is the most generalized usability control system with work based on pre-defined attributes, conditions, and obligations. It uses a set of policies for maintaining usage before, during, and after access is granted to a user.

The main contribution of this work is a comprehensive and purpose-built permission and usage control for virtual tourism cases, in which the customers can register into the system, subscribe to and visit the virtual tours, and pay for services in a trusted environment. At the same time, the service provider can financially benefit from the virtual tours.

The proposed system is also a derivation of the UCON model, in which the model has been modified according to the requirements based on the customer as a consumer of the service, virtual tours as the service and tourist cites as the service providers. Along with the usage control of UCON, the access and permission control systems have been embedded for the completeness of the solution. A working system on top of a consortium blockchain using the Hyperledger Fabric as a proof of concept has also been proposed. The next section provides the detail of the proposed system.

III. Blockchain-based Permission and Usage Control

To enable content streaming for virtual tourism, a comprehensive system has been proposed, which allows users to register, maintain their accounts and subscriptions, and handle bills. For tourism agencies and service providers, it supports content provision, billing itineraries, and service management. It also enables secure and trusted permission, rights, and usage control using blockchain technologies. Fig. 3 shows the modules available in the proposed system and the following sections discuss the details of each module.

A. Membership

The system is developed on a permission blockchain, therefore only the members are allowed to enter the system. A new user must register into the system by creating an account through a valid email address, contact, and financial details, which enable the system to charge or/and remit the user for the services. There are mainly two types of users: customers and service providers. On registering into the system, a user is created in the blockchain's couch DB, which serves as a distributed database, while the certification authority creates security credentials for authentication and maintaining the confidentiality of the data. In this section, the various participants and entities involved in the system, and their interactions are defined.

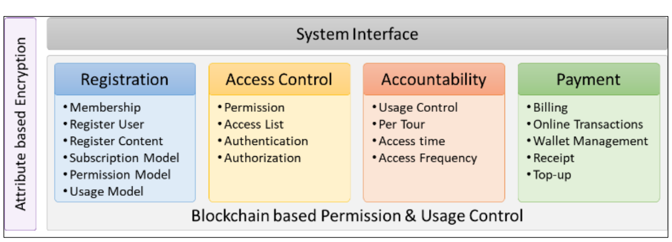

Fig. 3. Modular Diagram of the Proposed Blockchain-based Permission and Usage Control System.

Definition 1: A customer is defined in the system as an active member (ui), such that ui ∈ U, while U specifies the set of customers with attributes: identity (id), credentials (ci), set of subscribed tours (Ti), wallet (wi), and states Σ and A.

$$u_i \in U(id, c, T, w, \Sigma, A) \quad (1)$$

Where, Σ is the state of user (customer or service provider) as signed-in, signed-out, granted, or denied. A is a set of states for the customer against the set of subscribed tours. The state value can be initial, subscribed, requested, accessing, or denied.

Definition 2: A Service Provider (ri) is also a user in the system, (ri), such that ri ∈ R, while R specifies the set of service providers with attributes: identity (id), credentials (ci), set of provided tours (Ti), wallet (wi), and state Σ. A service provider can create/provide a set of Tours (Ti).

$$r_i \in R(id, c, T, w, A) \quad (2)$$

Upon arrival into the system, a user must sign-in to the system with its id and credentials. A membership function (Ψ) checks a user (u_i/r_i) against corresponding credentials (c_i) to allow or deny an entry into the system by updating the state Σ.

$$\Sigma(u_i): \Psi(u_i, c_i) \quad (3)$$

The membership model provides authentication in the system by defining who can access and enter into the system.

B. Content Registration

The content provided by the service provider to the customers is in the form of virtual tours. A customer can view and experience the virtual tour as per the subscription and is charged according to the subscription rate, which is paid through an associated wallet.

Definition 3: A tour is defined as a service consumed by the user (u) and provided by the service provider (r). It is denoted by t such that t ∈ T (id, r, β, ρ, τ, U).

Where, r is the service provider of the tour, β is the subscription fee, ρ is the usage rate per frequency of access, τ is the usage rate per time of access, and U is the set of users subscribed to consume that resource.

A service provider ri can enter a tour tj into the system if the state is signed-in. An online portal allows the service provider to manage her/his tours in terms of editing information, subscription rates, content, narration, etc. A tour has various attributes such as id, location, subscription rates, viewer's rating and reviews, historical information, content, narration, information and details, and list of attractions, etc., out of which the mandatory tuple is defined in definition 3. After a successful signed-in into the system, the service provider can add tours along with their details and media files, which are saved in a distributed database. After a service provider creates a tour, it is assigned an id and associated with that service provider. The function O(r, t) checks if the service provider ri is the owner of the tour ti, based on which the system permits the service provider to edit the tour or deny access.

C. Subscription

The subscription model defines which user and service provider can access and edit/manage which tours respectively. It enables authorization in the system.

After a customer (u_i) has successfully signed into the system, the customer can enlist the tours available in the system. The customer can search the tours based on various categories such as id, location, subscription rate, viewer's rating, etc. From the available list of tours, a customer can subscribe to the tour given that the subscription condition is fulfilled, which checks if there is enough amount in the customer's wallet to pay the subscription fee against the selected tour:

$$\Delta : w(u_i) \geq \beta(t_j) \quad (4)$$

Where u_i is the customer, t_j is the selected tour, $w(u_i)$ is the amount in the wallet of customer u_i, and β is the subscription fee for the tour t_j, and Δ is the subscription status.

