ISSN: 0975-833X

Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 11, Issue, 04, pp.3274-3282, April, 2019

DOI: https://doi.org/10.24941/ijcr.35209.04.2019

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

RESEARCH ARTICLE

DSSE: DISTRIBUTED SECURITY SHIELDED EXECUTION FOR COMMUNICABLE CYBER THREATS ANALYSIS

Satinderjeet Singh, Anil Lamba and *Sivakumar Sai Rela Muni

SR-MLC: Scalable Resilience Machine Learning Classifiers Approach in Cyber Security

ARTICLE INFO

Article History:
Received 11th January, 2019
Received in revised form
17th February, 2019
Accepted 14th March, 2019
Published online 30th April, 2019

Key Words:

Cloud Computing, Cyber Security, Network, Cyber, Cyber Threats, Threat Analysis, Information Security, Data security, Distributed Security Shielded Execution (DSSE) Secure Identity Based Encryption (SIBE), Multi-Cloud Environment.

***Corresponding author:* Sivakumar Sai Rela Muni**

ABSTRACT

Cloud computing is a new computing model which enables individuals and organizations to attain access to huge computing resources without capital investment. It does mean that users can utilize computing resources in pay per use fashion. Transmissible cyber threats nowadays have been considered one among the major serious security problems in cyberspace. Several techniques were proposed to model, simulate and identify threats' sources and their propagation in large-scale distributed networks. Most techniques are based on the analysis of real networks dataset that contains sensitive information. Traditional in-memory analysis of these dataset always leads to data leakage because of system vulnerabilities. If the dataset itself is compromised by adversaries, this threat cost would be even higher than the threat being analyzed. To overcome this problem, in this paper, a new Distributed Security Shielded Execution (DSSE) for communicable cyber threats analysis using the Secure Identity Based Encryption (SIBE) technique is proposed. The purpose of the proposed work is to provide security sensitive operations then implemented properly in the Disef enclave. And to minimize potential Iago attacks in Disef system. To introduce a Novel SIBE unbreakable new key-value encrypted format integrated with the version number and update counter to prevent possible rollback and replay attacks. Also to secure a Disef system in high availability with multi-cloud support. The experimental outcomes showed that the proposed framework supports secure cost analysis and response time analysis of large network dataset and has comparable performance with systems that have no confidentiality and integrity guarantees.

***Citation: Satinderjeet Singh, Anil Lamba and Sivakumar Sai Rela Muni,* 2019.** "Dsse: distributed security shielded execution for communicable cyber threats analysis", *International Journal of Current Research*, 11, (04), 3274-3282.

INTRODUCTION

Identity Based Encryption (IBE) provides a public key encryption mechanism in which a public key is an arbitrary string. It can be an email address or a telephone number. The corresponding private key is possible to be generated by a Private Key Generator (PKG) who has knowledge of a master secret. In an IBE system, users authenticate their selves to the PKG and get private keys according to their identities. Though the identity based encryption model was proposed twenty years ago (Shamir, 1984), and a few early precursors suggested over the years (Tanaka, 1987; Tsujii, 1989; Maurer, 1996), it is only currently that the first working implementations were proposed. Boneh and Franklin (Boneh, 2001; Boneh, 2003) introduced a security model for identity based encryption and provided a construction based on the Bilinear Diffie-Hellman (BDH) issue. Cocks (Cocks, 2001) proposed another construction by quadratic residues modulo, a composite. For security of these systems, cryptographic hash functions are needed that are modeled as random oracles, i.e., these systems are only proven security in the random oracle model (Bellare, 1993). The same holds for many other identities based systems featuring signatures (Choon, 2003), key exchange (Ogishi *et al.,* 2007), hierarchical identities (Ryuichi Sakaim, 2000), and signcryption (Gentry, 2002). It is natural to ask whether secure IBE systems is possible to exist in the standard model, i.e., without resorting to the random oracle heuristic. This question is specifically corresponding to various current uninstantiable random oracle cryptosystems (Boyen, 2003; Canetti, 1998), which are secure in the random oracle model, but are probably without security as per any actual instantiation of the oracle. Towards this result, many recent outcomes (Bellare *et al.,* 2004; Canetti, 2003; Boneh, 2004) construct IBE systems securely without random oracles in weaker versions of the Boneh-Franklin model. In one such model, called "selective-ID" IBE (Bellare *et al.,* 2004), the adversary must commit beyond certain time to the identity it desires to attack. Existing security sensitive operations are not implemented properly in theDisef enclave. Existing Disef allows one secure interface between the enclave and non-enclave space to reduce potential Iago attacks. The new key-value encrypted format and integrated with the version number and update counter to prevent possible rollback and replay attacks are inefficient. Existing Disef system are not secure the high availability with multi-cloud support. Many business enterprises are migrating towards Multi-Cloud to store the enterprise data to meet these growing demands of the consumers. Multi-Cloud approach solves several limitations faced by single cloud environment.

The benefits of the work include freedom from vendor-lock-in, improved availability, interoperability, and data security.

Our contribution

The purpose of the proposed work is to

- To provide security sensitive operations then implemented properly in the Disef enclave.
- To minimize potential Iago attacks in Disef system.
- To introduce a Novel SIBE unbreakable new key–value encrypted format and integrated with the version number and update counter to prevent possible rollback and replay attacks.
- To secure a Disef system in high availability with multi-cloud support.

This paper is organized as follows: Section 2 gives the Literature review; Section 3 describes the proposed methodology i.e., Secure Identity Based Encryption (SIBE)**.** Section 4 discusses the results followed by the conclusion and future work in section 5.

LITERATURE REVIEW

Cheng et al, (2018) proposed a new distributed shielded execution framework (Disef) for cyber threats analysis. The Disef framework enables efficient distributed analysis of network dataset while achieves security guarantees of data confidentiality and integrity. In-memory dataset is protected by using a new encrypted key–value format and could be efficiently transferred into Intel SGX enabled enclaves for shielded execution. The investigational outcomes showed that the proposed framework supports secure in-memory analysis of large network dataset and has comparable performance with systems that have no confidentiality and integrity guarantees (19). Rehman *et al.* (2018) explored about why the volume and severity of cyber attacks are far beyond with the abilities of their alleviation methods and how the preventive safety measures could decrease the losses from cybercrime for specific type of attacks in future. It further expressed the necessity to have an improved technological vision and stronger defenses, to change the picture where human thought might be the subsequent big weapon as a security assurance toolkit (Rehman *et al.,* 2018).

