Asian Journal of Information Technology 16 (10): 746-753, 2017
ISSN: 1682-3915

Secured Cloud Computing with ECC: Web Services and Logging Mechanism

[1]V. Gopinath and [2]R.S. Bhuvaneswaran
[1]Department of Sciences and Humanities, Sathyabama University, Chennai, India
[2]Department of Computer Science and Engineering, Anna University, Chennai, India

Abstract: In this study, a stack of three level security, the security mechanism are proposed on top of Secure Socket Layer (SSL) Virtual Private Network (VPN) with Elliptic Curve Cryptography (ECC) applied for XML web services in cloud computing and the authentication of user that avoid the privacy risk in VPN. It provides a secure logging mechanism. The connection is established seamlessly when the user connects VPN on the cloud ensuring data security and privacy of users. On the existing SSL based VPN, ECC encryption enhances the security level of user's private data to create an encrypted key that is difficult to decrypt and thereby making it difficult to hack the network. Same digital ECC key is being archived by the web services component. By this mechanism, the security of data transmission from the VPN client to the server is assured. The server's data processing speed is effectively upgraded. This technique is analyzed and compared with the existing scheme. The process is more reliable and rapid based on the outcome of the groundwork.

Key words: ECC, SSL, VPN, XML, web services, cloud computing

INTRODUCTION

Cloud computing plays a major role by providing simple way to access resources like servers, storage, databases and a vast set of application services, secure communication is a significant need to perform high end data transfers. The profit and savings of cloud computing are wanted by many firms. Web service provides a consistent, enforceable security and strategy despite of where or how the end-user exploits the internet. In the existing scheme to launch the SSL based VPN, providing customized security for data packets. VPN server web service application is used by the client. The VPN client dynamically loads the XML based web services safety measures to provide incorporated safety measures.

The proposed system provides secure usage of cloud computing frame work based on web services protection intended to provide a high level design of the SSL VPN. This concept surpasses the protection level of the VPN that supports the connection of Peer to Peer (P2P) network via. ECC. The cloud computing framework offers a high level of confidentiality, safety and privacy for PaaS and IaaS which have security concerns initially. It secures web services and established network applications, same set of keys are being utilized by the VPN and web service. The XML based concurrent features create a lot of challenges when it comes in implementing security for web service through the internet, especially in cloud network. To meet the needs of web services protection, many hard works has freshly been made (Hashizume *et al*., 2013; Deshmukh *et al*., 2013) are worth mentioning.

The cloud service provider is unable to track user activities and transfers data effectively within cloud environment in this framework, a new concept is introduced, that is, i.e., generating logs for every activity of the user. In additional, we provide a secure logging system for upholding the logs thus shielding the confidentiality of users and integrity of logs. Liability can be a key to tackle such issues. Methods that promote accountability and audit ability of CSPs, logs generate is very crucial in cloud environment, cloud users can alter the generated logs. They can restructure the entries as well add some fake log entries. File-centric logging mechanism accounts all the entry happening on VM's and data transfer taking place between the VM's and external users. A secure logging scheme is provided which enable the user to check whether the logs provided by cloud service providers are not tampered.

Most vital characteristic of this content are ECC ANSI X9.62 (Deshmukh *et al*., 2013) is utilized for the private cloud VPN establishment this security system will provide advanced threats to defeat known and unknown threats. Provides fortification throughout the performance. Hence, better visibility and control offered. Cloud web services are established with the same ECC digital key. Secure logging mechanism for safeguarding of logs and track the user's activity details by the CSP.

