We are IntechOpen, the world’s leading publisher of Open Access books
Built by scientists, for scientists

6,600 Open access books available
177,000 International authors and editors
195M Downloads

154 Countries delivered to
TOP 1% Most cited scientists
12.2% Contributors from top 500 universities

WEB OF SCIENCE™
Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com
Chapter

Secured Storage Mechanism for the Cloud-Based Banking Applications Using ECC

Gopinath V.

Abstract

The security of data becomes more vulnerable despite the various safety measures. The significant contribution of the chapter is to describe the design of a secured Cloud storage mechanism using ECC. Many banks are using cloud technology. However, it is not free from hacks and threats. Here comes the focus on Cloud Security and a lot of research work is happening around it. ECC-based secured cloud framework with a logging mechanism provides a high-level design of the SSL VPN and secured cloud mobile banking. It helps to secure data transfer for users. Software applications and web services are handled by maintaining the log files securely, and this security scheme shields the aligning of different kinds of data elements to each provider using an ECC algorithm. It is implementing a two-layered security system, which includes both private and public cloud storage environments, also providing a customized secure logging mechanism that will be encrypted by ECC. The TRA (banking) customers will get lots of benefits from this cloud storage mechanism. The proposed design acts like a buffer for end users, Internet service providers, and Banks, so it is more effective and secure in the Cloud environment.

Keywords: ECC, SSL VPN, XML, web services, cloud computing, logging, cloud computing, banking, security, transaction rich applications (TRA)

1. Introduction

In this modern world, secured communication is a significant need to perform high-end data transfers. In banking, the secure cloud storage mechanism is gaining importance and is yet to become omnipresent. The concern with the banking storage mechanism is that it requires immediate attention, such as rewards, precautions, confidentiality, perspective, usability, data management, unpredictable growth in the volume of transactions, and price reduction. Cloud computing provides everything as a Service (XaaS) to the banking sector in the form of Software-as-Service (LaaS) Which SaaS supports customer relationship management, accounting, invoicing, and Enterprise resource planning. PaaS is a suitable platform for applications. It helps in reducing the cost of IT Infrastructure & the spending on hardware devices & software applications can be cut down significantly. With LaaS businesses can procure those assets as a completely redistributed service. The same approach is
applied for secured usage of a cloud storage mechanism with XML Web services and a logging mechanism to provide a high-level design of the SSL VPN to track the user activity and identify in case of any vulnerable security issues. In addition, it is implemented for a safe storage mechanism using ECC in transaction-rich applications (TRA). Many TRA makes use of storage mechanism 4 due to the service-oriented architecture, seamless accessibility, and other advantages of this advent technology. At the same time, it is vulnerable to hacks and threats. Hence securing this environment is of utmost importance, and many research works are being reported focusing on it. Cloud service provider gives base security, but it is not sufficient to handle the financial data in the cloud environment. Both the banking industry and the customer look for more protection when it comes to funding transactions like payroll, CRM, accounting, invoicing, Enterprise resource planning, etc. Security Breaches are not tolerable in the banking sector. In the current environment of cloud computing, TRA has got only a single layer of security. So, the mechanism to provide high-level security with cost-saving, high performance, and bandwidth is needed. The proposed system is aimed at providing a high-level design of the SSLVPN with ECC, which is applied to a private cloud with a secure logging mechanism. The model has experimented with java coding and open VPN with the MS Azure cloud system. SOAP UI tool has been used for performance analysis, which is deployed into the cloud web server [1, 2].

The design applied to a private cloud environment, in addition to providing a customized secure logging mechanism that will be encrypted by ECC, ensuring data security and privacy protection of entire user activities in the cloud. To launch the ECC-based SSL VPN, the private cloud VPN frameworks should be added to the system. The same approach is applied in the cloud storage mechanism for the TRA (banking) customers can get lots of benefits [3, 4].

- Utilization of Time: Customers can use it 24*7 hours.
- Increase Adaptability: It helps banks to enjoy the promotion of adaptability ratios and operating leverage [5, 6].
- Decrease Invest Amount: The banks are not ready to invest a large amount to purchase software, hardware, and related workforce for the usage of cloud computing.
- Security Comparison: ECC-based secure cloud storage mechanism for TRA is more sheltered than online and internet banking.

It provides an additional layer of security with ECC. While the banking customer is connecting to the cloud, it connects the P2P network with a second layer of security SSL and ECC [7], and the banking application utilizes the same set of ECC digital keys.

2. Design of Secured Cloud Storage Mechanism

ECC-based secure cloud framework with logging mechanism is to provide a high-level design of the SSL VPN and secure cloud mobile banking. It helps secure data transfer for the users [8–10]. Software applications and web services are used with secure log files maintained. **Figure 1** shows the high-level design of the Cloud storage mechanism.
3. Design of Secured Cloud Storage Mechanism

The architecture of a secure cloud TRA is explained in Figure 2. Movable equipment and the remote systems are connected to the mobile machinist and wireless access point. These are connected to either an entry point or satellite via the base station. The mobile user’s request and response are delivered to cloud banking through the presently available SSL VPN with an additional layer of protection from ECC.

