Cheng J., Tao X. and Das M. (2020). "An encryption key distribution strategy for secure sharing of sensitive information using blockchain in construction projects" In: *Proc. 37th CIB W78 Information Technology for Construction Conference* (CIB W78), São Paulo, Brazil, pp. 363-373. DOI: http://dx.doi.org/10.46421/2706-6568.37.2020.paper026

AN ENCRYPTION KEY DISTRIBUTION STRATEGY FOR SECURE SHARING OF SENSITIVE INFORMATION USING BLOCKCHAIN IN CONSTRUCTION PROJECTS

Jack C. P. Cheng[1], Xingyu Tao[2] and Moumita Das[3]*

Abstract: Project information is shared in construction projects via centralized web-based or cloud-based platforms. However, pertaining to the distributed nature of construction projects where project participants are bound by contractual relationships for the duration of a project, a lack of complete trust among them is likely. Therefore, centralized platforms that require entrusting ownership and management of information to a single entity are unsuitable for construction projects. Therefore, public blockchain platforms that can facilitate irreversibility in records through their distributed ledger technology is recommended.

However, information on public blockchain ledgers is public which is unsuitable for sharing sensitive project information such as payment and tendering related information. Therefore, this paper proposes an encryption key distribution strategy for construction projects using which project participants can authenticate their identities and share sensitive in-formation between two contracting parties in a confidential manner. Considering the high degree of sub-contracting in construction projects and the non-technical nature of construction project participants, the proposed key distribution strategy is designed to create minimum key management overhead for the project participants. The security of the proposed key distribution strategy is validated with a symbolic attack model using a security protocol verification tool called Tamarin prover and supporting discussions.

Keywords: Blockchain, Cryptography, Construction Data Sharing, Key Distribution.

1 INTRODUCTION

In construction projects, project information is shared using centralized cloud-based or web-based platforms such as Aconex (ACONEX 2018), BIM 360 (Autodesk 2018), and PMWeb (PMWeb 2020). On such centralized platforms, the ownership of data for management is required to be entrusted with project participants or a trusted third party. Data on cloud is stored on virtual machines that share resources via a common hypervisor (Studnia et al. 2012). Therefore, risks such as data loss, data corruption, and denial of data access are some of the risks posed by centralized cloud-based platforms (Beckham 2011). Construction projects, in general, have a fragmented project-organizational structure where project participants are bound by contractual relationships for a short duration of time. Due to this reason, they do not fully trust each

[1] Associate Professor, The Hong Kong University of Science and Technology, Hong Kong, cejcheng@ust.hk
[2] PhD. Student, The Hong Kong University of Science and Technology, Hong Kong, xtaoab@connect.ust.hk
[3] Post Doctorate Fellow, The Hong Kong University of Science and Technology, Hong Kong, moumitadas@ust.hk

other. Therefore, a trustless method of sharing information where project participants do not have to trust each other or a centralized entity for safekeeping of information is required in construction projects. Blockchain is a peer-to-peer technology that facilitates irreversibility in records stored on them through distributed ledger technology and probabilistic consensus algorithms (Crypto51 2020, Zhang and Lee 2019). However, due to the public nature of the blockchain ledgers, they are not suitable for recording sensitive information such as payment, design changes, and tendering related information in a plain text format. Therefore, data may be encrypted with military-grade symmetric encryption algorithms such as AES (Advanced Encryption Standard) (Dobbertin et al. 2004) before storing on blockchain platforms. However, construction projects are an aggregation of contractual relationships where encryption should be done in such a way that sensitive information common between two contracting parties are accessible only by them while non-sensitive information is available publicly to other project participants for monitoring and auditing purposes. For example, sensitive information in payment claims such as personal financial information and amount paid should be kept confidential between contracting parties only. However, the fact a payment transaction has taken place between the two parties and non-sensitive information such as payment date and status should be public to facilitate transparency in payments.

In this paper, a key distribution strategy is proposed for construction projects by deploying public-key encryption (Dolev 1983) based method to share encryption keys among project participants for facilitating data confidentiality between two contracting parties and user authentication. Distribution of encryption keys in hierarchical networks such as wireless sensor networks has been explored by researchers. Researchers (Indu et al. 2016, Ali et al. 2017) have proposed key distribution methods using a trusted third party and centralized cryptographic servers between users, data owners, and cloud storage in wireless sensor networks. However, due to the fact that cloud-based key management is faced with problems of high latency (Kahvazadeh and Garcia 2018) and the requirement of entrusting central authorities with parameters of encryption, they are not appropriate for construction projects. Chen et al. (Chen et al. 2014) proposed a distributed hierarchical key distribution strategy to create common encryption keys for messaging between two sensors via a sensor cluster head. This approach requires transferring a common key to the communicating sensors via a network that may be corrupted. Diffie Hellman (DH) key exchange (Rescorla 1999) protocol facilitates the establishment of common encryption keys between two communicating parties without having to actually transfer the common encryption key over a network. However, the DH key exchange protocol approaches that do not deploy methods to authenticate the identities of honest users suffers from the vulnerability of Man-in-the-Middle (MIM) attacks (Conti et al. 2016). In MIM attacks, an attacker poses as an honest user to establish a secure communication channel between himself and another honest user and trick them into leaking sensitive information.