Corresponding Author: V. Gopinath, Department of Sciences and Humanities, Sathyabama University, Chennai, India

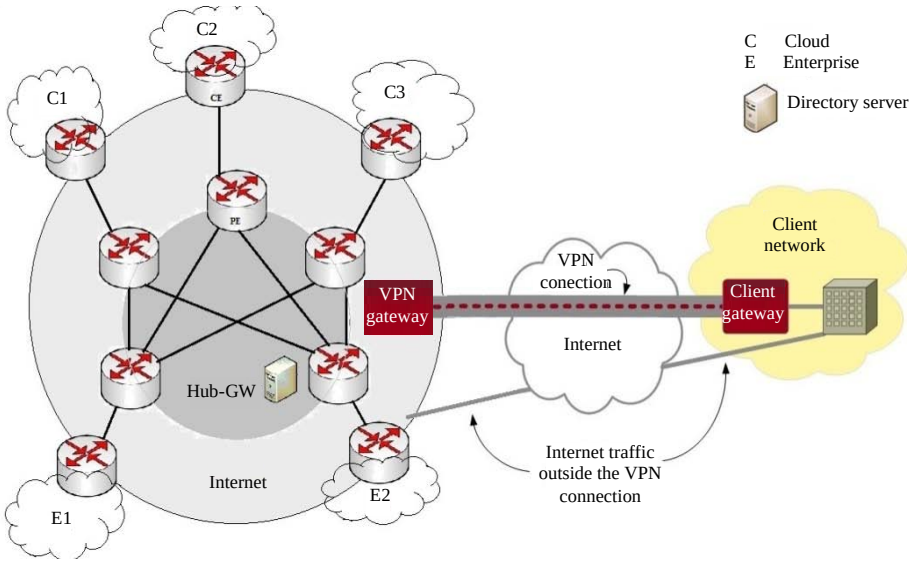

Fig. 1: High level design of the private cloud VPN system

Literature review: By Harfoushi *et al.* (2014), VPN methodology is appropriate for the application of cloud computing is based on hub-and-spoke and bipartite. The user has to hook up with hub-GW by using VPN. The supervision of hub-GW bring into play bipartite, divides users into categories of enterprise and cloud computing provider with only intercategory users allowed linking one another. VPN structure is classified into full-mesh and hub-and-spoke. In full-mesh each device is associated with others by committed lines. It can make sure rate quality and performance in hub-and-spoke, there's one node called Hub-GateWay (Hub-GW) and rest nodes are spoke nodes. Hub-GW is charged with the connection to all spoke nodes. By Hashizume *et al.* (2013), web service and long-established network based function utilize the same digital ECC key, soap message security, ECC algorithm and other supporting functions. By Hashizume *et al.* (2013), ECC cryptography is helps to increase the speed of encryption and decryption and shortening the CPU execution cycle. The algorithm point by Deshmukh *et al.* (2013) depends on a numerical crisis is more tricky for hackers to attack than the current encryption it can offer equivalent security with significantly minor key sizes.

Aforementioned algorithm tends to have some limitations. Web service protection is not available in the framework. Security is provided only when the link is recognized and RSA algorithm is inbuilt in SSL here RSA 128 bit key is used the occurrence of session reuse determines the operations of public key which eliminates the need of operations for some transactions the cost of encryption and hashing depends on the sum of data transferred. Here, XML web service and secured framework component with ECC is proposed (Fig. 1).

MATERIALS AND METHODS

Proposed system

Design of security system for private cloud VPN: The proposed system is aimed at providing a high level design of the SSLVPN with ECC which is applied on a private cloud. It helps in linking the P2P network through an elliptic curve cryptography. The projected security theory enhances the level of protection that currently supports the private cloud VPN and facilitates the realization of P2P network. Figure 2 planned structural design.

With the idea of cloud security, the design target is as follows 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 in to the system. Providing customized security for data packets which will be encrypted by ECC implementing risk assessment and security monitoring on TCP and UDP based protocols implementing a 2 layer security system to safeguard the integrity and confidentiality of user's private data to create a key that is difficult to decrypt and thereby making it impossible to hack the network providing customized secure logging mechanism which will be encrypted by ECC implementing a XML-based web services protection structure which will be encrypted by ECC. Figure 3 show the high level plan of the structure.