Liang *et al.* (2018) proposed a new, distributed block chain-based protection framework to enhance the self-defensive capability of modern power systems against cyber-attacks. A comprehensive discussion was surveyed on how block chain technology could be utilized to improve the robustness and security of the power grid, by using meters as nodes in a distributed network that encapsulated meter measurements as blocks. Proposed protection framework efficiency was validated through simulation experiments on an IEEE-118 benchmark scheme (Liang *et al.,* 2018). Diaz et al, (2018) proposed a complete study about security issues in IoT devices, studying the most common security events, susceptibilities, and attack surfaces. Precisely, a multirelations mapping among these three categories was proposed which might be useful to support the security administrator to better comprehend the causes, symptoms, and attack vectors of security incidents. Specially, security events have been utilized to create correlation rules which were preferred through a SIEM system to distinguish security incidents. Defensive security was explored by describing dissimilar probable actions that could be executed from a SIEM so as toreduce the influence of an incident or evade the development of the incident (Díaz López *et al.,* 2018). Ezhei *et al.* (2018) proposed a differential game among the networked firms in which attackers act strategically. In the proposed game, by employing a linear substitution model for characterizing the process of target selection by the attacker, the open-loop Nash solutions are highlighted in an analytical form. The analytical results show how interconnectivity between firms and the strategic behavior of the attacker determines the firms' incentives for security investment. It was showed that overinvestment or underinvestment could occur depending on the degree of interdependency among the given firms. Accordingly, mechanisms were designed to encourage the firms to invest at a socially optimal level. The achieved results in the paper helps security designers to better formulate their policies in tackling strategic attackers (Ezhei, 2018). Sauerwein *et al.* (2018) conducted a structured, in-depth analysis of shadow cyber threat intelligence and investigate how it was used in organizations. The study revealed that many heterogeneous and overlapping cyber security information sources serve as input for information security and risk management processes and that the obtained shadow threat intelligence was shared internally in a largely unstructured and informal manner (Sauerwein *et al.,* 2018).

METHODOLOGY

Identity-Based Cryptography (ID-PKC): Identity-based cryptography uses publicly known identifiers such as user email-id, IP addresses and organization-name for public key generation. Identity-based encryption also known as IBE was initially proposed by Adi Shamir in 1984, it was an open problem until 2001. The Boneh-Franklin scheme (Boneh, 2001) is based on pairings but Cock's encryption scheme (Cocks, 2001) used quadratic residues, both the approaches proposed solution to IBE but Cocks scheme is inefficient because it generates large ciphertexts compared to pairing based systems. In ID-PKC, private key generator (PKG) is responsible for generation of public keys using master private key for a given identity of the participating entity. The Weil or Tate pairings are used for bilinear pairing generation using elliptic curves efficiently.

Protocol Framework: ID-PKC comprises following four algorithms (Boneh, 2001) having polynomial time complexity:

- **Setup:** Initially the PKG generates master key pair (related public and private key) denoted as sk_{PKG} and pk_{PKG} respectively. Setup is a randomized algorithm which returns system parameters (which are later published to all communicating entities) using the security parameter k along with the master secret key msk. The system parameters include message space M, the ciphertext space C and the identity space I and the master public key (mpk). The master secret key is kept local to private key generator (PKG)
- **Private Key Extraction or Key Derivation:** The receiver (for example Bob) authenticates to Private key generator and obtains a private key sk_{IDBob} associated with identity ID_{Bob} and along with system parameters

- **Encryption:** Using receiver's identity (here ID_{Bob}) and master public key pk_{PKG}, the sender encrypts a message M and converts to ciphertext C which is then transmitted
- **Decryption:** At receiver side the cipher text C is decrypted using receiver private key sk_{IDBob} and recovers the plaintext M.
- An inverse operation is used to obtain the Identity-based Signature (IBS) scheme which is used for digital signatures. The sender of message initially obtains a signing key associated with identifier information from the PKG and then signs the message, then verifier uses senders' identifier information for signature verification.

Advantages of ID-PKC cryptosystems

Following are some of the advantages associated with ID-PKC schemes:

- As the public keys are derived from user identification variables, there is no need for infrastructure for public key delivery and CRL management. The public keys are trusted as long as the keys are transferred securely and corresponding user private key is kept secret.
- ID-PKC schemes allow encoding additional information to the user keys or cipher text. For example, it is possible for sender to specify an expiration date of a message by appending timestamp information to the recipient's identity.
- Given finite number of users, ID-based encryption allows destroying third-party secret once all the users are issued secret keys assuming there is no revocation of keys henceforth.

PROPOSED METHODOLOGY

Distributed Security Shielded Execution for communicable cyber threats analysis using Secure Identity Based Encryption (SIBE): SIBE is used to secure the key-value in the effective manner and providing the better attack prevention result. The pseudo code for the proposed Distributed Security Shielded Execution for communicable cyber threats analysis using Secure Identity Based Encryption algorithm is given in this section. The test set size considered is computed by using the below given formula.

$$s_{testsetsize} = \sum_{i=0}^{testsetsize} (\text{testset}(i) < i) \quad (1)$$

$$\Phi(z) = \frac{1}{1+e^{-1}} \quad (2)$$

Deep recurrent neural networks (DRNN): Deep Recurrent Neural Networks (DRNN) Deep neural networks consist of more than a hidden layer of neurons. Several studies (22-23) suggest that with more hidden layers, the neural network is capable of representing complex function more efficiently than RNN with less hidden layers. Fig. 1 shows the DRNN with a deep transition. The deep transition raises the number of nonlinear steps, which might increase the complexity of training such a network.

