Research Article

Implementation of Block-Level Double Encryption Based on Machine Learning Techniques for Attack Detection and Prevention

S. Gnanavel, K. E. Narayana, K. Jayashree, P. Nancy, and Dawit Mamiru Teressa

1Department of Computing Technologies, SRM Institute of Science and Technology-Kattankulathur, Chengalpattu (District), Tamil Nadu 603203, India
2Department of Computer Science and Engineering, Rajalakshmi Engineering College, Anna University, Chennai, India
3Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, (Deemed to Be University), Chennai, India
4Department of Chemical Engineering, College of Biological and Chemical Engineering, Addis Ababa Science and Technology University, Addis Ababa, Ethiopia

Correspondence should be addressed to S. Gnanavel; gnanaves1@srmist.edu.in and Dawit Mamiru Teressa; dawit.mamiru@aastustudent.edu.et

Received 10 March 2022; Revised 19 June 2022; Accepted 22 June 2022; Published 9 July 2022

Academic Editor: Mohammad Farukh Hashmi

Copyright © 2022 S. Gnanavel et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Cloud computing is one of the most important business models of modern information technology. It provides a minimum of various services to the user interaction and low cost (hardware and software). Cloud services are based on the newline

1. Introduction

Information technology (IT) structure is one of the most popular researches in the current new product line environment of cloud computing hosting the various software services running the Internet. The aim is to facilitate data migration and infrastructure applications on new product

lines from virtual access through the Internet. Cloud services are based on the virtualization of the entire structure with multilease multiple virtual machines (VMs) for better resource management and robust isolation. However, cloud security attacks and takes advantage of many tenants. These files do not pose a major security threat to cloud data centers and cryptographic systems. Side-channel attacks are easier
to implement because they take advantage of the weaknesses of cryptography and avoid leaving the fingerprints of the attacks. It can be described as the most successful data breach attack. Some existing cross VM is called cache-based VM side-channel attacks that use different levels of leak-proof of sensitive data in the CPU cache via horizontal VMs. Cross-VM side-channel attack is easy to implement because it exploits the weaknesses of the encryption system and avoids leaving fingerprints to attack shown in Figure 1.

Cross-VM attacks the cryptosystem using a time-side flow path as a major parameter such as CPU and cache memory. Cache memory is located between RAM and CPU cores to remove the delay added by accessing the data. The main objective of the cache memory is to decrease the required time for accessing data from the main memory. The cache is divided into L1, L2, and L3 levels. Each VM has its L1 and L2 cache, whereas L3 is a sharable cache. Although cross-VM side-channel (SC) attacks existed in the past in multilevel systems, including database, operating system, and networking, the cloud computing (CC) co-residency feature makes cross-VM SC attacks more effectively in this paradigm. It was not easy to gain physical access to the system in the past, but with shared resources, physical access can be easily accomplished in the cloud. They extract the full encryption key of the well-known cryptographic algorithms, including RSA and AES without any direct or physical interaction with the cryptographic devices.

The object of the proposed work implements the hybrid encryption algorithm to encrypt the data and adds more blocks to the ciphertext. Secure Block-Level Double Encryption (SBLDE) algorithm for user signature verification in the cloud server. Identity-based linear classification (IBLC) method is used for classifying the attacker channel when the data is transferred/retrieved from the VM cloud server.

This paper discusses different aspects of the cloud computing-related cross-VM side-channel attacks. Several solutions for breaks and general side-channel attacks are resolved. A technology used in the new production line to prevent cross-VM side-channel attacks is cryptographic algorithms. The second part of this paper describes different algorithms for mitigating cross-VM side-channel attacks. It can provide timing information, electromagnetic leakage power consumption, or additional information for which sound is available.

2. Related Work

This section could be found in the literature addressing the problem of cross-VM side-channel attacks. The authors in [1] discussed the sliding window side-channel attacks (SW-SCAs) for acquisition device could be applied encryption device. The attack situations independent of the source for encryption require a trigger signal. The cross-VM side-channel attack (SCA) technique estimates and statistically correlates between measured power supply current traces to extract the key.

This paper [2] presents ASNI: Attenuated Signature Noise Injection with AES-128 encryption as general measures for lower efficiency and display SCA properties. This enterprise integration [3] technologies include new major updates and technologies hidden at regular intervals. It is driven by a maximum of side-by-side leak traces required for a trace VM side-channel attack.