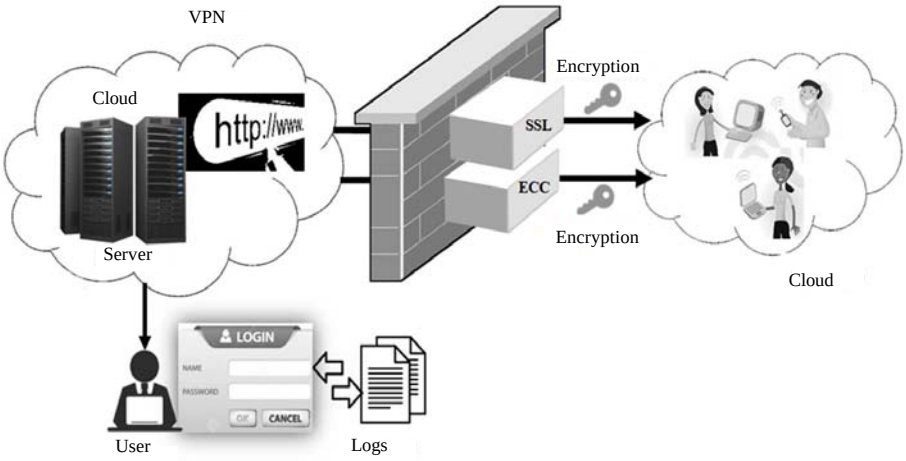

Fig. 2: VPN architecture with 3 level securities

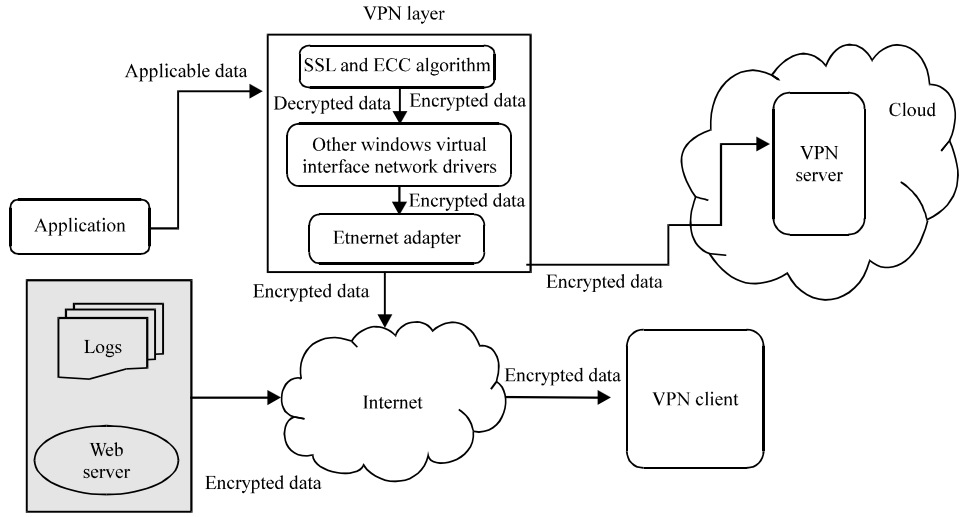

Fig. 3: Flow diagram of cloud VPN

Private cloud in VPN: This framework can be classified vastly into full-mesh and hub-and-spoke. In full mesh every node is linked straightly to others by committed lines. In hub-and-spoke there is one node known as Hub-GateWay (Hub-GW) and the rest are spoke nodes. Cloud VPN framework incorporate the theory of bipartite network (Harfoushi *et al*., 2014). The three aspects of this framework: CE, PE and directory server. CE is stimulated with routing at spoke end, PE is charged with diverting and forwarding at Hub-GW. Directory server is authentication server. If the enterprise needs to connect with other cloud services it only takes to activate the corresponded routes at directory server.

ECC encryption in VPN: ECC encryption technology addresses the weakness of RSA encryption in public key structure. It helps to increase the speed of encryption and decryption and limits the CPU implementation phase it also improves the data processing speed of the server (AWSI., 2014). The modified ECC is more secure and difficult for hackers to hack the existing encryption, it can offer equivalent security with substantially smaller key sizes (Marinos and Briscoe, 2009).