- If Δ = allow (1), then tj is inserted into the set of Ti for customer ui.

$$T_i(u_i) \leftarrow t_j, \text{ or } t_j \in T_i(u_i) \quad (5)$$

At the same time, for the tour t_j, u_i will be inserted in its set of subscribed users U_j.

$$U_j(t_j) \leftarrow u_i, \text{ or } u_i \in U_j(T_j) \quad (6)$$

- If Δ = deny (0), then the customer ui is informed that his wallet (wi) does not have the required amount to subscribe to the tour tj.

Θ(ui, tj) is the subscription function that checks for a user's (ui) access to a tour (tj), which can be initial, subscribed, requested, accessing, or denied. The state is maintained in aj, where aj ∈ Ai for ui.

D. Access and Permission Control

Permission control has two stages. The first stage is for the entry into the system for the users (authentication), while the second stage is for permitting users to access only their owned and subscribed contents (authorization). When a user logs-in to the system, the permission control checks corresponding credentials (id and password) to ensure a user is allowed in the system, based on which it permits or denies access. Ψ(ui, ci) is the membership function that checks user ui against user credentials ci to grant or deny entry into the system.

$$\Psi(\alpha i, ci) \rightarrow \Sigma(\alpha i) = \{\text{granted, denied}\}, \text{ where:} \quad (7)$$

$$Role\ (\alpha)\ = \begin{Bmatrix} customer, & \alpha \in U \\ service\ provider, & \alpha \in R \end{Bmatrix} \quad (8)$$

In case of access is denied, the user is advised to register into the system. When a customer registers successfully, a customer entry is created and added to the set of Customers U:

$$U \leftarrow u_i\ (id_i, c_i, T_i, w_i, \Sigma_i, A_i) \quad (9)$$

When a service provider registers successfully, an entry is created and added to the set of Service Providers R:

$$R \leftarrow r_i\ (id_i, c_i, T_i, w_i, A_i) \quad (10)$$

The second stage of permission control is used when a customer tries to view one of the tours. The permission control uses the subscription model to check if a customer is authorized to access the said tour. Similarly, when a service provider is trying to edit/manage the tour, the permission model checks the ownership.

Θ(ui, tj) is the subscription function that checks if a customer ui is subscribed to a tour tj to granted or denied access to the tour tj. The state is maintained in the set Ai. Similarly, Θ(ri, tj) checks if a service provider ri is the owner of a tour tj to granted or denied access to the tour tj. The state is maintained in the set Ai.

E. Usage Control

The usage control maintains accountability in the system by calculating the use of the service by each customer and ensures that the user does not exploit the system by excessing the usage defined for the said content. After the permission control grants access to the customer to view the tour/content, the usage control is activated. Usage control maintains a state of access for each user accessing a tour t_i. However, in the case of a service provider accessing pre-owned tours, the usage control is not activated.

Usage control used two functions to calculate the usage of the content/tour (t_j) for a customer (u_i). $\Xi(u_i.\ t_j)$ calculate the usage in terms of time; as the amount of time the customer u_i has accessed the tour t_j. While $\Pi(\ u_i.\ t_j)$ calculates the usage in terms of frequency, as how many times the tours t_j was accessed by user u_i.

For each tour t_j the service provider adds a rate ρ_j that is used to calculate the usage in terms of access time and rate τ_j, which determines how many times a tour is accessed. This approach is used to calculate the billed amount ($Q(u_i)$) for the customer (u_i), which is debited from the wallet of the customer u_i, and credited into the service provider r_k, who owns the tour t_j. These two usage parameters are designed to give flexibility to the service provider to charge its customer. If the service provider wants to charge the customers based on how many times the user accesses the system only, then a non-zero value is provided to τ_j and a zero value is provided to ρ_j. On the other hand, when the charge is based on the time of access only, then a zero value is provided to τ_j and a non-zero value is provided to ρ_j.

When the customer is allowed to access the tour and allowed by the permission control, the state of the tour for that customer is changed to 'requested.' After that, the usage control first checks the balance in the wallet of the user and if the value of τj is non-zero, then the user is allowed access only if the user has a balance of more than τj. In that case, the state for the tour is changed to accessing. If there is not enough balance, then the state is changed to 'denied' and the user is informed about the lack of balance. The change of state is visualized in Fig. 4. In case the value for τj is zero, then the condition in step 3C1 in Algorithm 1 is true and the customer would only be charged based on λ(ui), which is the amount of time in seconds.

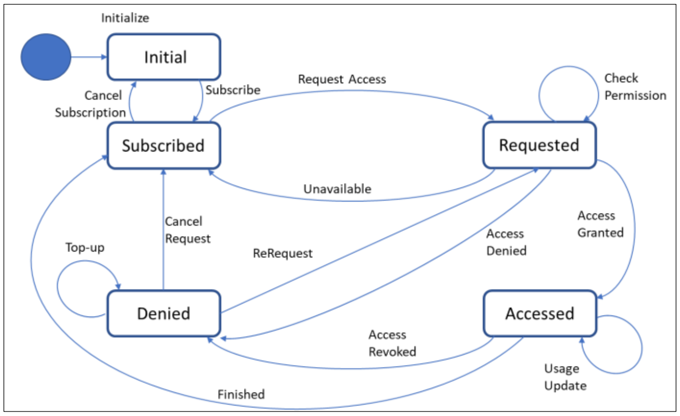

Fig. 4. The State-Model Diagram of the Usage Control Model, which Monitors the usage of a Tour by a Customer and Ensures that the user has enough Balance to Keep viewing the Tour.