Therefore, the proposed key distribution strategy in this paper uses a blockchain network as a platform to authenticate the identities of honest parties interested in a confidential communication. This key distribution strategy does not require entrusting a central entity for distributing encryption keys. In the proposed approach, every project participant holds one public-private key-pair that is used to authenticate their identity as honest users and generate shared encryption keys for as many as contractual relationships necessary. Considering the high degree of sub-contracting in construction projects, the key distribution strategy is designed to create minimum key management

overhead by facilitating the on-the-fly generation of shared encryption keys by users using public parameters from the blockchain platform and their own encryption key-pair. The proposed key distribution strategy is validated using a symbolic attack model that simulates attack scenarios by an adversary. A security protocol verifier tool called Tamarin prover (Basin et al. 2017a) is used to create protocol and adversary models to generate proofs demonstrating the robustness of the proposed key management strategy.

2 METHODOLOGY

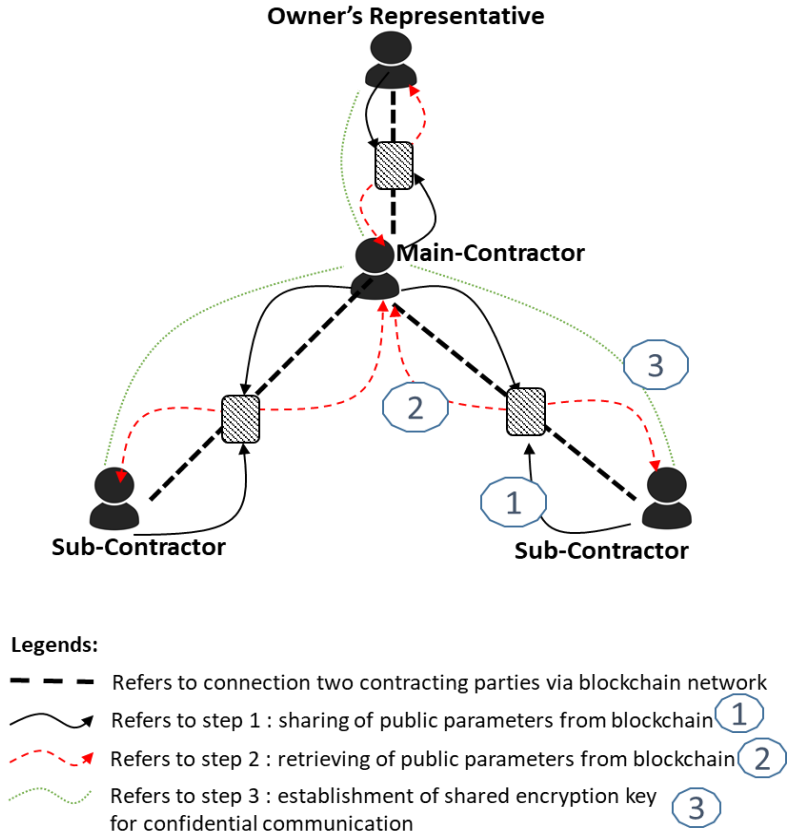

Figure 1 The System Architecture of the Proposed Key Management Strategy

In this section, the methodology of the proposed key distribution strategy for construction projects is presented. Figure 1 shows the system architecture of the proposed key distribution strategy that consists of three parts – (1) sharing of public parameters to a blockchain platform, (2) retrieving of public parameters from the blockchain platform for the generation of a shared encryption key, and (3) establishment of a shared encryption key between two parties interested in sharing confidential information. Section 2.1 introduces the components of public-key cryptography used in the proposed key distribution strategy. Section 2.2 and Section 2.3 present the methodology and the design consideration in the proposed key distribution strategy respectively.