In this study, attribute selected dataset is initialized and then configured the parameters with a length of 20 and the number of epochs are 30000 in addition to weights ranging from -100 to 100. Then the recurrent neural network is generated by means of using the equation (2). Further, the samples are generated from the trained set.In addition to this, the test set and the trained set values are validated and then evaluated the process of learning by means of considering the above mentioned equation (1). The classification techniques are endorsed and initiated the greedy select algorithm for i, which selects the first activity in a constant manner. Then also initiated for j<n which considers the rest of the activities. The selection of the activity time will be greater than or equal to the finish time of previously selected activity i.e., $s[j] \geq f[i]$ using the equation (1). The process of recurrent neural network is ended and the new RNN process is started which further generates or forms the input, hidden and the output layers with sigmoid equation (2). Next, the net core generator was generated and in case if it is false, then the last layer will be -1. Update the last layer using network generator of the input layer in RNN and link the last layer and the current layer. Verify the hidden layer, if last layer is less than 0, then it means that there is no defined input layer. Update the last layer and verify the hidden layer which create the feed forward links and recurrent links and enter into the return layer. Use final value of Boolean bias and double bias in the output layer.Generate the bias of the net core generator with double values and then end the process of DRNN.

Algorithm 1: SIBE

```
begin
ALGO = "SIBE";
TRANS = "SIBE";
Docipher (key, input File output File)
begin
do Crypto (Cipher. ENCRYPT_MODE, key, input File, output File);
end
doplain (key, input File, output File)
begin
do Crypto (Cipher. DECRYPT_MODE, key, input File, output File);
end
doCrypto (cipher Mode, key, input File, output File)
begin
secretKey ← new Secret Key Spec (key.get Bytes(), ALGO);
cipher ← Cipher.get Instance(TRANS);
length ← Cipher.get Max Allowed Key Length("SIBE");
cipher.init (cipher Mode, secret Key);
input Stream ← new File Input Stream (input File);
input Bytes [] ← new byte [(int) input File. length()];
input Stream. read (input Bytes);
```

Security Controller:

```
output Bytes [] ← cipher.doFinal (input Bytes);
output Stream ← new File Output Stream (output File);
output Stream.write (output Bytes);
input Stream.close();
output Stream.close();
end
```

Step 1: Initiate attribute_selected_dataset
Step 2: Configure the parameter with length=20, epochs=30000 and weighs = -100 to 100.
Step 4: Generate the RNN network using Equation (2).
Step 6: Generate the samples from the trained set.
Step 7: Validate the test set and trained set values.
Step 8: Evaluate the learning process using Equation (1).
Step 9: Endorse the classification technique.
Step 10: Initiate the greedy select algorithm.

- For i, select the first activity constantly
- For j<n, consider rest of the activities

Step 11: Select the activity time greater than or equal to the finish time of previously selected activity. $s[j] \geq f[i]$ using Equation (1)
Step 12: End the process of RNN.
Step 13: Generate new RNN process.

Algorithm 2: DRNN

Step 14: Generate the input layer, hidden layer and output layer with Sigmoid Equation (2).
Step 15: Next, Generate the net core generator. If it is false, then, last layer will be -1.
Step 16: Update the last layer using Network generator of the input layer in RNN.
Step 17: Link the last layer and the current layer
Step 18: Verify the hidden layer, if last layer < 0, there is no defined input layer.
Step 19: Update the last layer and verify the hidden layer
Step 20: Create the feed forward links and recurrent links and enter into the return layer.
Step 21: Use final value of Boolean bias and double bias in the output layer.
Step 22: Generate the bias of the net core generator with double values.
Step 23: End the process of DRNN

RESULTS AND DISCUSSION

This section discusses the results and discussion which are obtained from the implementation. It mainly discusses the running process, home panel, configuration analysis, Loading process of all Cloud V Mdatabase user requests, Loading process of all Cloud VM database available cloud service providers VM configurations, choosing the service broker policy using the identity based encryption deep recurrent neural network / SIBE deep recurrent neural network, assigning service broker policy, internet configuration and finally the run analysis.

Run: The running process can be done by right clicking on the application of the DSSE and by selecting the cloudsim.ext. gui file and then selecting the GuiMain.java option and further need to click on “Run As” option. Finally,the java application is selected for running the whole DSSE application.

Home Panel: The home panel contains four options i.e., configure analysis, define internet characteristics, run analysis and exit option.

Configure Analysis: The configuration analysis of the DSSE application includes the main configuration, cloud service provider configuration and advanced options. It also shows the particulars of the simulation duration, cloud users and then the application deployment configuration. The simulation duration is set to 60 minutes and the options under the cloud users are name, region, OS type, etc. along with its loading and removing options. The application deployment configuration mainly includes the option to select the policy of the service broker which further includes the data center, virtual machines, image size, memory and bandwidth details. It also has an option for loading and removing of the particulars.

Main Configuration Load Cloud Users Requests: The process can be achieved by clicking on the cloud users and then by loading all the Cloud VM database user requests in a successful manner.