Similarly, the usage control monitors and calculates the usage based on the time using equation (11) and deducts the amount from the wallet using equations (12), (13), and (14), respectively.

$$\Xi(u_i.\ t_j) = \rho_j \times \lambda(u_i) \quad (11)$$

$$\Pi(u_i.\ t_j) = \tau_j \times \nu(u_i), \quad (12)$$

$$Q(u_i) = \Xi(u_i.\ t_j) + \Pi(u_i.\ t_j) \quad (13)$$

$$w(r_k) = w(r_k) + Q(u_i) \quad (14)$$

Where, $\nu(u_i)$ is the number of times a tour (t_j) is accessed by the customer (u_i) and $\lambda(u_i)$ is the time (how long) for which the tour (t_j) is accessed by the customer (u_i).

Once the amount is lower than the threshold (which is zero in most cases), then the usage control will revoke the access of tj for the user (ui) and change the state to denied.

$$\Delta: w(u_i) = w(u_i) - \Xi(u_i.\ t_j), \text{ where } \Delta: w(u_i) \geq 0 \quad (15)$$

- If Δ = allow (1), then ui keeps accessing the tour tj and aj = {accessing}
- If Δ = deny (0), then tj access for ui is revoked and aj = {denied}

The customer is informed in the system about the revoked access and given an option to top-up her/his account by crediting the balance to the account through the manage account section of the portal. Afterward, the customer can send a re-request for the content/tour, which change the state to 'requested.' The usage control checks the condition (given in equation 15), if the result is 'allow', then the state is changed back to accessing and the tour is continued for the user. At any time, when the customer selects to cancel the access to the tour, the system stops the access and changes the state back to 'subscribed.' This whole process is outlined in Algorithm 1 and the flowchart in Fig. 5 for better understanding.

Algorithm 1: Customer accessing a Tour; Permission and Usage Control

1. System enlists the available tours based on id and location,
2. Customer u_i selects a tour t_j to view
3. Permission Control checks subscription
   if $u_i.a(t_j)$ = ***subscribed***
   a. *Permission = Allowed*
   b. state(t_j) = ***requested***
   c. **if** $w(u_i) \geq \nu(u_i)$ //(deduct usage based on number of access)
      1. $w(u_i) = w(u_i) - \tau_j * \nu(u_i)$,
      else
      i. **inform** user "*Not enough balance to view the tour*"
      ii. state(t_j) = ***subscribed***
      iii. **goto** step 1
      end if
   else (**if** not ***subscribed***)
      subscribe tour t_j
   a. Check for the subscription fee
      if $w(u_i) \geq \beta_j$
      i. $u_i.a(t_j)$ = ***subscribed***
      ii. $w(u_i) = w(u_i) - \beta_j$ //(deduct subscription fee)
      iii. **goto** step 3
      else
      i. **deny** request (*Permission = Not Allowed*)
      ii. **end**
      end if
   end if
4. Grant Access to the Tour, state(t_j) = ***accessing***
5. **while** state(t_j) = **accessing**
   a. **if** *user.action = Cancel Tour*
      i. state(t_j) = ***subscribed***
      ii. **end** //Stop Access to the tour
      end if
   b. **monitor** Usage Control, $\lambda(u_i)$ = time in seconds for how long the customer u_i is accessing the tour t_j
   c. **calculate** time usage: $\Xi(u_i.\ t_j) = \rho_j \times \lambda(u_i)$, // Debit the wallet and check for available balance
   d. $w(u_i) = w(u_i) - \Xi_{i,j}$
   e. **if** $w(u_i) < 0$
      i. **revoke** access
      ii. state(t_j) = ***denied***
      iii. **inform** user "*Not enough balance to view the tour*"
      iv. **if** *user.action == Cancel Tour*
         a. state(t_j) = ***subscribed***
         b. **goto** step 1
      v. **else if** *user.action = ReRequest*
         a. state = ***requested***
         b. goto step 3.c
      vi. **else if** *user.action = Topup Account* AND $w(u_i) \geq \Xi_{i,j}$
         a. Action = *ReRequest*
         b. goto step 5.e.iv
         else
         a. **goto** step 5.e.iii
         end if
      end if
6. **end while**

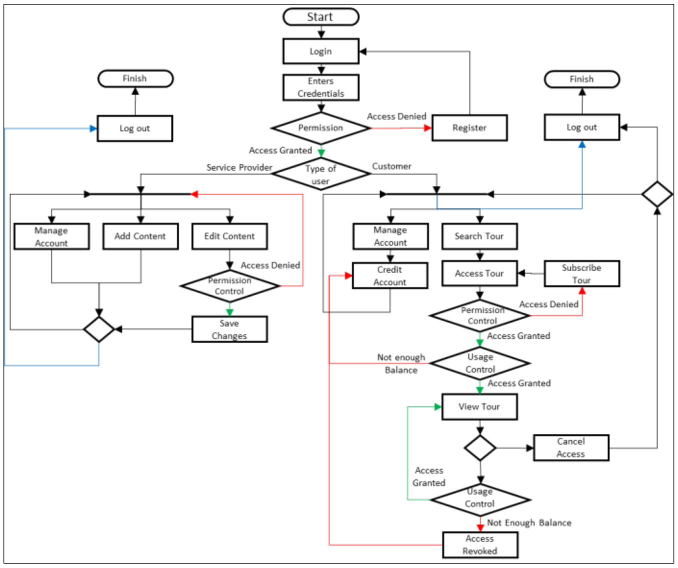

Fig. 5. The Flowchart Depiction of Algorithm 1, Showing the Permission and usage Control.

