International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 3 Issue: 5 3042 - 3048

Implementation of Secure Log Management Over Cloud

Mr. Kolhe Harshal Narayan
Department of Computer Engineering,
S.N.D.College of Engineering, Babhulgaon.
Dist- Nashik, Maharashtra, India
harshalkolhe352@gmail.com

Prof. Imran R. Shaikh
Department of Computer Engineering,
S.N.D.College of Engineering, Babhulgaon.
Dist- Nashik, Maharashtra, India
imran.shaikh22@gmail.com

Abstract- A Log records are very important information which is related to activities of systems, applications or networks and these log records having various fields and their syntax. Actually logs are automatically generated on activities that are done and doing by user on system, or on any Applications such as Google Chrome or in networks. These logs are costly and need to any organization for future references such as to identify or finding any problems, to record all events, to find performance, and to investigate malicious activities in systems or networks or in application. So, protection of logs from attackers is required. Hence organization should maintain integrity, confidentiality, and security of logs. The cost to maintain logs for organizations for longer period is very less. Hence, we developed secure log management over cloud to decrease cost as well as provide security of log from attackers. To achieve this, we have done this with the help of Blowfish algorithm to Encrypt log records then SHA-1 is used to provide confidentiality while transmitting and at end point security purpose we used Shamir's Secret sharing algorithm.

Keywords—*privacy, encryption, security, cloud computing, log record management, integrity, secret sharing.*

*****

I. INTRODUCTION

A Log is the recording the information of activities on systems or applications such as Google chrome or on networks running in any organization. Log files are very useful to find problem. Log records are use to solve problems such as to identify the fraudulent works, security incident, and any policy violations. Log consists of costly, useful, and very important information for any organization so that's why we have to provide protection from third party attacker.

A. Generation and Maintenance of Log

The generation of log is depends on activities done on any application such as we have used History of Google chrome is a log file to system. These logs having number of steps to compute secured log management. Steps are generation of log, storage of log, analyzing of log, transmission, displaying of secure log data. For any type of organization required log generation and maintenance. But it is more complicated by some factors like lack of log resources, improper logs content, lack of proper format, and timestamp of each sources, and large or heavy volumes of log data. Log management consist maintenance phase, this maintenance means to achieve properties for example confidentiality, integrity, and availability of logs. To Designing secure logging information for all the above challenges cloud management is best way for any organizations.

B. Logs storage to Cloud

The storage on cloud is the best medium for storage purposely and authenticated user can access from anywhere. The storage on cloud required minimum resources to end users. In storage on cloud, any type of data is delivered acts as a service (Xaas) and there are some main services models :

1) **Software as a Service (SaaS)** - This SaaS service is used to provides or delivers softwares on internet. SaaS consist of running the software on the provider's cloud infrastructure. The software delivery is use to a single or multiple clients as per demand through a thin client e.g. browser .

2) **Platform as a Service (PaaS)** – This PaaS service provides the flexibility for a client to build, develop, test and deploy applications on provider's platform. PaaS service provides the infrastructure besides PaaS. It provides the platform and development tools to the end user i.e end PaaS user.

3) **Infrastructure as a Service (IaaS)** –This IaaS Service provide access to resources as per demand such as network related works, servers and storage for organization, can be accessed with a service API.

The owners of the infrastructure of cloud computing is depends on following four deployment models and this raises security issues.

A. **The Public Cloud**: It is basic view of cloud computing. Generally Public cloud owned by large organization, to provide availability to the general public over the Internet by a multitenant model on self-service basis. This Public cloud is cost effective cloud for saving, security, privacy issues from physical/ actual location of the infrastructure provides.

B. **The Private Cloud:** Private Cloud infrastructure is used by a single tenant environment, and which is to manage by the single organization or by third party within or

 ISSN: 2321-8169
3042 - 3048

outside the premises. This cloud having more cost than the public cloud model.

C. **The Community Cloud:** Community Cloud model infrastructure used or shared by multiple organizations of a specific community, which is to be managed by any one of the organizations or a third party.

D. **The Hybrid Cloud:** Hybrid Cloud model is combination of any two or all of the three models discussed above. However user can select the models based on organization's requirements but the major security issues raises in cloud computing are availability of data, data secure, third party control, Privacy and legal issue based on the model.

C. Secured Log and challenge of Secured Log :

The above discussed issues over a cloud environment have to provide a secured logging as a services, there are some properties to achieve are as below:

A) **Availability:** Availability property of cloud management is the logs over cloud storage must be available at any time as per demand. This availability is a high prior for cloud database users.