Central processors that are linked to the mobile network receive the mobile users’ requests as well as send details such as user IDs and user location to banking database storage. The projected security theory enhances the level of protection that currently supports mobile banking and facilitates the realization of P2P networks.

The following are the objectives of the proposed framework:

- Assuring information security and privacy protection of entire mobile/remote user’s activities in the cloud. ECC-based SSL VPN.
• ECC encryption helps to provide less bandwidth, computing power, and memory for creating customized security for data packets.

3.1 Storage mechanism for transaction rich application

The cloud storage mechanisms are exclusively designed for cloud-based supply. Like how the physical server can release virtual server images, similarly, the device’s instances can be made virtual. They can provide fixed-increment capacity allocation in support of the pay-per-use mechanism. Via cloud storage devices, stored data can be exposed with the help of remote access. It gives general logical elements of data storage, such as:

• The Group of data that are stored in the folders are called files.
• The lowest level of storage is called blocks.
• The collections of data are called datasets.
• Objects are called metadata.

There is a huge risk of data loss, it is essential to recover the data when there is an issue and failure. However, customers are ensured of data availability post such issues. So, it is essential to provide data security in the cloud. The proposed design provides a secure cloud storage mechanism for TRA.

Figure 3 [1] shows the data request of the user via the internet to access the private cloud data servers and applications to utilize the Bank’s web server and a firewall. The design is trialed with java coding 43 and inbuilt ECC algorithm into the webserver. Logs are maintained in the DB tables as well as the web server, and the user accesses the web server system with the help of SOAP UI. The user hits directly the bank’s web
3.2 Cloud-based ATMs architecture using ECC

An ATM is used after installing the package, and it will be ready for user transactions. Usually, the ATM package consists of an operating system, machine third-party software and hardware packages, and core banking software; normally, this package deployment takes more than 4 hours to get completed. So, each machine installation takes 8 hours daily to be available to users. Banks are investing a huge amount in this activity. For increasing transactions, many banks are extending their activity by using ATMs. ATMs have become more user-friendly nowadays, which is a must need for the customer to use on a day-to-day basis; this promotes establishing banking. As there is a need for creating more ATMs to provide easy access to customers, banks need to invest more in setting up these machines, which can now be reduced by using the cloud. However, security becomes a concern worry about using this installation in a cloud environment. Here, a private cloud is used to install/deploy the package in a single trigger; within 4 hours entire installation is performed for more than 1000 ATMs; also, it provides the double-level security to connect the Host server and the core bank. Below is Figure 4. Shows the ATM package deployment with the help of a secure cloud environment.

Here, the package is deployed in the private cloud host server itself. Also, it is possible to increment the ATM node by more than 1000 in a single installation time, so this method is more cost effective and also provides high-level security for the customer data.

Figure 5 discusses the ATM package and machine architecture using a secure cloud environment. The core banking server interacts with the Cloud hosting server,
which has the ATM software packages, which are customized bank screens, ATM software, other 3rd party machine software, and operating system.

ATM packages have two-layer securities like SSL and ECC which gave high-level security of the bank’s data and package while the installation got initiated.

The proposed design will give high standard security and more cost savings to the banks. Therefore, they can meet their customer demands.
4. Performance analysis for the cloud-based TRA

Cloud computing is characterized by accessibility at whichever instance is desired, with the reason for the reduction in costs and increase in operational efficiency. The ECC VPN that connects to cloud computing should possess the above-mentioned characteristics. The network atmosphere should be trouble-free to set up, modernize, or erase connections based on the demand for cloud computing, to ease the complex system management.

The below Figure 6 shows the most vulnerable cyber-attack on the bank in the cloud environment; in this measurement, a high percentage of cybercriminals risk occurs in web applications and web services in the cloud environment. Many banks have identified and reported 89% of breaches in the last few years. Many a time, SQL injection and malware have accessed sensitive information in Cloud storage mechanisms like DB, web applications, and web services. It is hard to find web application attacks because banks have millions of users accessing their sites at the same point in time. So, spotting cybercriminals is a difficult task. The second highest-rated risk is the DoS attack; it completely 58 hacks down a system, and shuts down a machine/network, making it inaccessible to its intended users. The rest of the other attacks is very minimal to the overall bank attacks in the cloud environment.