IV. Implementation and Evaluation

The proposed usage and permission control system is based on the concept of consortium blockchain, where some permanent nodes are part of the blockchain, and other nodes can register for membership and become part of the blockchain. As shown in Fig. 2, for making the tours available for the customer without needing a centralized server, the service providers can register as a member and then register their content on the blockchain. Similarly, customers can become members and can search for and access the tours. Fig. 6 shows the registration activity for the service provider on the blockchain, verified by various nodes.

Due to its inherent properties, the blockchain provides distributed trust among customers and service providers. Content distribution and financial transactions can be implemented securely on the blockchain. Using smart contracts, the usage and permission control can implement the policies and makes sure that authentication, authorization, and accountability are maintained.

Fig. 7 shows the working of the permission and usage control system on top of the blockchain network A user can register on the blockchain as a member, then search and subscribe for various tours by paying the subscription fee. Afterward, the user can view the tours. The permission control maintains a permission state against a user accessing a tour. It checks the customer's permission to access the said tour and allows or denies the customer based on permission rights.

Similarly, during the access, the usage control maintains the usage state. It monitors the frequency and time of view for the customer against a tour. The system calculates the bill and makes sure that the customer has enough balance in the associated wallet. Accordingly, the usage control can revoke (or grant) access to the customer with a notification.

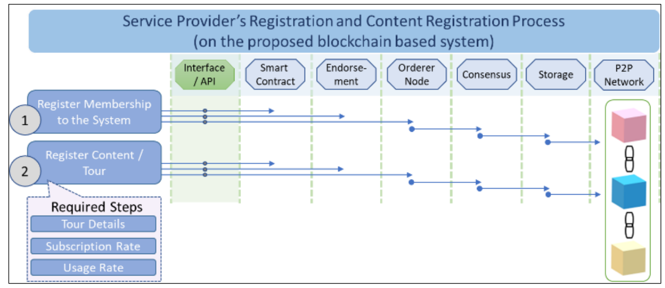

Fig. 6. Member and Content Registration Steps for the Service Provider on the Consortium Blockchain.

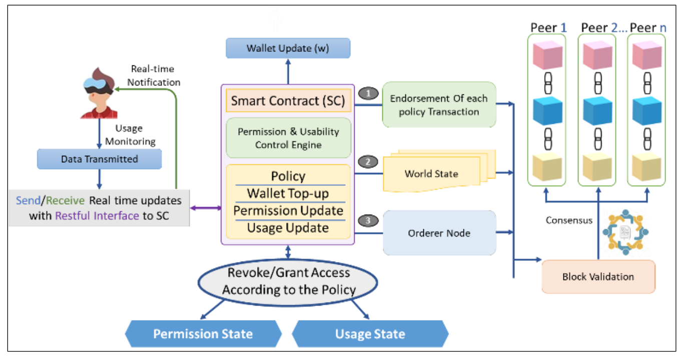

Fig. 7. Permission and Usage Control Model on Top of the Consortium Blockchain.

F. *Implementation of Blockchain*

The following text details the configuration of the testbed created to verify the permission and usage control model presented in the previous section. The proposed system is implemented on Hyperledger Fabric 1.4 designed as a consortium blockchain, with two permanent nodes and three service provider nodes, and two customer nodes. In the end, the performance evaluation of the consortium blockchain is presented in terms of the throughput and the latency of the system. However, first, a small introduction to blockchain technologies is given, along with the motivation for using the Hyperledger Fabric.

1) Blockchain technologies: The earlier blockchains, such as bitcoin, etc., face challenges with energy consumption and speed due to their public nature. On the other hand, modern blockchains have been developed as private or consortium concepts, to help overcome these limitations and provide practical value for other business uses and applications. The top modern blockchain frameworks are R3 Corda, Hyperledger, and Ethereum. Some of the modern blockchain frameworks are discussed below:

2) IBM Blockchain: IBM Blockchain is a private, decentralized blockchain network that has been the most successful with enterprise clients as it links into enterprise cloud and legacy technologies more seamlessly than is possible in other decentralized networks. The developer tool was designed to be flexible, functional, and customizable with a user-friendly interface to simplify critical tasks, such as setting up, testing, and rapidly deploying smart contracts [43].

3) R3 Corda: R3 Corda [44] is a distributed ledger platform that uses a novel consensus mechanism in which transactions are cryptographically linked but does not periodically batch multiple transactions into a block. It processes all transactions in real-time, which makes it superior in performance. However, R3 declared it as a blockchain and not a blockchain at the same time, as some large companies have observed implementation constraints and is deemed not suitable for consortium blockchains.

4) Ethereum: Ethereum [48] is one of the oldest and most established blockchain platforms. It provides a truly decentralized blockchain that is comparable to the Bitcoin blockchain network. It enables true decentralization with support for smart contracts. Its key weaknesses include slow processing times and higher transaction processing costs compared to other platforms.