2.1 Encryption for Data Confidentiality and User Authentication

Figure 2 shows two encryption primitives namely, (a) asymmetric or public-key cryptography and (b) symmetric or secret-key cryptography that facilitate security properties, user authentication and data confidentiality respectively. Encryption is

defined as "the cryptographic transformation of data (called "plaintext") into a form (called "cyphertext") that conceals the data's original meaning to prevent it from being known or used. If the transformation is reversible, the corresponding reversal process is called "decryption," which is a transformation that restores encrypted data to its original state" (NIST 2015). Figure 2 (a) illustrates asymmetric encryption (Rivest et al. 1978) in which a cryptographic key pair – a public key and a private key is used, that can be used to represent and verify a user's identity. In this key pair, the public key as the name suggests can be shared with everyone whereas the private key should be kept confidential.

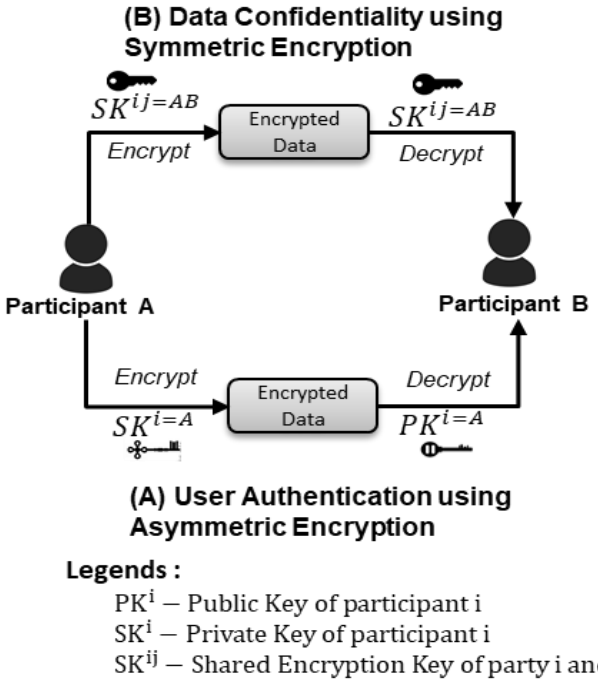

Figure 2 Asymmetric and Symmetric Encryption for User Authentication and Data Confidentiality

In order to use public-key encryption for user authentication, users should first create an asymmetric key pair (PK^i and SK^i as shown in Figure 2(a)) using public-key encryption algorithms such as RSA (Rivest et al. 1978). The public key consists of two random large prime numbers (n_i) and (e_i) (Rivest et al. 1978), whose public disclosure which has no effect on the security. The private key consists of an integer (d_i) derived from the public components by solving an NP-hard problem. The strength of asymmetric encryption lies in the fact that the private component, (d_i) cannot be derived from the public components, (n_i) and (e_i) through trial and error approach. For example, a 3072-bit or longer asymmetric keys will require more than 10 years to be broken through brute force as evaluated by the National Institute of Standards and Technology (NIST) of the U.S. Department of Commerce (Barker and Barker 2019).

Table 1 shows how user authentication and data confidentiality is facilitated by public-key cryptography ('m' and 'c' are plain text and cyphertext respectively in Table 1). As shown in Table 1, for user authentication, a user uses his private key to encrypt a message, more commonly known as a digital signature. This digital signature is can be verified by any public user by decrypting it with the public key of the corresponding

signer, hence authenticating his identity. However, asymmetric encryption is slow and computationally expensive for encryption of large data and therefore is not used to facilitate data confidentiality (Salama et al. 2009). For data confidentiality, symmetric encryption such as AES (Advanced Encryption Standard) (Dobbertin et al. 2004) that uses a single encryption key for both encryption and decryption as illustrated in Figure 2 (b), is used. The robustness of asymmetric and symmetric encryption for user authentication and data confidentiality can be found in well-established standards for encryption (Barker and Barker 2019). Therefore, this paper uses asymmetric and symmetric encryption primitives for user authentication and data confidentiality in the proposed key distribution strategy for construction projects.

Table 1 Public Key Encryption (Rivest et al. 1978)

	Encryption	Decryption
Data Confidentiality	$c = (m)^{e_i} mod(n_i)$	$m = (c)^{d_i} mod(n_i)$
User Authentication	$c = (m)^{d_i} mod(n_i)$	$m = (c)^{e_i} mod(n_i)$