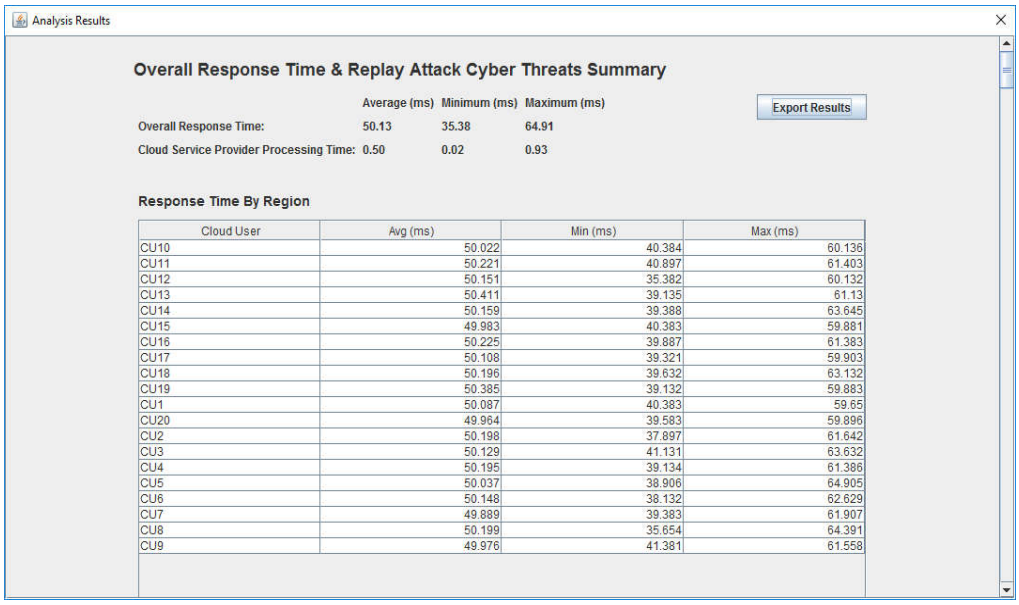
Cloud Service Provider Configuration Load All Service Providers: In this process, the data centers include the particulars of name, region, OS, VMM, cost per VM dollar per hour, cost of the memory, storage cost, data transfer cost, physical HW units, time of the total service given along with loading and removing options. The values are loaded for cloud service provider configuration as shown in the below figure 7. This can be done by clicking on the load in cloud service provider configuration tab and then loaded all Cloud VM database available cloud service providers VM configurations successfully.

Configure Panel: The configuration panel of the DSSE application can be achieved by choosing the service broker policy and then clicking on IBE Deep Recurrent Neural Network / SIBE Deep Recurrent Neural Network.

Assign Service Broker Policy: The assignment of the service broker policy can be achieved by selecting the SIBE deep recurrent neural network option in the service broker policy. This can further configure application deployment configuration for all data centers.

Final Configuration: The configuration process of the cloud users and cloud service providers which are represented in red and blue colors. The red color indicates the cloud service providers and the blue color points indicates the cloud users. Lastly, the done option is clicked to complete the phase of configuration. Then internet configuration is done. The run analysis of the application which can be achieved by clicking on the run simulation and then clicking on simulation started successfully. In background SIBE Public Key Cryptography invoke and processed all cloud users request securely. Invoke DRNN to train and test the cloud service provider to predict side channel attacked cloud service providers. Finally, penalty cost for affected cloud service providers is calculated as shown in performance results below.

Performance metrics: This section mainly includes the performance metrics like overall response time and cloud service provider processing time, cloud users hourly average response times, CSP hourly average processing times and CSP request servicing response time.The below Figures 2, 3, 4, 5, 6 and 7 show the perfomance metrics results. Figure 8 shows the performance comparison of different key–value operations under the Enc KV, and the Disef, and DSSE systems. The same type of operation 1000 times is executed and its average performance is calculated.



Cloud User	Avg (ms)	Min (ms)	Max (ms)
CU10	50.022	40.384	60.136
CU11	50.221	40.897	61.403
CU12	50.151	35.382	60.132
CU13	50.411	39.135	61.13
CU14	50.159	39.388	63.645
CU15	49.983	40.383	59.881
CU16	50.225	39.887	61.383
CU17	50.108	39.321	59.903
CU18	50.196	39.632	63.132
CU19	50.385	39.132	59.883
CU1	50.087	40.383	59.65
CU20	49.964	39.583	59.896
CU2	50.198	37.897	61.642
CU3	50.129	41.131	63.632
CU4	50.195	39.134	61.386
CU5	50.037	38.906	64.905
CU6	50.148	38.132	62.629
CU7	49.889	39.383	61.907
CU8	50.199	35.654	64.391
CU9	49.976	41.381	61.558