5) HydraChain: Hydrachain blockchain platform [45] is an extension to the Ethereum platform that supports the development of a permissioned distributed ledger. Therefore, the tools for developing smart contracts and decentralized apps are same as the Ethereum; however, the issues of Ethereum are also present.

6) Quorum: Quorum [46] is a customized version of Ethereum developed by the financial services company JPMorgan. It has been optimized to support high-speed transactions between institutions such as banks and insurance companies on a private blockchain. It also adds various privacy enhancements to Ethereum to improve support for regulations. It supports confidentiality and privacy for all transactions using smart contracts. Quorum has lower throughput at higher load and experiment results suggest an increase in transaction latency with regard to reading and writing operations.

7) IOTA: IOTA [47] is a blockchain platform designed specifically for IoT blockchain applications, which uses a directed acyclic graph to store transactions on its ledger, motivated by potentially higher scalability over blockchain-based distributed ledgers. IOTA achieves consensus through a coordinator node, operated by the IOTA Foundation. It is not a strict blockchain solution due to the coordinator serving as a central control feature, which goes against the idea of blockchain as “distributed ledger technology”.

8) Hyperledger Fabric [49]: Hyperledger Fabric provides a collection of tools to create permissioned blockchain applications. It is engineered with distributed ledger as the core concept. It supports components that can be plugged into a modular architecture, which works well in private and permissioned blockchain deployments, which improves security and speed. An open smart contract model that can support various data models is also provided.

After the comparative analysis of these modern blockchain frameworks, it is concluded that Hyperledger Fabric, due to its prominent features, was inspired to be selected as the number one candidate for the implementation of the proposed system. Now, a brief overview of the various components of the Hyperledger is presented in the following subsection.

G. Setup of Hyperledger Components

As a proof of concept, the proposed system is implemented as a consortium blockchain using Hyperledger fabric with the essential components, such as certificate authority, working as a membership service provider, orderer node, anchor node, committer nodes, endorser nodes, etc., carrying out the transactions, creating blocks of transactions, implementing the consensus algorithm, and saving them on the shared ledger, as shown in Fig. 8. The discussion about these components, their roles, and their implementation on Hyperledger fabric are discussed in the following sub-section.

1) Certificate authority: The job of the Certification Authority (CS) is to provide secure communication between the blockchain nodes, authenticate the users, and ensure the integrity and confidentiality of the data. It effectively works as the Membership Service Provider (MSP) that provides access to the peer nodes (blockchain nodes, customers, and service providers) in the system, creates digital certificates along with pairs of private and public keys for encryption and decryption of the messages, ensure trust, and enable registration for the new users. Each peer has its own identity given by a membership services provider (MSP), which authenticates each peer to its channel peers and services.

2) Channel: Channels are used to provide communication between the peer nodes of the blockchain consortium to conduct private and confidential transactions. A channel constitutes of. Each transaction on the network is executed on a channel, which constitutes authenticated and authorized members, anchor nodes, the shared ledger, smart contracts or chaincode, and orderer nodes.

3) Smart Contract: Smart Contracts, also called chaincode in Hyperledger fabric, are responsible for processing transaction requests and determining whether transactions are valid by executing policies. These could be third-party applications connected to the blockchain network, or chaincode can be written in Go, JavaScript (node.js), and eventually other programming languages, such as Java through Restful interfaces. The chaincode implements the logic of the usage and permission control and validates the login defined through the policies.

4) Validation of system chaincode: Validated System Chain-Code (VSCC) instigates the source of transaction in the blockchain, validates each transaction signed by peer nodes, and ensures the endorsement of the transaction is performed by enough member nodes.

5) Anchor peer: The anchor peers are responsible for providing communication between the members of various organizations, such as service providers, permanent nodes of the consortium blockchain, and customer nodes. It uses the gossip protocol that identifies other available member peers, disseminates ledger data across peers on a channel, and brings new peers up to date quickly.

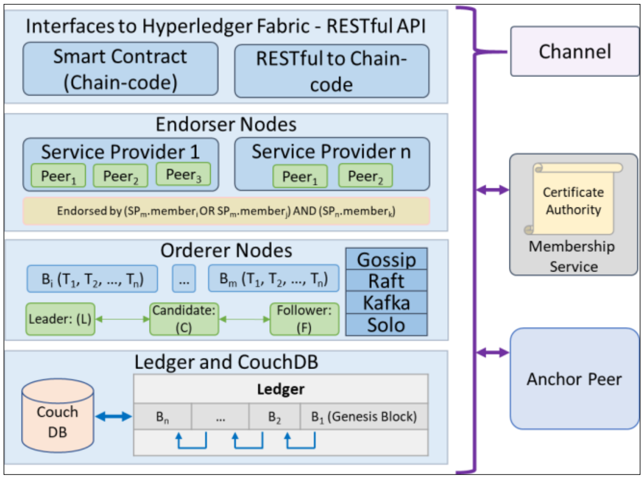

Fig. 8. The Modular Diagram of the Deployed Hyperledger with Various Internal Components.

6) Endorsement policy: The endorsement policy is implemented to ensure that before adding details of transactions into the ledger, the details are first signed by predetermined peers. Every chaincode has an endorsement policy that specifies the set of peers on the channel that must execute the chaincode and vouch for the outcome of the execution for a transaction to be considered valid. To launch different transactions from the smart contract, the multiple interface interaction is ensured through RestAPI.