2.2 The Proposed Key Distribution Strategy

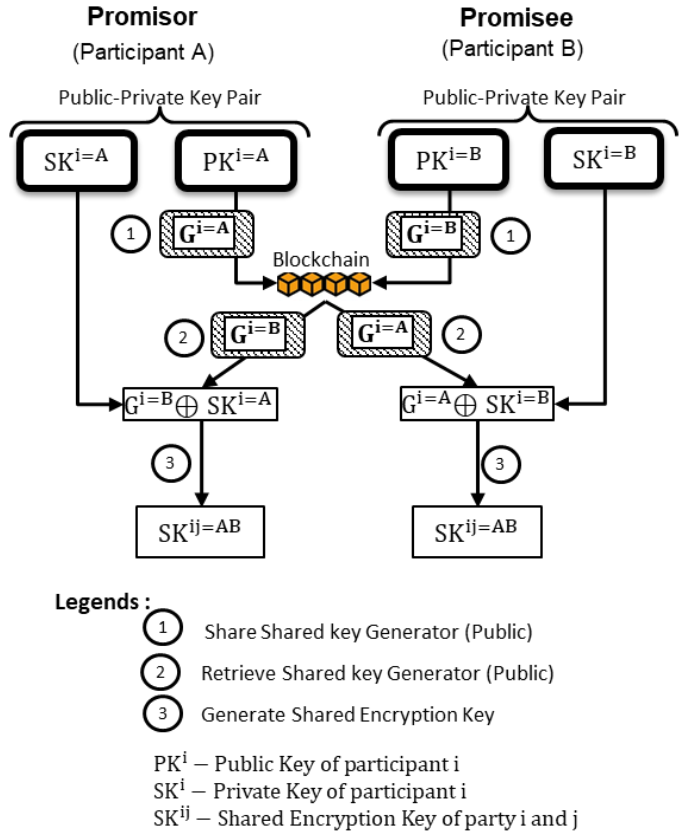

Figure 3 The Proposed Key Distribution Strategy

This section presents the proposed key distribution strategy based on the Diffie-Hellman Key (DH) exchange method (Rescorla 1999), and public-key cryptography standards such as RSA cryptography (Rivest et al. 1978) and Blockchain technology. The

methodology as illustrated in Figure 3 consists of three steps – (1) sharing of a shared key generator to a blockchain platform, (2) retrieving of the shared key generator from the blockchain platform, and (3) establishment of a shared encryption key between two contracting parties. As shown in Figure 3, the contracting parties consist of a promisor (one who promises to pay and is on a higher level in the project organizational hierarchy) and a promisee (one who is entitled to a payment upon completing work and is relatively lower on the project organizational hierarchy) such as a main-contractor and a sub-contractor respectively (as shown in Figure 1).

In this first step, as shown in Figure 3, two contracting parties who are interested in establishing a secure channel for sharing confidential information create generators, $G^{i=A} = (e_A)^{d_A} \bmod(n_A)$ and $G^{i=B} = (e_A)^{d_B} \bmod(n_A)$ each (where e_A and n_A are the public key parameters of the promisor) and share it on a blockchain platform. Once the generators are shared on the blockchain platform, the relevant parties confirm the authenticity of those generators through other channels of communication such as email (discussed further in Section 2.3). In the second step, as shown in Figure 3, the project participants download each other shared key generators. Smart contracts may be deployed for uploading and downloading customized information (such as generators in this case) from blockchain platforms (Ahmadisheykhsarmast and Sonmez 2018, Regnath 2018). In the third step, as shown in Figure 3, a shared encryption key is established between the two participants (promisee and promisor). Both the participants A and B generate a common encryption key, $SK^{ij=AB}$ (as shown in Figure 3) by using their own private keys and shared public parameters such that $SK^{ij=AB} = (G_{i=B})^{d_{i=A}} \bmod(n_A)$ and $SK^{ij=AB} = (G_{i=A})^{d_{i=B}} \bmod(n_A)$.