Elliptic curve digital signature algorithm: The ECDSA algorithm is providing handwritten signatures. It is a number dependent on some secret known only to the signer's private key and as well on the stuffing of the message being signed. Signature must be certifiable if a disagreement arises as to whether an entity signed a document an unbiased third party should be able to resolve the matter, equitably, without require right of

entry to the signer's private key. Disagreement may arise when a signer tries to deny a signature, it did create or when a faker makes a fake claim. The computational efficiency, minimal sized signatures in the ECDSA makes it ahead over other digital signatures.

Key generation: An entity A's private and public key pair is coupled with a particular set of elliptic curve (Liu *et al*., 2007) domain parameters (q, FR, a, b, G, n, h).

To generate a key pair, entity A does the following:

- Select a random number d in the interval [1, n-l]
- Compute Q = dG
- A′s public key is Q and A's private key is d

The setup for generating and verifying signatures using the ECDSA is the following. Suppose that signer A has domain parameters D = (q, FR, a, b, G, n, h) and public key Q_A also suppose that B has authentic copies of D and Q_A. In the subsequent SHA-1 denotes the ECC-160 bit hash function (Algorithm 1).

Algorithm 1; Elliptic curve digital signature algorithm for 160 bit hash function:

To sign a message m, A does the following:
1: Select a random integer k from [1, n-l]
2: Compute kG = (x1, yl) and r = xl mod n. If r = 0 then go to step 1
3: Compute k-1mod n
4: Compute e = SHA-l (m)
5: Compute $s = k^{-1} \{e+d_A \times r\} \bmod n$
If s = 0 then go to step 1
6: A′s signature for the message n′1 is (r, s)
To validate A′s signature (r, s) on m, B execute 1s the next steps:
7: Verify that rand s are integers in t [1, n-l]
8: Compute-e = SHA-l (m)
9: Compute $W = S^{-1} \bmod n$
10: Compute $u_l = ew \bmod n$ and $U_2 = rw \bmod n$
11: Compute $u_lG+u_2Q_A = (xl, yl)$
12: Compute v = xl mod n
13: Signature is confirmed if v = r

Web services security and framework component: VPN server web service process is used by the client. The client dynamically loads the XML web services protection factor in order to provide integrated security solution. It secures both web services and traditional network based application. Both utilize the identical digital ECC key, soap message security ECC algorithm and other supporting roles (Gopinath and Bhuvaneswaran, 2014; Hashizume *et al*., 2013). Web service safeguard unit firmly applies more measures if the needs are from external of VPN whereas, requests from the VPN call for no such actions. This factor has been built with Micro Soft (MS) packages of XML security soap security and ECC algorithm other supporting function based on MS

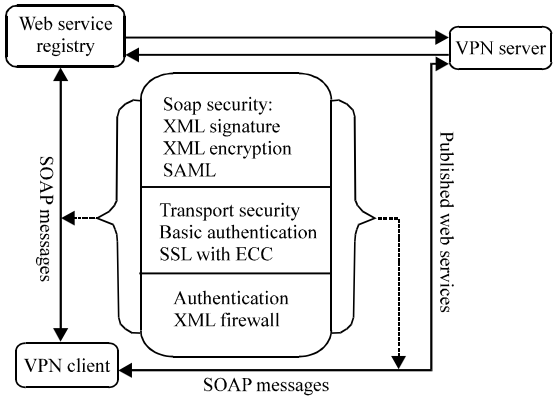

Fig. 4: High level design of the web service security environment

framework 4.0. A web services client VPN authenticates the user and creates two soap messages a client VPN soap message to server VPN. Therefore, developing and adopting XML-based web services protection standards is important to leverage the capabilities of web services (AWSI., 2014). Given a range of emerging web services security standards a cloud service provider should not prefer security solutions randomly. Figure 4 shows the web service security environment.