7) Consensus: There are three main steps of consensus in Hyperledger Fabric. First is the endorsement, which is driven by the policy upon which the members endorse a transaction. Second is ordering, in which the endorsed transactions are arranged in an order to be committed to the ledger. Third is validation which takes a block of ordered transactions and validates the correctness of the results, including checking endorsement policy. A number of consensus mechanisms were opted for and tested among RAFT, Kafka, and Solo; however, in terms of efficiency, Raft has proven to be much better.

8) Byzantine Fault Tolerance (BFT): BFT is a simple consensus mechanism in which every node has to vote. Each node represents a blockchain ledger (Kafka divides these ledgers into clusters on each node to make searching easy). They need to reach a consensus on the current state of the network. This means a good number of the participating nodes have to reach a decision and execute an identical action to avoid failures. The algorithm is Crash Free, which means even if some nodes fail to vote, the algorithm will not crash.

The endorser node in the blockchain consortium endorses every single transaction performed related to the aforementioned activities i.e. the new members joining the system, and transactions related to funds deposited in favor of the service provider. Once the endorsement is performed, the next operation is carried out which is termed ordering. During the ordering process, all endorsed transactions are executed to form a block. That is passed through a validate node that performs validation of the created blocks. Once the blocks are validated, they are approved via a consensus mechanism. A number of consensus mechanisms exist; however, in this implementation scenario, the default consensus mechanism of Hyperledger Fabric, Raft is used. Raft has a network of nodes that function to update the blockchain ledger. Raft was selected because it is fault-tolerant, in case of a node failure, and efficient as compared to Kafka and Solo. In addition, Raft conducts an election among the nodes for electing a leader in the blockchain network.

Once Raft opts for the leader, the endorser sends the block to the leader which actually ensures data state synchronization among members of the blockchain consortium. Ultimately, the ledger state is updated, synchronized, and trusted.

H. Performance Analysis Results

This paper, in particular, incorporates features with regard to usage control, access control, and permission control for customers associated with the Tours. In order to measure and evaluate the prototype implementation with regard to the latency and throughput of the proposed system, a testbed environment was created that only processes transactions related to usage control and permission control.

The testbed is implemented on two virtual processors of Intel Xeon server as the permanent nodes of the consortium blockchain, along with eight-gigabyte memory. The two nodes are connected through a gigabit ethernet connection. IBM chaincode's fabric-load-gen tool is used with various test cases with varying latencies and endorsement policies. The generated transactions are configured as both read and write by using load-generating scripts. The latency is calculated as the delay in completing a whole transaction, including endorsement time, broadcasting time and commit and validation time. The delay introduced by the orderer nodes is also included, while the number of transactions executed per second is used to calculate the throughput of the system.

Fig. 9 and 10 depict Hyperledger performance over different block sizes, the latencies (Fig. 9) and throughput (Fig. 10) of the transactions for different transaction arrival rates, and the effect of various endorsement policies with various transaction arrival rates. The linear increment in the throughput can be seen in Fig. 10, according to the increase in the rate of incoming transactions. It can be observed that after 150 tps, this linear increase is limited due to the saturation point. The saturation point is presented in Fig. 8. This saturation is mainly due to the three computationally intensive operations of policy validation, which are (1) x.509 certificate identity validation, which involves reconstruction of the certificate from the received serialized byte stream; (2) validation of the identity by the membership service provider and, (3) verification of the digital signature to authenticate the transaction. The size of the block gets bigger as the number of endorsements is increased with each transaction, which includes encoding and decoding of x.509 certificates.

The testbed is configured using Hyperledger Fabric 1.4 which employs Raft consensus. Raft produces more acceptable performance results as compared to other versions of the Hyperledger Fabric. These results support the idea of blockchain-based permission and usage control mechanisms for decentralized content distribution in virtual tourism.

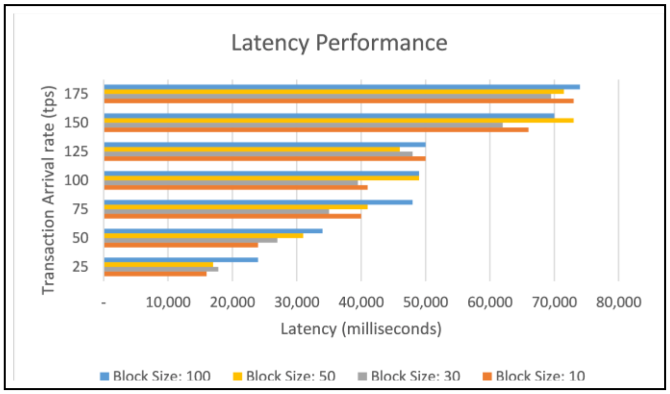

Fig. 9. Latency Performance of the Deployed Hyperledger with Various-Sized Blocks.

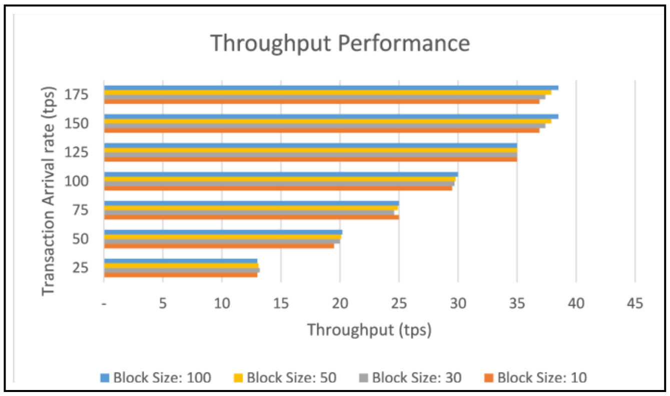

Fig. 10. Throughput Performance of the Deployed Hyperledger with Various Sized Blocks.