2.3 Reduction in Key Management Overhead

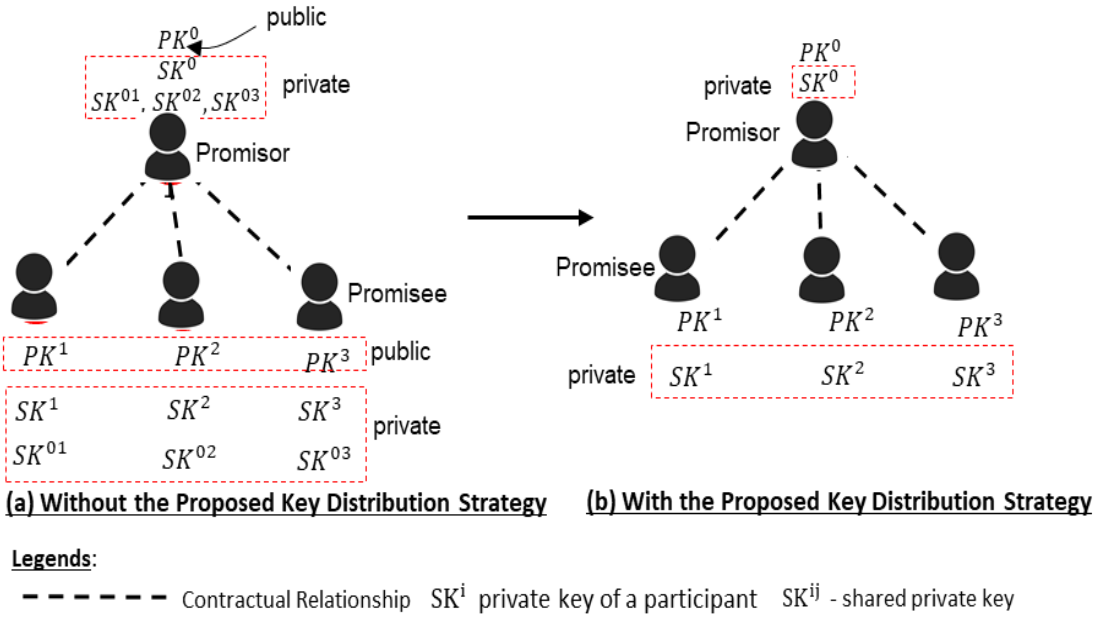

Figure 4 Shared key distribution with and without the proposed key distribution strategy

Figure 4 shows a scenario of shared key distribution between two contracting parties with and without the design considerations of the proposed key management strategy.

Figure 4 (a) shows a case where shared keys are distributed using other methods such as using encryption to protect keys when being transferred through a network. In such cases, due to the hierarchical organizational structure of construction projects with a high degree of sub-contracting, every project participant will have n+1 number of private keys to manage, where n is the number of contractual relationships of a project participant. As shown in Figure 4 (a), the proposed approach reduces the key management overhead to one key per to one per project participant. Furthermore, there may be additional keys to manage if encryption is used to secure key transfer. Therefore, the proposed key distribution strategy provides a method that is unaffected by the security of a network (demonstrated in Section 3) to distribute shared encryption keys. In the proposed approach, shared encryption keys or private keys are never exchanged over a network and therefore, the security of the proposed strategy does not depend on the security of the network.

3 VALIDATION

In this section the security of the proposed key distribution strategy is assessed for two cases – (a) leaking of private keys to an adversary and (b) an adversary posing as an honest user (as shown in Figure 5). Tamarin prover (Basin 2017b), a security protocol verifier tool is used to deploy a symbolic attack model to validate the security of the proposed key management strategy under the cases shown in Figure 5. Tamarin prover has been widely used by researchers for the verification of security protocols (Basin et al. 2015, Dreier et al. 2018). It provides a first-order logic-based modelling language and uses equational reasoning with heuristics for verification and falsification of a symbolic attack model. A protocol model and adversary models may be developed using the constructs, "rule" and "lemma" of the tamarin modelling language followed by the deployment of the tamarin prover engine to validate the protocol against the adversary models. The results prove robustness or security loopholes in a cryptographic protocol.