In Table 1, most of the efforts in the below-mentioned checklist talk about the number of vulnerabilities and threats focused on Cloud Computing. It describes the analysis report that is related to the analysis, and test used in cloud environments, and it specifies that the cloud service representations are open to these vulnerabilities and threats. More prominence on threats has been applied that are allied with statistics being stocked up and practiced remotely, sharing resources, and the usage of virtualization [3, 4].

This section presents the result and performance analysis for the proposed secure cloud storage mechanism for TRA. The design has experimented with java coding and an inbuilt ECC algorithm, which is deployed in a cloud environment. SOAP UI tool has been applied for performance analysis; it is deployed into the cloud web

![Cloud mobile bank Attacks](image)

Figure 6. Vulnerable analysis report to cyber-attack for the bank.
<table>
<thead>
<tr>
<th>S. No</th>
<th>Vulnerability &amp; threats analysis</th>
<th>Existing method (RSA)</th>
<th>Proposed method (ECC+ SSL)</th>
<th>Analysis report</th>
<th>Existing frameworks</th>
<th>Proposed framework</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Secure even after the loss of user identity and password</td>
<td>✓</td>
<td>✓</td>
<td>Configuration Management</td>
<td>Single</td>
<td>Single</td>
</tr>
<tr>
<td>2</td>
<td>Indexing of data, Keyword Search</td>
<td>✓</td>
<td>✓</td>
<td>Fault Management</td>
<td>Simple</td>
<td>Simple</td>
</tr>
<tr>
<td>3</td>
<td>Insecure interfaces and APIs, Account or service hijacking</td>
<td>✓</td>
<td>✓</td>
<td>Performance Management</td>
<td>Integrated</td>
<td>Integrated</td>
</tr>
<tr>
<td>4</td>
<td>Unlimited allocation of resources, Denial of Service</td>
<td>✓</td>
<td>✓</td>
<td>Security</td>
<td>Support</td>
<td>Support</td>
</tr>
<tr>
<td>5</td>
<td>Data-related vulnerabilities, Data leakage</td>
<td>✓</td>
<td>✓</td>
<td>Management</td>
<td>directory server</td>
<td>encrypted data and directory server</td>
</tr>
<tr>
<td>6</td>
<td>Vulnerabilities in Virtual Machines, VM hopping</td>
<td>✓</td>
<td>✓</td>
<td>Accounting Management</td>
<td>Convenient</td>
<td>Convenient</td>
</tr>
<tr>
<td>7</td>
<td>Vulnerabilities in Virtual Machine Images, Data leakage</td>
<td>✓</td>
<td>✓</td>
<td>Confidentiality</td>
<td>File encryption not provided</td>
<td>Symmetric key</td>
</tr>
<tr>
<td>8</td>
<td>Vulnerabilities in Hypervisors, VM escape</td>
<td>✓</td>
<td>✓</td>
<td>Authentication</td>
<td>Only base-level security is provided.</td>
<td>Password-based advanced-level security is provided.</td>
</tr>
<tr>
<td>9</td>
<td>Vulnerabilities in Virtual Networks, VM escape, Customer-data manipulation</td>
<td>✓</td>
<td>✓</td>
<td>Access Control</td>
<td>Exposure to the normal area</td>
<td>Encryption of security area information</td>
</tr>
<tr>
<td>10</td>
<td>Log Security in Virtual Machine, Data Packets</td>
<td>✓</td>
<td>✓</td>
<td>Impersonation Attack</td>
<td>No impersonation</td>
<td>Two stages of user authentication</td>
</tr>
</tbody>
</table>

Table 1. Vulnerability & Threats Analysis Report RSA vs. ECC.
server. With the help of this, the performance can be checked with load balance. Single usage of customized SOAP UI can trigger 1000 users at a time in the cloud storage server. Logs are stored in the database table as well as in the server and client system based on the load balance testing and performance tracking below Table 2. Shows the various comparisons of the proposed and existing systems in the Cloud storage mechanism [1].

Below Table 3 ECC and RSA key size strength ratios are measured in the table; for example, the RSA system requires 7680 bits to provide security, whereas ECC requires only a 384-bit key to acquire the security and its key ratio stands at 1:20. So ECC provides greater efficiency in terms of key size and bandwidth; it means higher speed and lower power consumption [1].