Figure 2. Response time by region

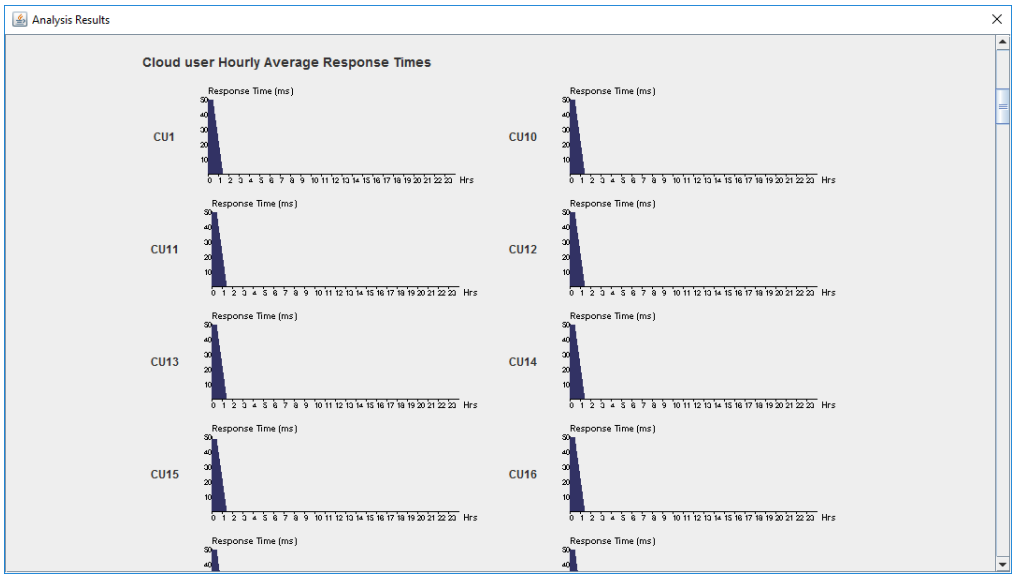
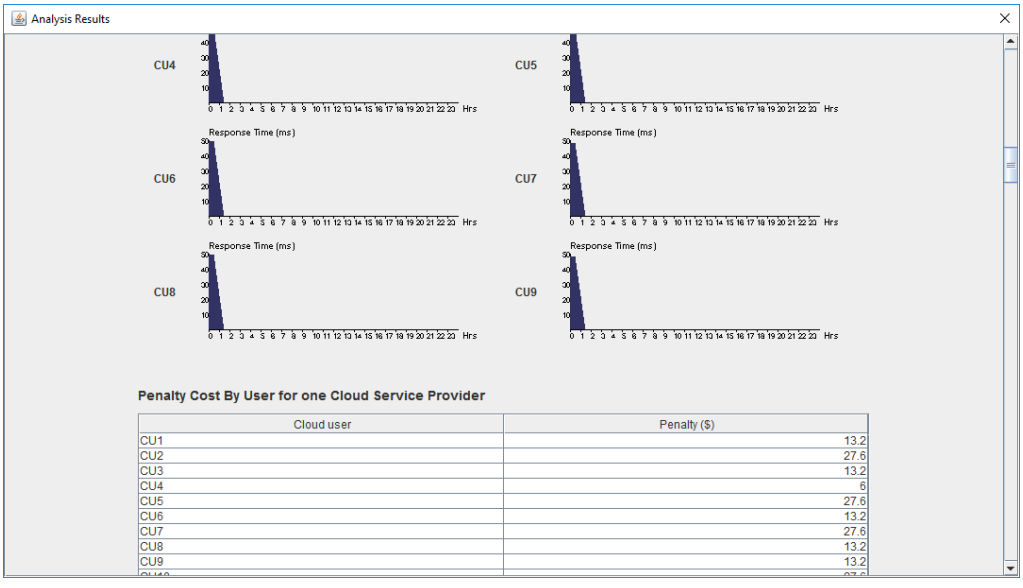
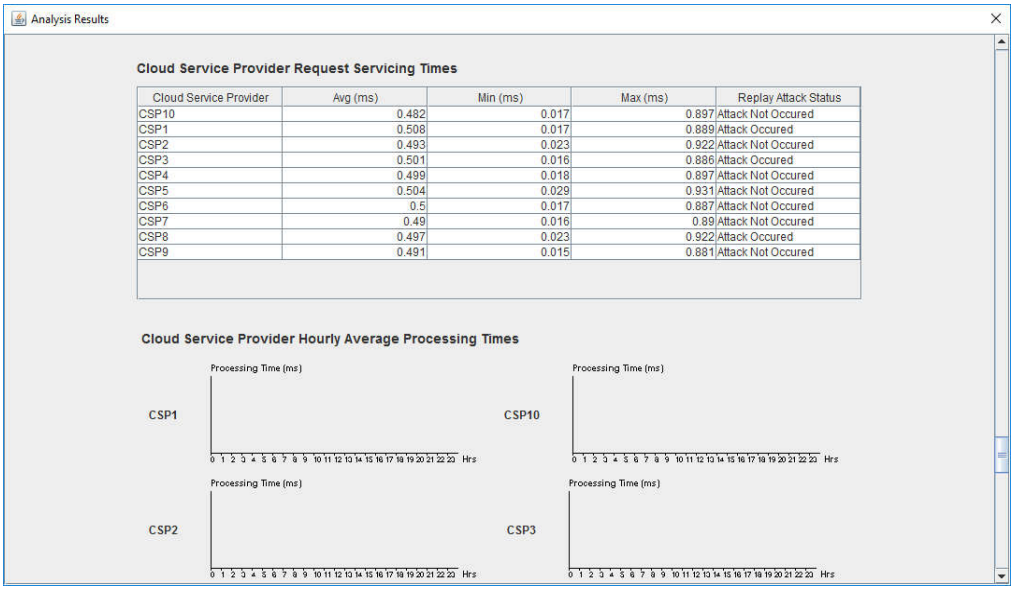


Figure 3. Cloud user hourly average response times



Cloud user	Penalty ($)
CU1	13.2
CU2	27.6
CU3	13.2
CU4	6
CU5	27.6
CU6	13.2
CU7	27.6
CU8	13.2
CU9	13.2

Figure 4. Cloud user hourly average response time



Cloud Service Provider	Avg (ms)	Min (ms)	Max (ms)	Replay Attack Status
CSP10	0.482	0.017	0.897	Attack Not Occured
CSP1	0.508	0.017	0.889	Attack Occured
CSP2	0.493	0.023	0.922	Attack Not Occured
CSP3	0.501	0.016	0.886	Attack Occured
CSP4	0.499	0.018	0.897	Attack Not Occured
CSP5	0.504	0.029	0.931	Attack Not Occured
CSP6	0.5	0.017	0.887	Attack Not Occured
CSP7	0.49	0.016	0.89	Attack Not Occured
CSP8	0.497	0.023	0.922	Attack Occured
CSP9	0.491	0.015	0.881	Attack Not Occured