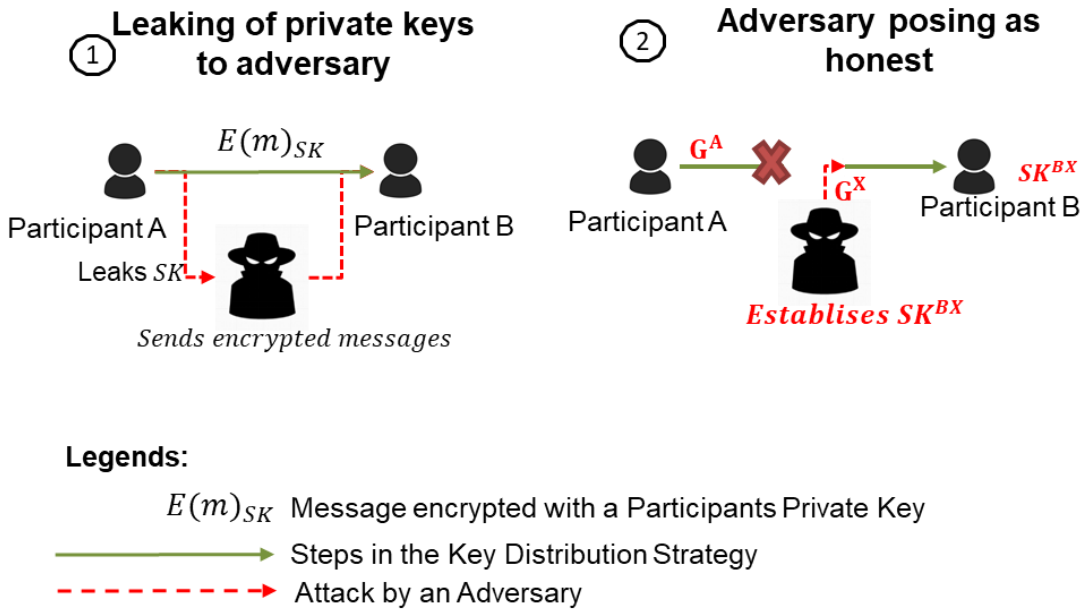

Figure 5 Illustration of Cased of Security Vulnerabilities

Figure 5 (1) shows the case in which an adversary controlling the network can steal private keys during key distribution through the proposed key distribution strategy. Figure 5 (2) shows the case where an adversary poses as an honest user and establishes a shared encryption key to communicate with another honest user and then tricking him into leaking sensitive information. These two cases are modelled using the Tamarin modelling language as shown in Figure 6(b) and Figure 6(c) respectively. The protocol model of the proposed key distribution strategy is shown in Figure 6(a).