The authors in [4] discussed the effective convolutional neural network (CNN) method to reduce large datasets. The optimal number of convolution blocks is used to extract strong features within the established cost range. Machine learning-based side-channel attack (SCA) for noise reduction model was proposed in [5]. It uses hardware performance counters (high-performance computers) to collect behavioral data for parallel processes while running. This SCA uses several machine learning models to diagnose, which gives a very low performance. The cross-VM attack classification system used in [6–8] allows us to analyze side-channel attacks and systematically promote new countermeasures.

The paper [9] can present a new hidden cross-VM channel attack with high bandwidth and reliable data transmission in the cloud. The first virtual environment utilizes existing cache channel technology, thus having obvious key drawbacks and difficulties. The instantiation GPU model and AES implementation can be used for performance evaluation [10]. The leaking cross-VM side-channel attacks from single instruction and multiple thread (SIMT) systems are also applicable to any GPU leak model that specifically supports CUDA.

Determine to formally remap all protected memory to design a sequential classification selection algorithm proposed in [11]. This working core attack in a cross-VM [12] configuration showed a real threat to the system successfully in real life and recovered all encryption keys in a short amount of time. A malicious user created a cross-side-channel in [13] to retrieve confidential information from virtual machines colocated on the same server that is focusing on such threats and coresident attacks. Users want to apply semisupervised learning techniques to the classified attacker or malicious users. It is modeled as the game theory security of two players in [14, 15]; it has analyzed our attack
The paper [16] discussed a new computer architecture that enables secure data processing and a complete framework for common cryptographic calculations without the present shared key. A new architecture is proposed here to combine benign traffic and classified network traffic such as DDoS attacks. The authors proposed a well-stacked sparse autoencoder (AE) as a feature of the deep neural network (DNN) method in [17].

The Reliable Trusted Computer System (RTCM) in [18] is based on multisource feedback and fog computing VM fusion algorithm to solve this cross-VM attack problem. It is based on the emergence of another cloud deployment model. The papers [19, 20] present an overview of the state of the present art cloud architecture and the classification of potential countries. The most complex system of instruction increases the execution time and security issues. Many methods are being researched for monitoring client virtual machines. Some common nonintrusive methods of technical monitoring of guest VMs are observation of computational indicators [21, 22], observation of system calls [23–26], and virtual machine introspection [27, 28]. The data security and the isolation assurances are the major threats in the cloud computing; thus, our proposed methods provide the linear-based identity classification approaches; it helps to detect from the VM attacks in the networking cloud computing [29].

To solve the above problems and develop a plan to improve the cross-VM attack detection performance in cloud system, one of the most sophisticated attacks is the cross-VM cache side-channel attack that exploits shared cache memory between VMs. The physical coexistence feature allows an attacker to use another virtual machine on the same physical machine for inadequate logical isolation of sensitive information leaks through communication. It protects information after it has been decrypted by an unauthorized person, unlike encryption. The purpose of cross-VM side-channel attacks is to use this encryption system, and hidden communication provides less efficiency in a cloud server.

### 3. Implementation of the Proposed Method

Implementing the proposed Secure Block-Level Double Encryption (SBLDE) method generates the signature keys to provide cross-VM side-channel detection on the cloud. The signatures are also kept in the signature database for reference, and they are also sent to the cloud server running these virtual machines [30]. Then, each cloud VM is monitored using an identity-based linear authentication, and the attacker monitor instructs the performance counter to monitor entire virtual machines (VMs) concurrently.

The attack monitor phase verifies the user request authorized or irrelevant on a cloud virtual machine from cloud users. The cryptographic signature algorithm verifies the request when such a false request is received. Identity-based linear classification is used to predict the cache memory and VM channel based on a given feature value [31]. This classification is used for predicting the probabilities of the various classes, does an analysis, and gives a group of independent variables. It makes use of a linear equation with independent predictors for predicting a value to train the classifier using benchmark data set and estimate the feature attack probability in the cloud server. It provides secure service access with the help of Cloud Service Verification, which verifies user access data. The above Figure 2 presents the block diagram of cloud authentication and cross-VM side-channel attack detection.