Figure 5. Cloud service provider request servicing times

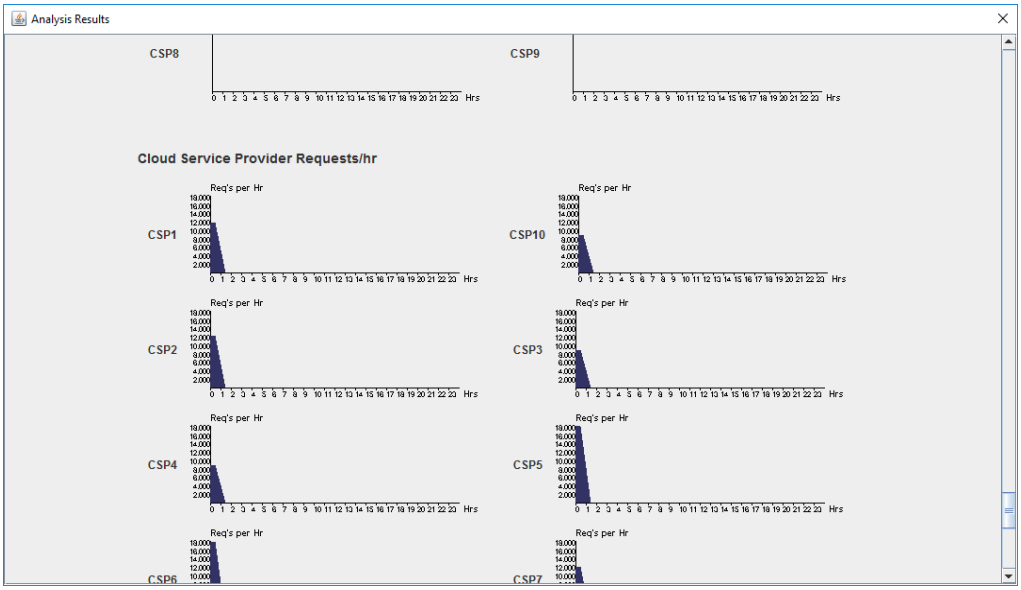
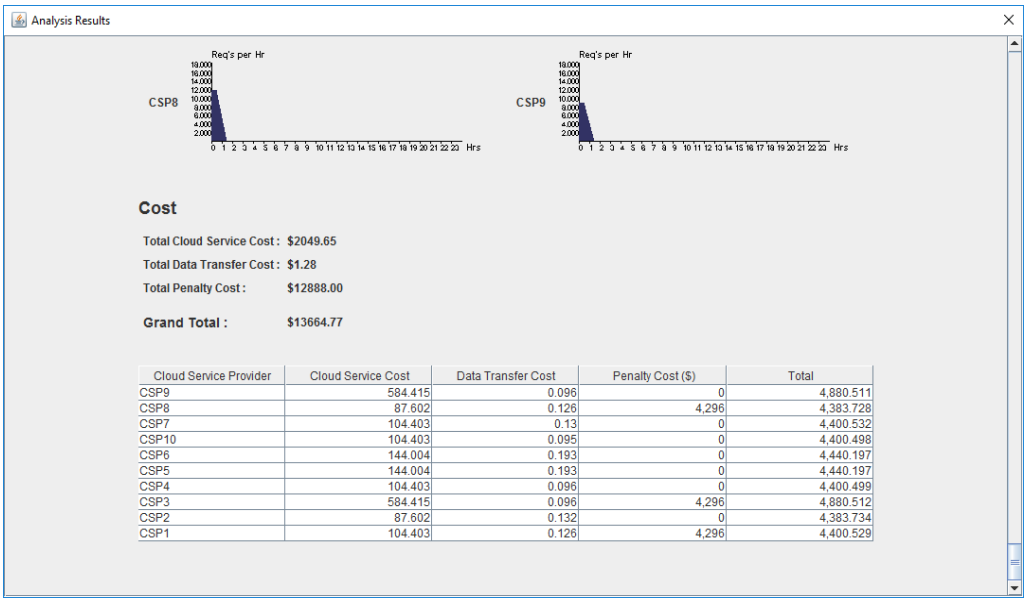


Figure 6. Cloud service provider requests per hour



Cloud Service Provider	Cloud Service Cost	Data Transfer Cost	Penalty Cost ($)	Total
CSP9	584.415	0.096	0	4,880.511
CSP8	87.602	0.126	4,296	4,383.728
CSP7	104.403	0.13	0	4,400.532
CSP10	104.403	0.095	0	4,400.498
CSP6	144.004	0.193	0	4,440.197
CSP5	144.004	0.193	0	4,440.197
CSP4	104.403	0.096	0	4,400.499
CSP3	584.415	0.096	4,296	4,880.512
CSP2	87.602	0.132	0	4,383.734
CSP1	104.403	0.126	4,296	4,400.529

Figure 7. Cost analysis

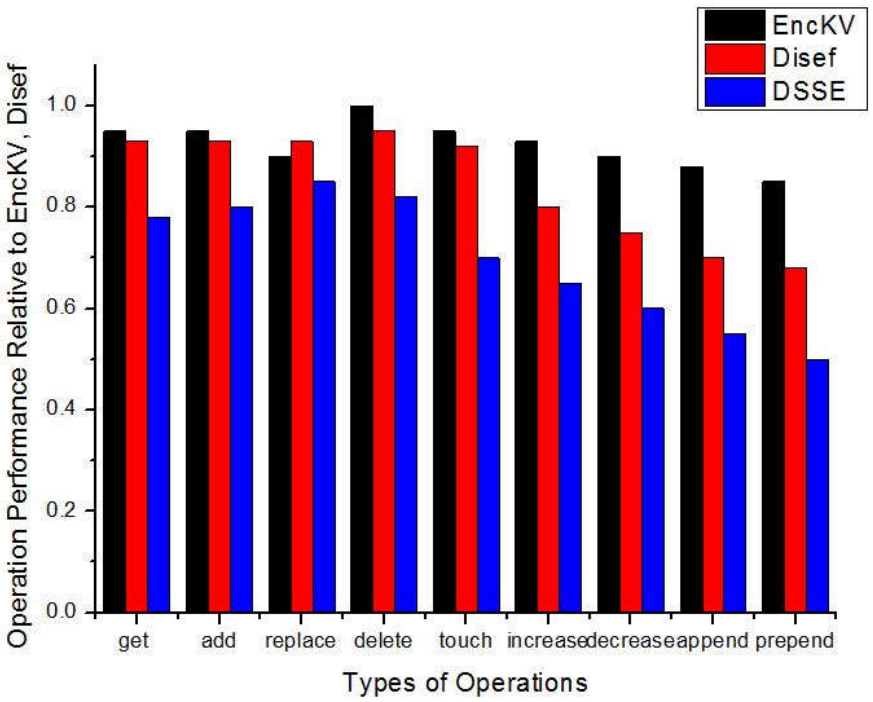


Figure 8. Each individual key–value operation's relative performance compared to the BaseKV system