Dataflow in the VPN layer: VPN layer contains SSL and ECC algorithm other windows virtual interface network driver like VM's, finally, it sends the encrypted network traffic to VPN server via. the ethernet adapter. After receiving the encrypted network traffic, application traffic is redirected to internet it searches whether, the traffic needs to be sent through tunnel and if network traffic has to be channeled then the VPN client applies the encryption algorithm to the network traffic. Finally, it sends the encrypted network traffic to VPN server via. the VPN layer. This communication secure logging is done by the every user activities and web service, application access. It is through involuntarily exclusive of any pre-request and it enables the secured generation of public key and encryption and decryption. Like public clouds, private clouds also comprise multi-tenancy security requirements.

ECC based secure logging mechanism: Logging mechanism records every user file accesses, network access happening within the VM's thus, assuring complete transparency of entire consumer behavior in the cloud. Principally two types of logs are retained per user, i.e., file access logs and network logs. The following information is captured for every file access done on VM (Liao and Su, 2011):

- VM accessed file name and full path
- VM file access date time
- VM IP address
- Virtual machine MAC address
- Machine type VM/PM
- User ID of file owner of the accessed file
- Group IDs of file owner of the accessed file
- User ID of process owner who accessed the file
- Group IDs of process owner who accessed the file

Concept involved during the process of user identification is described as follows: Web log file F:

$$F = <f1, f2, ..., fn>, ..., n> = 0 \quad (1)$$

N is the total of log records in log file database. Format of r in Web log record:

$$f = <\text{file name, full path, date time, IP address, MAC address, Type VM/PM}> \quad (2)$$

Each domain in F is the attribute of log records used during the process of user identification. User:

$$User = <\text{UserID, GroupID, c\_ip, f\_date\_s, f\_time\_s, f\_date\_e, f\_time\_e, <fs, ..., fe>>} \quad (3)$$

UID is the identification number of this user the only identification of this user, Group ID is the identification number of this group user c-IP identifies the IP address of this user file access-date-s file-time-s identifies the initiation time of this user access date-e, time-e identifies the time when this user leaves the current access cloud, fs identifies the first piece of log record of this user access and fe identifies the last piece of access record. User identification is to find out the collection U of all users u corresponding in each piece of record r in log file R:

$$U = (User1, User2, ..., Userk), User \times c\_ip = ri \times c\_ip \quad (4)$$

Action done to accessed file, e.g., create, read, write, socket (send message) socket (receive message) delete. Apart from the file access logs, network logs, i.e., information about VM-VM contact and communication of VM with outside world is also captured. The network logs will store the following information: User ID of the user, Group IDs of user from IP2IP. Port number, time stamp information, i.e., date and time. After the file access logs and network logs are captured, it is sent to the underline Physical Machine (PM) where it is consolidated with PM's IP address and PM's MAC address (Algorithm 2).

Algorithm 2; Algorithm for secure loging mechanism:

```
Logging mechanism () {
UserID = ex. get type (). Name to string ()
GroupID = ex. get type (). G name to string ()
MACID = exg et type (). Address to string ()
IP = context.Current.Request.Url.ToString ()
Location = ex.Message.ToString ()
if (!Directory Exists (filepath))
{Directory Create Directory (filepath) }
filepath = filepath+date time today to string
("dd-MM-yy")+"txt"; //Text file name
if (!File.Exists (filepath)
{File.Create( filepath).Dispose () }
using (Stream Writer sw = File.AppendText(filepath)) {string Logging =
"Login Date:" +" "+ DateTime.Now.ToString ()+line + "UserID: "+" "+
UserID +line+"GroupdID:"+" "+ GroupdID+line+"MACID:"+" "
+MACID+line +"
Location:"+""+Location+line+"Page Url:"+""+exurl+line+"User Host
IP:"+""
+hostIp+line;sw.WriteLine("---User Logging Details on
"+""+DateTime.Now.ToString()+"---")
sw.WriteLine("---")
sw.WriteLine (line)
sw.WriteLine (logging)
sw.WriteLine
("---*End*---")
sw.WriteLine (line); sw. Flush (); sw. Close ()}
```

The consolidated logs are further sent for the encryption purpose (Zhang *et al*., 2010). Further, logs generated are crucial to the intact cloud atmosphere hence, they should be safeguarded. Since, catalog in which the logs are stored comes under the cloud service providers architecture there are chances that administrator or some malicious cloud employee can delete some log entries or can reorder the log entries or even add some fake entries to it. Thus, a mechanism for preventing such tampering of logs is needed. Once log record is saved, the system will furthermore stock up the proof of this entry in the proof database. When an examiner needs logs of a particular user to inspect a happening he can get the essential logs by an API call. In order to confirm that, logs as not tampered proof of logs is provided along with logs.