#### 3.1. Secure Block-Level Double Encryption (SBLDE)

The Secure Block-Level Double Encryption method encrypts the cloud data and verifies the user certificate keys to allow authorized users. The key authentication method to identify
the cross-VM side-channel attack is by effectively virtualizing cloud data center and providing the better future authentication result. The SBLDE is a public key with authentication for configuring regular DES and AES algorithms [32]. It has improved the high security of doubled-time accept inputs such as 64-bit key size with 512-bit block size and matrix size. To analyze the cache memory data and encrypt using this method, SBLDE algorithm performs the 8-round operations divided into two parts. It is invoked round key and shift row with the maximum column. In this, each message block is shifting the maximum column and encrypts with the key. The round key stores on the cloud server signature database to authenticate the original request user [33, 34].

This method uses double encryption to secure the data, and it is divided into three levels. First, the hybrid encryption algorithm (DES and AES with block-level) encrypts the data and adds more blocks to the ciphertext. Finally, it encapsulates the cipher data with the key to transmit to the cloud.

3.2 Cloud User Service Verification. The cloud service verifies the data through verifiable outsourcing auditing policy to manage integrity. The data request is verified by accessing the key log provided by the owner submitted to the cloud service provider. The cloud controller enriches the verification for stored logs. To provide the right authentication whether the key is validated to access the rights, the key is verified after the file is allowed to decrypt the data.

The above algorithm varies the access point key validation from the right access person to provide the data. The data owner is given the access rights to access the data to the responder. Cloud service provider integrates the verifiable access security. Finally, the authentication verifies key validation and then provides permission to decrypt the data.

**Algorithm 1: Steps.**

Input: plain text or file \((I_p)\), round key \((Rk)\), data modification \((Dm)\), key \((k)\).

Step 1: the data split into blocks \((d_b)\)

For \(x\) in read \(I_p\) then

If \(x < 32\) then

\(d_b \leftarrow x\); Return next block.

Else return to step 1

End

Step 2: Each block data DB to generate a round key \((Rk)\). In block \((I_b)\), left block \((L_b)\) and right block \((R_b)\).

To initialize the permutation table \((d_b)\)

\(Rk[\] \leftarrow\) set all sub key \((d_b(k))\)

For each round \(\leftarrow\) 0 to 8 do

Data out \([\] \leftarrow d_b[round] [data]

If \(d_b[round] < 8\) then

[Data] \leftarrow Dm \((d_b[round] [data out])

Data out \([\] \leftarrow\) mixer \((I_b, L_b, R_b)\)

End if

End for

d_b \leftarrow Complain \((Data\ out [\].64,I_o, L_o, R_o)\) return \(d_b;\)

Step 3: To encrypt each block. Cipher block \([64]\), round key \([8, 16]\).

While \((d_b < n)\)

Temp = \(d_b[i - 1]\)

If I mod n=0, then

Temp = sub words (temp) XOR \((d_b[i])\)

Else if n>8 and I mode n =4

\(d_b[i] = d_b[i - n]\) XOR temp

End

Encrypt DES and AES \([d_b]\)

Generate a public key \([d_p]\), private key \([d_s]\)

End while

To encapsulate the encrypted data with a round key

Complain \((CTD) \leftarrow (d_p, round key, public key, private key)\)

Step 4: To verify the user key log and decrypt the original file

Cloud authentication request key \((rk)\)

If \(rk\) equal to user key, then

To decrypt the original message \((M_o)\) for ciphertext

\(M_o = CT_{d(i)} \cdot e \mod \phi(n) . CT_{d(i+1)} \cdot \mod \phi(n) \cdot \mod (n)\)

End if
3.3. Identity-Based Linear Classification. The identity-based linear classification method analyzed the cloud user information and compared it with trained data. This VM monitoring method identifies each server’s user information based on the MAC address and cache memory. The linear method is used to learn each port status, memory and data transmit range, and user information for identifying the attacker channel to train the classifier from the existing side-channel attack dataset (KNU practicum). All the test input parameters (cache memory, key, socket, port, CPU core, etc.) are evaluated based on trained data to update the weightage of each data and compare it to the threshold value. 

In this input, weight functions take $W_{px} = 0$. Thus, by taking the threshold as 0, perceptron classifies data based on which side of the plane the new point lies. The VM channel input $x_n$ value is taken to train each bias and estimate the each input weight. This weight sum value is compared to the threshold value; the flow is present in Figure 3.