V. Conclusion

To facilitate the virtual tourism industry and its customers, this paper presents an online system that provides a consortium blockchain-based decentralized access to various tourism sites and content providers. Along with the permanent nodes of the blockchain, service providers and customers can become members of the blockchain, which maintains usage, permission, and payment control. It allows the tourism sites to announce their virtual tours and provides accessibility and accountability, at the same time, allows users to register, subscribe, access, and be billed according to their usage. The permission control module ensures authentication and authorization, while the usage control provides accountability and billing. The transactions are stored on the blockchain to ensure the integrity of the data and smart contracts are used to ensure automatic usage and permission control. An implementation on Hyperledger Fabric is provided as a proof of concept with performance measurements as the evaluation.

In the future, permission and rights controls can be enabled for the shared objects in a collaborated virtual reality, where various users can create their virtual objects, enter historical facts, create milestones, and frequently asked questions (FAQs) in the virtual tours. Through this, the users can control and permit who can access their virtual assets. Similarly, reviews and experiences could be added to a virtual tour with the control in the hands of the service provider.

Acknowledgment

The work presented in this paper is funded by the Deputyship for Research and Innovation, Ministry of Education, Kingdom of Saudi Arabia, under project No (20/16), titled: “Virtual journey into the history of the Kingdom of Saudi Arabia”. We would also like to extend special thanks to the Deanship of Research, Islamic University of Madinah for their support.

References

[1] Farshid, Mana, Jeannette Paschen, Theresa Eriksson, and Jan Kietzmann. "Go boldly!: Explore augmented reality (AR), virtual reality (VR), and mixed reality (MR) for business." Business Horizons 61, no. 5 (2018): 657-663.

[2] Alcantud, Francisco, Gerardo Herrera, Gabriel Labajo, I. Dolz, C. Gayá, V. Avila, A. Blanquer, Jose Luis Cuesta, and J. Arnáiz. "Assessing virtual reality as a tool for support imagination." In International Conference on Computers for Handicapped Persons, pp. 143-144. Springer, Berlin, Heidelberg, 2002.

[3] Pillai, Anitha S., and Prabha Susy Mathew. "Impact of virtual reality in healthcare: a review." Virtual and augmented reality in mental health treatment (2019): 17-31.

[4] Lányi, Cecília Sik. "Virtual reality in healthcare." In Intelligent paradigms for assistive and preventive healthcare, pp. 87-116. Springer, Berlin, Heidelberg, 2006.

[5] Freina, Laura, and Michela Ott. "A literature review on immersive virtual reality in education: state of the art and perspectives." In The international scientific conference elearning and software for education, vol. 1, no. 133, pp. 10-1007. 2015.

[6] Wu, Hsin-Kai, Silvia Wen-Yu Lee, Hsin-Yi Chang, and Jyh-Chong Liang. "Current status, opportunities and challenges of augmented reality in education." Computers & education 62 (2013): 41-49.

[7] Egger, J., Masood, T.: Augmented reality in support of intelligent manufacturing–a systematic literature review. Computers & Industrial Engineering 140, 106,195 (2020).

[8] Jamei, Elmira, Michael Mortimer, Mehdi Seyedmahmoudian, Ben Horan, and Alex Stojcevski. "Investigating the role of virtual reality in planning for sustainable smart cities." Sustainability 9, no. 11 (2017): 2006.

[9] The 360-degree virtual museum tour of The Louvre, https://360stories.com/paris/place/louvre-museum.

[10] These 12 Famous Museums Offer Virtual Tours You Can Take on Your Couch, https://www.travelandleisure.com/attractions/museums-galleries/museums-with-virtual-tours.

[11] de los Ríos, S., Cabrera-Umpiérrez, M.F., Arredondo, M.T., Páramo, M., Baranski, B., Meis, J., Gerhard, M., Prados, B., Pérez, L., del Mar Villafranca, M.: Using augmented reality and social media in mobile applications to engage people on cultural sites. In: International Conference on Universal Access in Human-Computer Interaction, pp. 662–672. Springer (2014).

[12] Poushneh, A.: Augmented reality in retail: A trade-off between user’s control of access to personal information and augmentation quality. Journal of Retailing and Consumer Services 41, 169–176 (2018).

[13] Poushneh, A., Vasquez-Parraga, A.Z.: Discernible impact of augmented reality on retail customer’s experience, satisfaction and willingness to buy. Journal of Retailing and Consumer Services 34, 229–234 (2017).

[14] Verma, Sanjeev, Lekha Warrier, Brajesh Bolia, and Shraddha Mehta. "Past, present, and future of virtual tourism-a literature review." International Journal of Information Management Data Insights 2, no. 2 (2022): 100085.

[15] Global Virtual Tourism Market Research Report, Industry Analysis & Forecast (2022-2027). https://www.marketdataforecast.com/market-reports/virtual-tourism-market.

[16] Muhammad, S. S., Toqeer, A. S., Nadeem, A., Nawaz, W., & Alkhodre, A. (2022). Virtual tourism and digital heritage: An analysis of VR/AR technologies and applications. International Journal of Advanced

Computer Science and Applications, 13(7) doi:https://doi.org/10.14569/IJACSA.2022.0130739.