RESULTS AND DISCUSSION

Cloud computing is characterized by the accessibility at whichever instance as desired with reason of dipping managerial costs and increasing efficiency in operation in secured way. The ECC VPN that connects cloud computing should possess the above mentioned characteristics. The network atmosphere should be trouble-free to set up modernize or erase connections according to the demand for cloud computing, so as to ease the complex system management. In Table 1, most of the efforts were talked about to recognize, sort, explore and number of vulnerabilities and threats focused on cloud computing. It describes the vulnerability and threats analysis report that are related to the analysis test

Table 1: Vulnerability and threats analysis report and comparison between existing framework and proposed frameworks

Vulnerability and threats analysis	Existing method	Proposed method	Analysis report	Existing frameworks	Proposed framework
Secure even after loss password	O	O	Configuration management	Single	Single
Indexing of data, keyword search	O	O	Fault management	Simple	Simple
Insecure interfaces and APIs, account or service hijacking	O	O	Performance management	Integrated	Integrated
Unlimited allocation of resources, denial of service	O	O	Security	Support	support
Data-related vulnerabilities, data leakage	X	O	Management	directory server	encrypted data and directory server
Vulnerabilities in virtual machines, VM hopping	X	O	Accounting management	Convenient	Convenient
Vulnerabilities in virtual machine images, data leakage	X	O	Confidentiality	File encryption not provided	Symmetric key
Vulnerabilities in hypervisors, escape	X	O	Authentication	Only base level security provided	Password-basedVMadvance level security provided
Vulnerabilities in virtual networks, VM escape customer-data manipulation	X	O	Access control	Exposure to the normal area	Encryption of security area information
Log security in virtual machine, data packets	X	O	Impersonation attack	No impersonation	two stages of user authentication
O-provided, secured X-not provided, not secured			Efficiency	Easy to implement	Easy to implement

used in cloud environments and it specifies that the cloud service representations are out in the open to these vulnerability and threats. We put more prominence on threats that are allied with statistics being stocked up and practiced remotely, sharing resources and the usage of virtualization.

Given such presupposition, the application of ECC VPN based encryption and decryption algorithm in cloud computing on various management and we found in the every case it was working efficiently and also satisfies the below requirements. It has solitary connections to Hub-GW hence, it's easier as the user needs to manage the setting for only one single configuration. It has only one connection, so in case of faulty connection, only the fault management for this connection needs inspection so, it is easy. The consumer only needs to monitor the connection of Hub-GW no additional management mechanisms are necessary even with changing the cloud computing dynamically. It has directory server which is dedicated to connection management and is able to ensure the enterprises connect to the right cloud computing service providers. It manages in a concentrated manner which is much more convenient.

This study presents the evaluation of portable VPN replacing RSA with ECC in the SSL protocol. ECC provides 160 bit security key. Algorithms involving public keys tend to reuse the session frequently for transaction which accounts for cost whereas encryption and message hashing amounts for the data transferred.

ECC performance measure of public key algorithms: In cloud VPN SSL protocol, RSA is replaced by ECC. ECC is a public key cryptography technique providing 160-bit security key, the effectual outlay of public key procedure is determined by the frequency of session reuse which eliminates the need for public-key operations for some transactions the charge of encryption and muddling depends on the quantity of information conveyed. We make use of open SSL speed plan to evaluate RSA decryption and ECDH function for different key sizes. Outcome of the projected system is given Table 2.