This value takes a weighted linear combination of input features. It passes it through a thresholding function that outputs 1 or 0 (if 0 means no attack detection and 1 means port under attack) to change the communication channel.

4. Result and Discussion

This section discussed the proposed method simulation result and compared it with previous sliding window side-channel attacks (SW-SCAs), MemWander, and Ciphertext-Policy Attribute-Based Encryption (CPABE) methods. The performance metrics response time, attack detection rate, throughput analysis, and false assurance estimate the proposed SBLDE method performance. This simulation is developed in a visual studio framework with c# language using the KNU practicum cloud attack data set to analyze attackers.

<table>
<thead>
<tr>
<th>Simulation parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tool</td>
<td>Visual studio</td>
</tr>
<tr>
<td>Cloud type</td>
<td>Public cloud (amazon)</td>
</tr>
<tr>
<td>Simulation time</td>
<td>10 min</td>
</tr>
<tr>
<td>Number of users</td>
<td>150</td>
</tr>
<tr>
<td>Dataset name</td>
<td>KNU practicum</td>
</tr>
<tr>
<td>File size</td>
<td>10 to 30 MB</td>
</tr>
</tbody>
</table>

Table 1 shows the proposed Secure Block-Level Double Encryption (SBLDE) method simulation parameter used to analyze the performance.

$$T_{\text{response}} = \frac{n}{r} - T_{\text{tic}},$$

where $n$ is the number of concurrent users, $r$ is the number of requests per second the server receives, and $T_{\text{tic}}$ is the average think (execution) time (in seconds).
The proposed system SBLDE produces a 2.4 sec of low average response time for cloud users, and the existing method SW-SCA has an 11 sec complicated response time of execution than others. Figure 4 above shows the comparison complexity time that the proposed SBLDE method produces more improvement than other SW-SCA, Memwander, and CPABE methods.

Analysis of the proposed SBLDE provides 91.5% of the cloud cross-VM side-channel attack prediction. This proposed method considers all the VM channel features (cache...
memory statues, CPU core, socket, port status, response time, and keys). It provides a higher detection rate than the existing method. SW-SCA, MemWander, and CPABE are present in Figure 5.

Throughput = packetsize * recv.packet * 8.0 / 1000. \hspace{1cm} (2)

Figure 6 shows the test results for the performance analysis comparison of the current method SW-SCA, MemWander, and CPABE with the proposed SBLDE. This analysis of SBLDE has provided an 1102 bps of higher throughput ratio for 100 sec compared to the existing method.

The false occurrence analysis is based on incorrectly classifying the attack in cloud VM, and the result is taken from different file sizes to prove the proposed method performance. It shows that the implementation of the SBLDE method has produced an active redundant false rate than previous SW-SCA, MemWander, and CPABE. The SBLDE provides a 9.2% low false rate than existing methods; the comparison is shown in Figure 7.

Security analyses are evaluated to effectively provide cloud users with access to data and documentation depending on the security key encryption process and cloud controller for overall execution.

Figure 8 shows the different methods that produce the different level of user to do the security activity. The current system provides security for SW-SCA, MemWander, and CPABE as 88.8%, 87.5%, and 89.8% for 150 users, respectively. The SBLDE produces 92.8% of 150 users’ impact safety performance compared to other different methods.

5. Conclusion

Cross-VM side-channel attacks are passive and noninvasive, which are challenging to protect your system from this attack. The criteria proposed in this work can prevent attacks from interrupting the system and serve as an effective means for cross-VM side-channel attacks. This proposed method protects the cloud data and prevents cross-VM channel attack detection efficiently, compared to other existing methods. The Secure Block-Level Double Encryption (SBLDE) with a linear classification algorithm is developed to identify the cross-VM side-channel attack and secure access cloud server. In this SBLDE, verify the cloud user signature for the cloud signature database and verify the cache memory key. It verifies that only the data is encrypted and
decrypted. The analysis of the experiment and the performance of the proposed framework are discussed. The proposed method provides a 93.7% of average secure, 2.4 sec of response time, and 91.3% of attack detection on VM cloud server. In future enhancement, this work can enhance the security with efficient tracking of data for communication purpose.

**Data Availability**

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

**Conflicts of Interest**

There is no conflict of interest.

**References**