[17] Parry, Ross. Recoding the museum: Digital heritage and the technologies of change. Routledge, 2007.

[18] TIG Global, https://www.tourdeforce360.com/advantage_hotel.html.

[19] 360 photos, https://mailchi.mp/8996c0149181/360-stats-sheet.

[20] Panomatics, http://panomatics.net/web/statistics/.

[21] Markletic, https://www.markletic.com/blog/virtual-event-statistics/#virtual-event-preparation.

[22] Trend Exchange, Virtual Events Industry Expected to Grow from $78 Billion to $774 Billion by 2030, https://www.prnewswire.com/news-releases/virtual-events-industry-expected-to-grow-from-78-billion-to-774-billion-by-2030-301095683.html.

[23] Dubai 360 virtual tours, https://www.360virtualtour.co/dubai-360-virtual-tours/.

[24] An interactive journey to the Great Barrier Reef, https://attenboroughsreef.com/.

[25] The Great Wall of China, Google Arts and Culture, https://artsandculture.google.com/story/igVxCi6iJJ6CrA.

[26] Virtual tours: Buckingham Palace, https://www.royal.uk/virtual-tours-buckingham-palace.

[27] Yosemite Virtual Tour, https://www.virtualyosemite.org/.

[28] View Colosseum in 360 virtual tour, https://www.touristtube.com/Things-to-do-in-Rome/Colosseum-360.

[29] Statue of Liberty :: 360° VR Panorama, http://www.samrohn.com/360-panorama/statue-of-liberty-new-york/.

[30] The Holy Land Places in 360-Degree Virtual Reality Tour, https://www.p4panorama.com/gallery-item/the-holy-land/.

[31] Experience Mecca in VR, https://www.oculus.com/experiences/gear-vr/1125286047502859/.

[32] Brave the Himalayas in 'Everest VR: Journey to the Top of the World, https://www.oculus.com/blog/brave-the-himalayas-in-everest-vr-journey-to-the-top-of-the-world/.

[33] 360° VR Space Safari, https://orsted.com/en/explore/space-safari.

[34] The 35 Best Virtual Tours Online So You Can Travel From Home, https://worldwidehoneymoon.com/best-virtual-tours-online-travel-from-home/.

[35] Rouhani, Sara, and Ralph Deters. "Blockchain based access control systems: State of the art and challenges." In IEEE/WIC/ACM International Conference on Web Intelligence, pp. 423-428. 2019.

[36] Kishigami, Junichi, Shigeru Fujimura, Hiroki Watanabe, Atsushi Nakadaira, and Akihiko Akutsu. "The blockchain-based digital content distribution system." In 2015 IEEE fifth international conference on big data and cloud computing, pp. 187-190. IEEE, 2015.

[37] Di Francesco Maesa, Damiano, Paolo Mori, and Laura Ricci. "Blockchain based access control." In IFIP international conference on distributed applications and interoperable systems, pp. 206-220. Springer, Cham, 2017.

[38] Maesa, Damiano Di Francesco, Paolo Mori, and Laura Ricci. "A blockchain based approach for the definition of auditable access control systems." Computers & Security 84 (2019): 93-119.

[39] Khan, Muhammad Yasar, Megat F. Zuhairi, Toqeer Ali, Turki Alghamdi, and Jose Antonio Marmolejo-Saucedo. "An extended access control model for permissioned blockchain frameworks." Wireless Networks 26, no. 7 (2020): 4943-4954.

[40] Marmolejo-Saucedo, José-Antonio. "An extended access control model for permissioned blockchain frameworks." OPENAIRE (2019).

[41] Park, J., & Sandhu, R. (2004). The UCON ABC usage control model. ACM Transactions on Information and System Security, (TISSEC), 7(1), 128–174.

[42] Khan, Yasar, Toqeer Ali, Megat Fariz, Fernando Moreira, Frederico Branco, José Martins, and Ramiro Gonçalves. "BlockU: Extended usage control in and for Blockchain." Expert Systems 37, no. 3 (2020): e12507.

[43] IBM, "IBM Blockchain Platform known issues," https://cloud.ibm.com/docs/services/blockchain?topic=blockchainknown-issues, 2020, [Online; accessed 30-Oct-2022].

[44] B. R.G, "Introducing r3 corda: A distributed ledger for financial services," https://gendal.me/2016/04/05/introducing-r3-cordaa-distributed-ledger-designed-for-financial-services/, 2016, [Online accessed 30-Oct-2022].

[45] T. Kumar, "What is Hydrachain technology and how it works," https://www.blockchain-council.org/blockchain/what-ishyderachain-technology-how-it-works/, 2020, [Online; accessed 30-Oct-2022].

[46] G. Scholar, "Performance evaluation of the Quorum Blochchain platform," https://arxiv.org/abs/1809.03421, 2020, [Online; accessed 30-Oct-2022].

[47] V. Skwarek, "Blockchains as security-enabler for industrial iotapplications," Asia Pacific Journal of Innovation and Entrepreneurship, vol. 11, no. 3, pp. 301–311, 2017.

[48] Wood, Gavin. "Ethereum: A secure decentralised generalised transaction ledger." Ethereum project yellow paper 151, no. 2014 (2014): 1-32.

[49] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich et al., "Hyperledger fabric a distributed operating system for permissioned blockchains," in Proceedings of the Thirteenth EuroSys Conference, 2018, pp. 1–15.