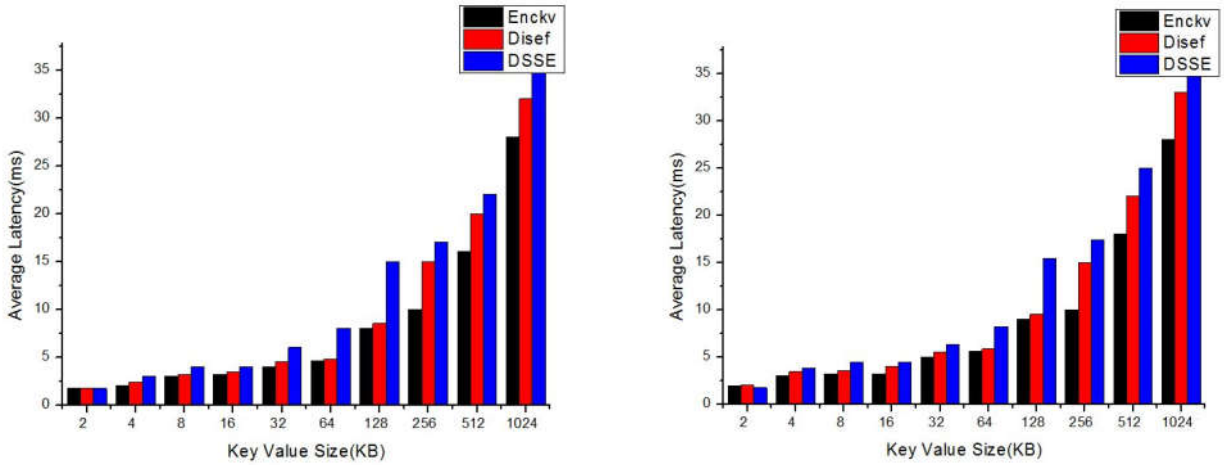


Figure 9. The key–value operation performance comparisons between Base KV, EncKV and Disef. Figure (a) represents the SET operations average latency due to the key–

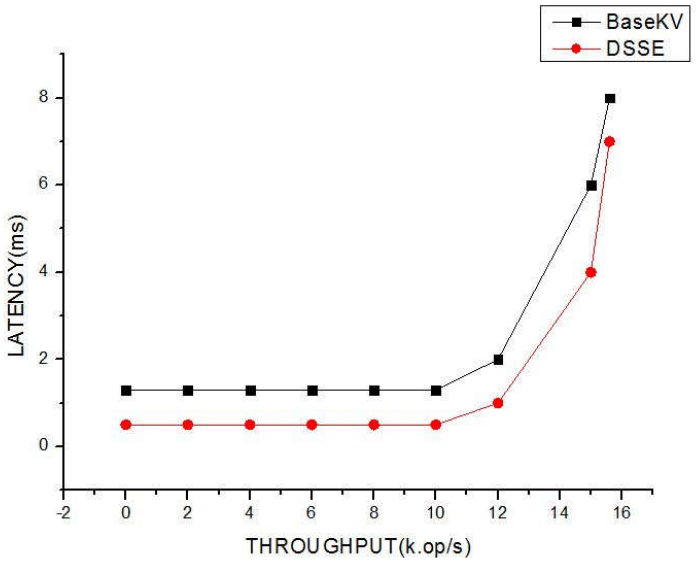


Figure 10. Throughput versus latency of three key–value store systems

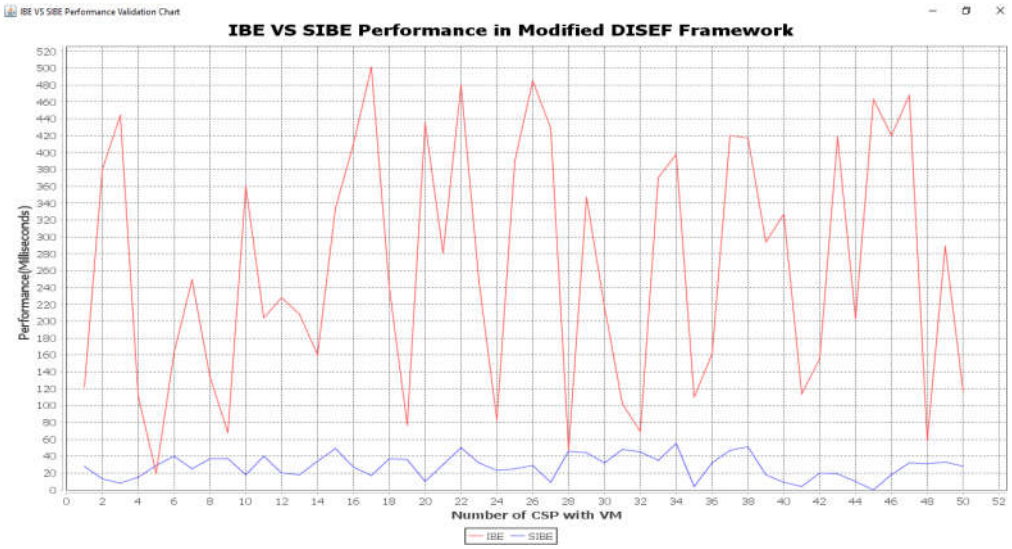


Figure 11. Performance Validation Graph

From the experimental result, it is observed that the performance degradation of the SET/GET operations in the DSSE system are better compared with the existing systems. The key–value size is varied from 1 KB to 1024 KB to show the different performance impact. Fig. 9(a) shows the performance of the SET operations with different key–value sizes under the three systems, EncKV, Disef, and DSSE. From the experimental result, it can be observed that the average latencies of the SET operations in all three systems increase as the sizes of key–value pairs increase. The reason is that the network transmission time increases due to the data size increases. As Fig. 9(b) shows, the trend is similar in the GET operations. For the largest size of key–value pair, the latency gap between the Disefand the DSSE are better considering the security guarantees provided in the DSSE system. Figure 10 shows the throughput versus latency of three key–value store systems. The average latencies of the two systems (Base KV and DSSE) for a given throughput are small and close to each other until the throughput reaches 150,000 operations per second (shown in Fig. 6). After this point, the latencies of the two systems increase significantly due to the network bottleneck. The latency of DSSE increases faster as the encryption and enclave transition overheads also increase.

Performance Validation Graph: The below graph shows the validation of the proposed and existing systems which mainly shows the performance of the two techniques. In this study, the existing technique considered for the analysis is the Disef framework (Cheng, 2018) which enabled efficient distributed analysis of network dataset while achieved security guarantees of data confidentiality and integrity. In-memory dataset was protected by using a new encrypted key–value format and could be efficiently transferred into Intel SGX enabled enclaves for shielded execution. The authors used both synthetic and real-life benchmarks to test the performance of key–value store systems. The synthetic key–value requests are manually generated operation commands. Figure 11shows the validation results of the IBE and SIBE performances in modified Disef framework. The X-axis denotes the number of CSP with Virtual machines and Y-axis represents the performance of the existing and proposed methodologies. From the below graph it is proved that the proposed system is having a better performance as compared to the existing technique i.e., IBE.