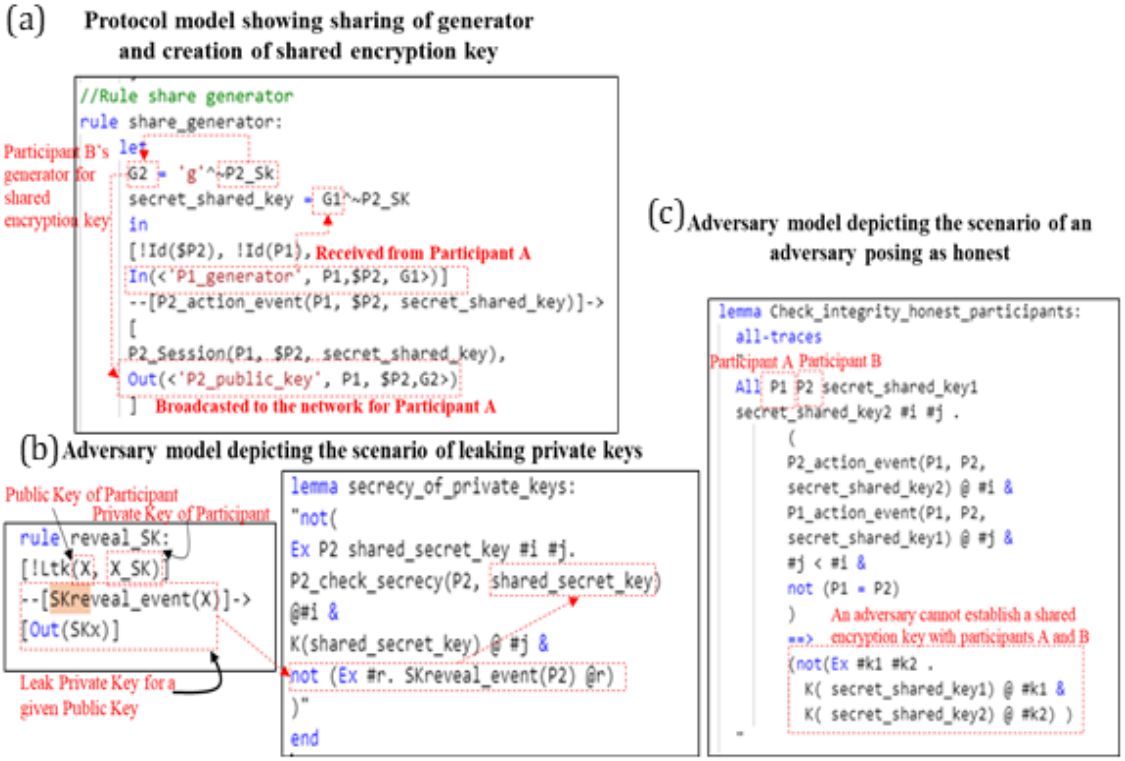

Figure 6 Protocol and Adversary Modelling using Tamarin Prover

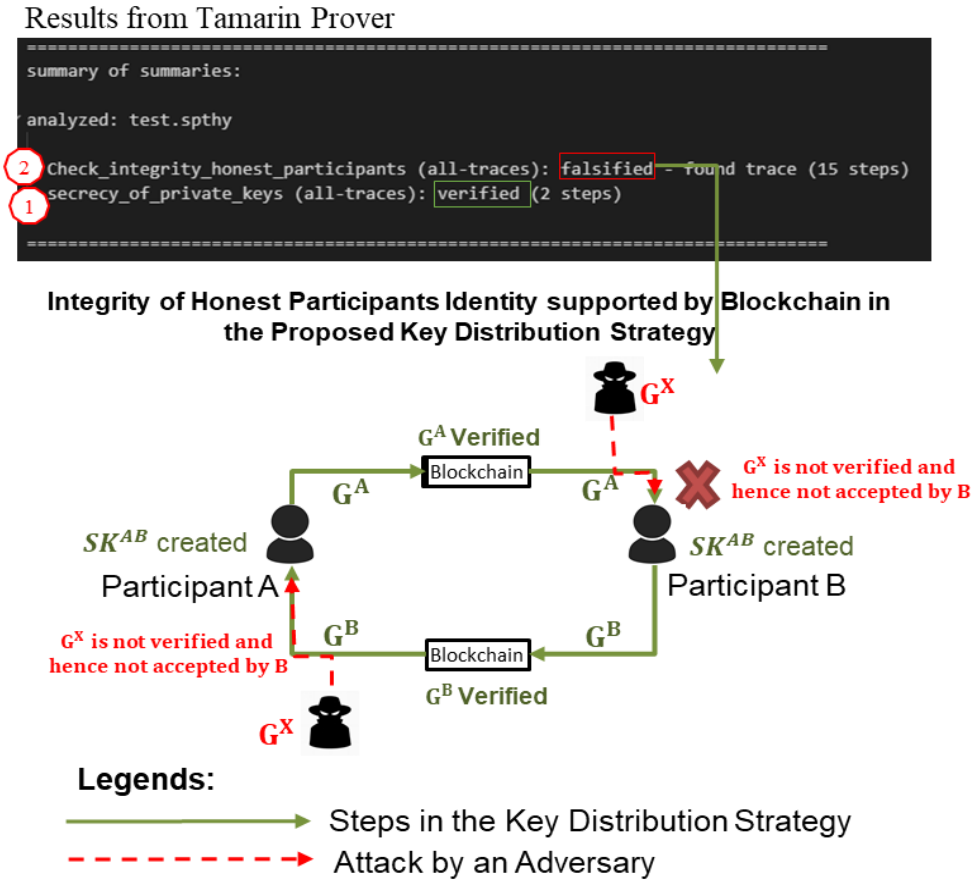

Figure 7 Results from tamarin prover and discussion

Figure 7 shows the results of the execution of the symbolic attack model using Tamarin prover. The results show that the proposed key distribution strategy, at no given point, leaks private key information to any adversary, unless it is purposefully leaked out by a participant (or stolen). This means that the proposed key distribution strategy is suitable for establishing shared encryption keys in compromised networks. Hence as discussed in Section 2.3, it provides the benefit of low-key management overhead with high security.

The results (as shown in Figure 7), however, shows that the proposed key distribution strategy is not resilient to the second security vulnerability scenario where an attacker poses as an honest user. This is because the protocol model (as shown in Figure 6) does not consider the security of identity verification provided by the blockchain platform. The proposed key distribution strategy deploys blockchain's property of immutability for verifying the authenticity of the key generators (as discussed in Section 2.3) by the respective participants. Therefore, as shown in Figure 7, any attempt by adversaries to intercept honest communication and inject his own key generators to establish a shared encryption key to communicate with honest users is prevented by the honest users (in contrary to the corresponding case shown in Figure 5(2)).

4 Conclusion

Construction projects consist of project participants who are bound by contractual relationships. Although it may be favourable to share project information among all

project participants to facilitate transparency and hence smooth project execution, some sensitive information is required to be kept confidential between contracting parties. Therefore, this paper presents an encryption key distribution strategy for sharing sensitive information using public blockchain platforms. The proposed key distribution strategy facilitates user authentication and confidential information sharing between two contracting parties with minimum key management overhead. The robustness of the key management strategy is demonstrated through a symbolic attack model and the results are discussed. The security of the proposed key management strategy for some cases, however, depends on the choice of the blockchain platform. It is designed to be deployed using large public blockchain platforms that have a network size of 8000~10000 nodes and use probabilistic consensus algorithms such as PoW (Proof-of-Work) and PoS (Proof-of-Stake) as in Bitcoin and Ethereum blockchain platforms. Such blockchain platforms facilitate high security in terms of the irreversibility of records which is required by the proposed key distribution strategy. Smaller public blockchain platforms and permissioned blockchain platforms, however, may not be able to provide high immutability compared to large public blockchain platforms due to low total network computational power and use of deterministic consensus algorithms respectively. However, the architecture of permissioned blockchains is preferred and being investigated for private organizations. Therefore, in the future, the proposed key distribution strategy will be extended with additional security measures for using permissioned blockchain platforms. The additional parameters of security that should be deployed to address various security threats on permissioned blockchain platforms for construction projects will be explored in the future.