Table 2: Measure of various public keys

Algorithm	Key generation time (msec)	Required memory size (bytes)	Encryption/decryption time (msec)
ECC (160)	108	125	16
RSA (1024)	2609	313	388
ECC (224)	121	140	15
RSA (2048)	18399	621	1867

Table 3: Public key performance measure

Veriables	Key-size ratio	Performance ratio
ECC-160	1.0	2.4
RSA-1024	6.4	1.0
ECC-192	1.0	7.1
RSA-1536	8.0	1.0
ECC-224	1.0	11.0
RSA-2048	9.1	1.0

The public key operation for ECC-160 is only 3.69 m sec it is 50% comparatively<RSA-1024 and other keys (Table 3). The flowchart given below explains that a key generation time and required memory size for both ECC key and 1024 bit RSA key, 160-bit ECC much better than RSA (ANSI X9.62, 1999). The protection measures for both 160-bit ECC key and 1024 bit RSA key similar. Hence, breaking a 160 bit key would be a hundred million times harder than breaking the 1024 bit key (Fig. 5-8).

Note that, the ECC-160 key size and performance ratio is more advantageous than RSA it provides better security needs than RSA (Zhang *et al*., 2010).

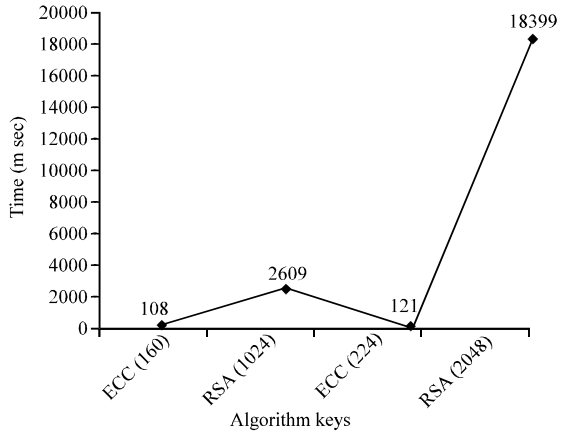

Fig. 5: Algorithm key generation time (m sec)

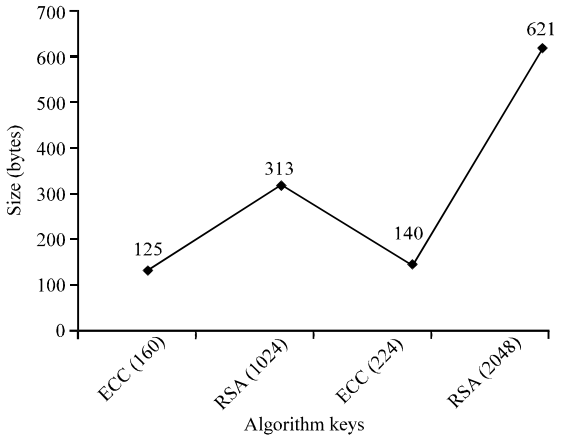

Fig. 6: Algorithm required memory size (bytes)

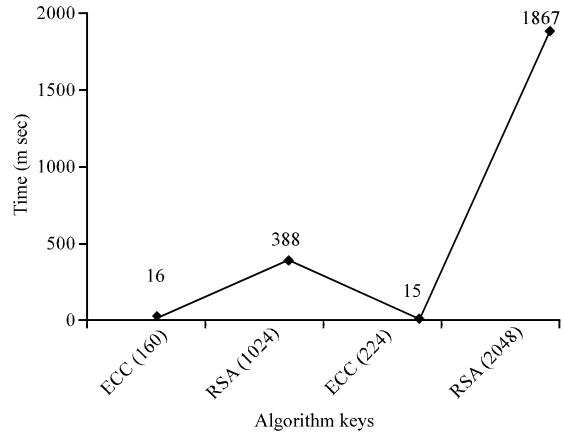

Fig. 7: Algorithm encryption/decryption time (msec)

The planned security proposal is employed as a supplementary aspect to the private cloud VPN standard. It contain a wide-ranging collection of configurations