Conclusion

A new Distributed Security Shielded Execution is proposed for communicable cyber threats analysis using the SIBE technique. Security sensitive operations are provided by implementing in the Disef enclave. The potential Iago attacks in Disef system has been minimized by introducing a Novel SIBE unbreakable new key–value encrypted format. This is in turn integrated with the version number and update counter to prevent possible rollback and replay attacks. Also a secured Disef system in high availability with multi-cloud support has been depicted. The experimental results of the proposed framework proved to support secure cost analysis and response time analysis of large network dataset and has showed comparable performance with existing systems.

REFERENCES

Bellare, M. and Rogaway, P. 1993. Random oracles are practical: A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM conference on Computer and communications security* (pp. 62-73). ACM.

Bellare, M., Boldyreva, A. and Palacio, A. 2004. An uninstantiable random-oracle-model scheme for a hybrid-encryption problem. In *International Conference on the Theory and Applications of Cryptographic Techniques* (pp. 171-188). Springer, Berlin, Heidelberg.

Boneh, D. and Franklin, M. 2001. Identity-based encryption from the Weil pairing. In *Annual international cryptology conference* (pp. 213-229). Springer, Berlin, Heidelberg.

Boneh, D. and Franklin, M. 2001. Identity-based encryption from the Weil pairing. In *Annual international cryptology conference* (pp. 213-229). Springer, Berlin, Heidelberg.

Boneh, D. and Franklin, M. 2003. Identity-based encryption from the Weil pairing. *SIAM journal on computing*, *32*(3), 586-615.

Boneh, D. and Boyen, X. 2004. Secure identity based encryption without random oracles. In *Annual International Cryptology Conference* (pp. 443-459). Springer, Berlin, Heidelberg.

Boyen, X. 2003. Multipurpose identity-based signcryption. In *Annual International Cryptology Conference* (pp. 383-399). Springer, Berlin, Heidelberg.

Canetti, R., Goldreich, O., &Halevi, S. 1998. The random oracle model, revisted. In *30th Annual ACM Symposium on Theory of Computing* (Vol. 45).

Canetti, R., Halevi, S. and Katz, J. 2003. A forward-secure public-key encryption scheme. In *International Conference on the Theory and Applications of Cryptographic Techniques* (pp. 255-271). Springer, Berlin, Heidelberg.

Cheng, Y., Wu, Q., Chen, W. and Wang, B. 2018. Distributed shielded execution for transmissible cyber threats analysis. *Journal of Parallel and Distributed Computing*, *122*, 70-80.

Choon, J. C., & Cheon, J. H. 2003. An identity-based signature from gap Diffie-Hellman groups. In *International workshop on public key cryptography* (pp. 18-30). Springer, Berlin, Heidelberg.

Cocks, C. 2001. An identity based encryption scheme based on quadratic residues. In *IMA International Conference on Cryptography and Coding* (pp. 360-363). Springer, Berlin, Heidelberg.

Cocks, C. 2001. An identity based encryption scheme based on quadratic residues. In *IMA International Conference on Cryptography and Coding* (pp. 360-363). Springer, Berlin, Heidelberg.

DíazLópez, D., Blanco Uribe, M., Santiago Cely, C., Vega Torres, A., Moreno Guataquira, N., Morón Castro, S. and Gómez Mármol, F. 2018. Shielding IoT against Cyber-Attacks: An Event-Based Approach Using SIEM. *Wireless Communications and Mobile Computing*, *2018*.

Ezhei, M. and TorkLadani, B. 2018. Interdependency analysis in security investment against strategic attacks. *Information Systems Frontiers*, 1-15.

Gentry, C. and Silverberg, A. 2002. Hierarchical ID-based cryptography. In *International Conference on the Theory and Application of Cryptology and Information Security*(pp. 548-566). Springer, Berlin, Heidelberg.

Heng, S. H. and Kurosawa, K. 2004. k-Resilient identity-based encryption in the standard model. In *Cryptographers' Track at the RSA Conference* (pp. 67-80). Springer, Berlin, Heidelberg.

Liang, G., Weller, S. R., Luo, F., Zhao, J. and Dong, Z. Y. (2018). Distributed blockchain-based data protection framework for modern power systems against cyber-attacks. *IEEE Transactions on Smart Grid*.

Maurer, U. M. and Yacobi, Y. 1996. A non-interactive public-key distribution system. *Designs, Codes and Cryptography*, *9*(3), 305-316.

Ogishi, K., Sakai, R., &Kasahara, M. 2007. *U.S. Patent No. 7,239,701*. Washington, DC: U.S. Patent and Trademark Office.

Ryuichi Sakai, Kiyoshi Ohgishi, and Masao Kasahara. Cryptosystems based on pairings. In Symposium on Cryptography and Information Security— SCIS 2000, Japan, 2000.

Sauerwein, C., Sillaber, C., &Breu, R. 2018. Shadow cyber threat intelligence and its use in information security and risk management processes. *Multikonferenz Wirtschaftsinformatik (MKWI 2018)*.

Shamir, A. 1984. Identity-based cryptosystems and signature schemes. In *Workshop on the theory and application of cryptographic techniques* (pp. 47-53). Springer, Berlin, Heidelberg.

Tanaka, H. 1987. A realization scheme for the identity-based cryptosystem. In *Conference on the Theory and Application of Cryptographic Techniques* (pp. 340-349). Springer, Berlin, Heidelberg.

Tsujii, S. and Itoh, T. 1989. An ID-based cryptosystem based on the discrete logarithm problem. *IEEE Journal on Selected Areas in Communications*, *7*(4), 467-473.

urRehman, H., Yafi, E., Nazir, M. and Mustafa, K. 2018. Security Assurance Against Cybercrime Ransomware. In *International Conference on Intelligent Computing & Optimization* (pp. 21-34). Springer, Cham.

****