5 REFERENCES

ACONEX (2018). Cloud-Based Information Management: A Guide for IT (WhitePaper), Available at: [Accessed 28 Jan. 2020]

Ahmadisheykhsarmast, S., and Sonmez, R. (Ed.) (2018) Smart Contracts in Construction Industry

Ali, M., Dhamotharan, R., Khan, E., Khan, S.U., Vasilakos, A.V., Li, K., and Zomaya, A.Y. (2017) SeDaSC: Secure Data Sharing in Clouds, IEEE Systems Journal, 11, (2), pp. 395-404

Autodesk, BIM 360, Available at: https://bim360resources.autodesk.com/whitepapers/bim-360-security-whitepaper [Accessed 28 Jan. 2020]

Barker, E. and Barker, W.C. (2019) Recommendation for Key Management, NIST Special Publication 800-57 Part 2, 2019, doi: doi.org/10.6028/NIST.SP.800-57pt2r1

Basin, D., Cremers, C., and Dreier, J. and Sasse, R. (2017) Symbolically Analyzing Security Protocols using Tamarin, ACM SIGLOG News, 4 (4), pp. 19-30, doi: doi.org/10.1145/3157831.3157835

Basin, D., Dreier, J., and Sasse, R. (Ed.) (2015) Automated Symbolic Proofs of Observational Equivalence, pp. 1144–1155

Chen, C.-M., Zheng, X., and Wu, T.-Y (2014) A Complete Hierarchical Key Management Scheme for Heterogeneous Wireless Sensor Networks, The Scientific World Journal, 2014, pp. 816549

Conti, M., Dragoni, N., and Lesyk, V. (2016) A Survey of Man in the Middle Attacks, IEEE Communications Surveys & Tutorials, 18, (3), pp. 1

Crypto51, Available at: https://www.crypto51.app/ [Accessed 28 Jan. 2020]
Dobbertin, H, Rijmen, V. and Sowa, A. (2004) Advanced Encryption Standard – AES, 4th International Conference, Lecture Notes in Computer Science, Vol. 3373, Springer, Bonn, http://doi.org/10.1007/b1377659
Dolev, D. and Yao, A.C (1983) On the security of public key protocols., IEEE Transactions on Information Theory 29 (12), pp. 198–208, doi.org/10.1109/TIT.1983.1056650
Dreier, J., Hirschi, L., Radomirovic, S., and Sasse, R. (Ed.) (2018) Automated Unbounded Verification of Stateful Cryptographic Protocols with Exclusive OR (Extended Version)
Indu, I., Anand, P.M., and Shaji, S.P. (2016) Secure File Sharing Mechanism and Key Management for Mobile Cloud Computing Environment', Indian Journal of Science and Technology
Kahvazadeh, S., Masip-Bruin, X., Diaz, R., Marín-Tordera, E., Jurnet, A., and Garcia (Ed.). (2018) Towards An Efficient Key Management and Authentication Strategy for Combined Fog-to-Cloud Continuum Systems, pp. 1-7
NIST (National Institute of Standards and Technology) (2015), Guide to Industrial Control Systems (ICS) Security, NIST Special Publication 800-82 2015, Available at: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-82.pdf, [Accessed 25 Apr. 2020]
PMWeb, PMWeb, Available at: https://pmweb.com/ [Accessed 28 Jan. 2020]
Regnath, E. and Steinhorst, S. (2018) SmaCoNat: Smart Contracts in Natural Language, Forum on Specification & Design Languages (FDL), Garching, pp. 5-16., doi: doi.org/10.1109/FDL.2018.8524068
Rescorla, E. (1999) Diffie-Hellman Key Agreement Method, Proposed Standard, RFC 2631, doi: doi.org/10.17487/RFC2631
Rivest, R.L., Shamir, A., and Adleman, L. (1978) A method for obtaining digital signatures and public-key cryptosystems, Communications of the ACM, 21, (2), pp. 120-126
Salama, D. D., Abdelkader, H., and Hadhoud, M.M. (2009) Performance Evaluation of Symmetric Encryption Algorithms, Communications of the IBIMA, 10
Studnia, I., Alata, E., Deswarte, Y., Kaâniche, M., Vincent, N. (Ed.) (2012) Survey of Security Problems in Cloud Computing Virtual Machines, pp. 61-74
Zhang, S., and Lee, J.-H (2019) Analysis of the main consensus protocols of blockchain, ICT Express, https://doi.org/10.1016/j.icte.2019.08.001