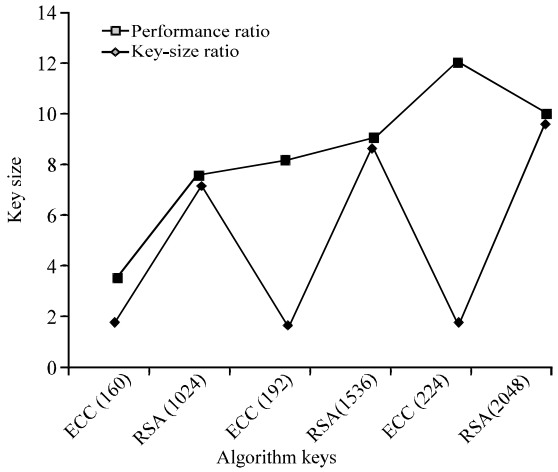

Fig. 8: Performance measure of various public keys

in adding together site-to-site VPNs, Wi-Fi security, enterprise size solutions and remote access provide many options for controlling the security of the VPN client and options for protecting the security of the server itself.

CONCLUSION

The above analysis suggests that, the framework is suitable for the implementation of cloud computing as it relies on hub-and-spoke and bipartite. The user has to connect hub-GW by using VPN which is bounded with ECC to accept the performance benefits of SSL clients, especially for the servers as the security needs are arising. The proposed security design advances the stage of safeguard in existing private cloud. It secures data transmission over the entire network route by utilizing the default P2P network over the private cloud the potential incompatibilities that arise from the concurrent use of ECC as well as the impact of user mobility on VPN operation is considered and detailed solutions are proposed also we have focused on security aspect of web services and integration of the web services security component with the application server is proved to be feasible and efficient.

REFERENCES

ANSI X9.62, 1999. Public key cryptography for the financial services industry. The Elliptic Curve Digital Signature Algorithm.

AWSI., 2014. AWS cloud security. Amazon Web Services, Inc., Seattle, Washington, USA. https://aws.amazon.com/security/.

Deshmukh, G., S. Powar and B.B. Meshram, 2013. Web services security standards. Intl. J. Emerging Technol. Adv. Eng., 3: 145-149.

Gopinath, V. and R.S. Bhuvaneswaran, 2014. Design of security system of portable device: Securing XML web services with ECC. Proceedings of the International Symposium on Security in Computing and Communication, September 24-27, 2014, Springer, Delhi, India, pp: 431-439.

Harfoushi, O., A. Bader, A.G. Nazeeh, O. Ruba and A.F.M. Mua'ad *et al*., 2014. Data security issues and challenges in cloud computing: A conceptual analysis and review. Commun. Netw., 6: 15-21.

Hashizume, K., D.G. Rosado, E. Fernandez-Medina and E.B. Fernandez, 2013. An analysis of security issues for cloud computing. J. Internet Services Applic. 10.1186/1869-0238-4-5.

Liao, W.H. and S.C. Su, 2011. A dynamic VPN architecture for private cloud computing. Proceedings of the 4th IEEE International Conference on Utility and Cloud Computing (UCC) 2011, December 5-8, 2011, IEEE, Taipei, Taiwan, ISBN:978-1-4577-2116-8, pp: 409-414.

Liu, Y., T.H. Yeap and O. William, 2007. Securing XML web services with elliptic curve cryptography. Proceedings of the Canadian Conference on Electrical and Computer Engineering (CCECE 2007), April 22-26, 2007, IEEE, Ottawa, Canada, ISBN:1-4244-1020-7, pp: 974-977.

Marinos, A. and G. Briscoe, 2009. Community cloud computing. Proceedings of the 1st International Conference on Cloud Computing, December 1-4, 2009, Beijing, China, pp: 472-484.

Zhang, S., S. Zhang, X. Chen and X. Huo, 2010. Cloud computing research and development trend. Proceedings of the 2nd International Conference on Future Networks (ICFN 10), January 22-24, 2010, IEEE, Tangshan, China, ISBN:978-1-4244-5666-6, pp: 93-97.