Figure 7 discusses the key length comparison of the proposed ECC algorithm vs. the existing RSA algorithm [1]. The public key operation of ECC-160 is only 3.69 milliseconds, and it is 50% comparatively lesser than RSA-1024 and other keys. The flowchart given below explains the key generation time and required memory size for both ECC key and 1024-bit RSA key, 160-bit ECC is much better than RSA [13]. The protection measures for both 160-bit ECC key and 1024-bit RSA key are similar. Hence breaking

<table>
<thead>
<tr>
<th>S. No</th>
<th>Cloud analysis report</th>
<th>Proposed system (ECC + SSL)</th>
<th>Existing system (RSA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Upfront Investment</td>
<td>Low investment</td>
<td>High investment</td>
</tr>
<tr>
<td>2</td>
<td>Additional Hardware/IT costs</td>
<td>Not required</td>
<td>Yes, Required</td>
</tr>
<tr>
<td>3</td>
<td>All-Time costs</td>
<td>Predictable cost</td>
<td>Unpredictable cost (but maybe lower)</td>
</tr>
<tr>
<td>4</td>
<td>Degree of customizations</td>
<td>Less customizable in general</td>
<td>Greater ability to customize</td>
</tr>
<tr>
<td>5</td>
<td>Control of data security standards</td>
<td>Customers can control the data</td>
<td>Banks can control the data</td>
</tr>
<tr>
<td>6</td>
<td>Confidentiality</td>
<td>Symmetric key</td>
<td>File encryption not provided</td>
</tr>
<tr>
<td>7</td>
<td>Authentication</td>
<td>Password-based advanced-level security is provided.</td>
<td>Only base-level security is provided.</td>
</tr>
<tr>
<td>8</td>
<td>Access Control</td>
<td>Encryption of security area information</td>
<td>Exposure to the normal area</td>
</tr>
<tr>
<td>9</td>
<td>Log Security in Virtual Machine, Data Packets</td>
<td>Provided and It is Secured</td>
<td>Not provided</td>
</tr>
<tr>
<td>10</td>
<td>Computational Overheads</td>
<td>Roughly 10 times that of RSA can be saved</td>
<td>More than ECC</td>
</tr>
<tr>
<td>11</td>
<td>Encryption Time</td>
<td>Much Faster than RSA</td>
<td>At good speed but slower than ECC</td>
</tr>
<tr>
<td>12</td>
<td>Decryption</td>
<td>Slower than RSA</td>
<td>Faster than ECC</td>
</tr>
<tr>
<td>13</td>
<td>Small Devices efficiency</td>
<td>Much more efficient</td>
<td>Less efficient than ECC</td>
</tr>
</tbody>
</table>

Table 2.
Comparison between existing and proposed framework.
Information Systems Management

A 160-bit key would be a hundred million times harder than breaking the 1024-bit key. In below Table 4, discuss the measure of proposed ECC versus existing RSA public keys.

In Table 4, the proposed public key is generated using the existing key algorithm, based on key generation time (ms), memory size (bytes), and encryption/decryption time. During the ECC key generation, the time taken to generate the RSA key and ECC key and size has a huge difference (ECC-224 and RSA-2048). Also, it is learned that during the signature generation process, ECC surpasses RSA. Conversely,
Secured Storage Mechanism for the Cloud-Based Banking Applications Using ECC

DOI: http://dx.doi.org/10.5772/intechopen.107931

RSA beats ECC in performance during the verification process and the Encryption/Decryption process is ECC much better than RSA. The below Figure 8 shows the encryption and decryption time during the run time of the proposed public key algorithm [13, 14].

The below Figures 9–12 discusses the key size generation time, required memory size, and encryption/decryption time of the proposed ECC algorithm vs. the Existing RAS algorithm.
The above experiment result tells us that the proposed ECC algorithm is much more efficient than the existing RSA algorithm in Speed, memory size, and security [15, 16].

The analysis of encryption/decryption execution time for both the proposed design and the existing system has been discussed in the below report, experimented with the sensitive banking data, and file transfer in the cloud environment is measured in regards to file size and execution time (mile second). Results tell us that the

Figure 10. Algorithm required memory size (bytes).

Figure 11. Algorithm encryption/decryption time (ms).
proposed design is better than the existing system, specifically bank data handle in the cloud environment.

Figure 13 discusses the memory size of the sensitive data transfer on the cloud environment, memory size, and the execution time of the encryption/decryption that is measured for the particular document. In execution, the proposed ECC is much better than the existing RSA, Figure 14 discusses the experimental results clearly showing that the proposed security model can encode the data with very low processing time and a lesser amount of memory size in the cloud system.
5. Conclusion and future work

The proposed security scheme improves the level of protection currently supported in cloud storage mechanisms. The potential incompatibilities that arise from the simultaneous use of ECC, as well as the impact of user mobility on VPN operation, are considered, and detailed solutions are proposed. A design is proposed to focus on and improve the security level in the cloud environment. This research focuses on simulating and analyzing the performance of secure cloud storage. Though various methods are cited in the literature for the stabilization of cloud computing and portable device VPN connection [17–19], securities provided by these methods are not adequate for the protection of data packets. The proposed method provides two layers of security.

The work shall be extended to transaction-rich applications (Banking) as banking customers need more protection for fund transactions like payroll, CRM, accounting, invoicing and ERP, etc. Banks cannot afford the risk of a security breach. The current system has a single layer of security, so our proposed design will provide double-layer security with high performance and bandwidth.
References