B) **Verifiability:** Verifiability with respect to Cloud management property is used to verify each and every entry in the log is present and did not modify from attacker. Each and every entry in logs must be verified its authenticity independent of others, and there must be all entries have linked together in such a way that to determine whether any entries are missing or not by using this linked entries..

C) **Privacy**: On Cloud storage, Log records would be distributed globally which may raise issue of the data exposure and privacy of information over a Cloud.

D) **Confidentiality**: Log records should not be easily searched to get sensitive information called as Confidentiality. Only Legitimate search access to users such as auditors or system administrators should be allowed only. No one have privileges to prevent an attacker who has logging system from accessing sensitive information that the system would put information in future log entries, the aim of confidentiality is protect the pre-compromised log records from confidentiality breaches.

To achieve these all above challenges a secure way log generation for example extracts, transforms, and encryption must be required.

II. LITERATURE SURVEY

Some techniques related to Secure Logging with disadvantages are as shown in Table 1. C. Lonvick, D.Ma and Indrajit Ray and co-authors introduced the major techniques and models used for secure logging. The different authors introduced different techniques, but these are often so inaccurately in practice it is not possible to verify performance.

Table 1: Various techniques for Secure Logging with disadvantages

Researcher	Year	Technique	Drawback
M. Bellare, B.S.Yee.	Nov-1997	Forward integrity of log.	Requires guaranteed server to maintain secret keys, and it may be attacked by unauthenticated user.
C. Lonvick	Aug-2001	Syslog	Uses UDP protocol, UDP Protocol is not providing reliable delivery of log message.
D.New, M.Rose	Nov-2001	Reliable Syslog	Does not prevent against confidentiality.
U.Flegel	Oct-2002	Syslog-pseudo	Does not ensure about correctness of logs.
J.E.Holt	2006	Logcrypt	Truncation Attack possible.
D.Ma, G. Tusdik	March-2009	Forward Secure sequential aggregate authentication	Efficient but very costly
J.Kelsey, J.Callas	May-2010	Syslog-sign	Does not provide privacy or confidentiality during transmission of data or at end.
Balabit IT Security	Sept-2011	Syslog-ng	Does not protect against log data modification.
Indrajit Ray, K. Belyaev	June-2013	Secure Logging As A Service-Delegating Log Management to the Cloud	Most efficient and secured technique but loosely coupled architecture.

III. SYSTEM ARCHITECTURE

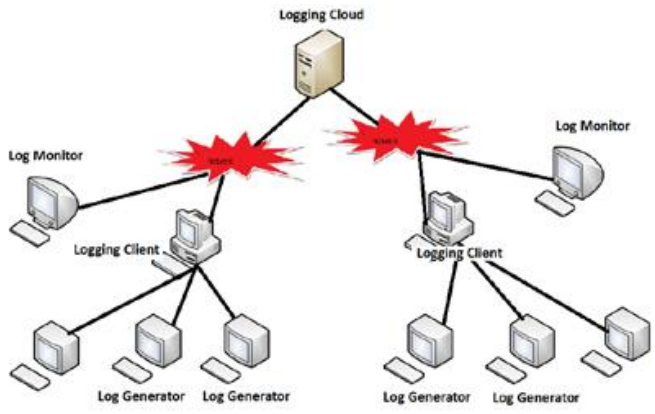


Figure 1: System Architecture

The Architecture of the secure log management over Cloud system is shown in above Fig. There are four important parts in this system.

1) Log Generators: Log Generators are computing devices that are used to generate log information of an Application such as Google Chrome. Every generator is capable of logging. The log files generated by these hosts are not hold on local machine except temporary until they're pushed to the logged client.

2) Logging client or Logging Relay: The logging client is act as collector that receives group of log records generated by one or a number of log generators, and prepares the log information in order to that it may be pushed to the cloud for future storage. The log information is transferred from the generator to the client in batches, either on a schedule, and once required depends on the quantity of log information waiting to be transfer. The logging client combines security protection on batches of accumulated log information and pushes every batch to the logged cloud. Once the logging client pushes log information to the cloud it acts as a logging relay. The term logging client and logging relay use to interchangeably. The logging client or logging relay may be enforced as group of collaborating hosts. For simplicity, we tend to assume that there's one logging client. Logging clients is capable to perform encryption of Log and generate MAC code to provide security of Logs during transmission as well as at the end also by using Shamir's secrete Sharing algorithm.

3) Logging Cloud The logging cloud provides future storage and maintenance service to log information received from completely different logging clients to different organization. The number of logging cloud is maintained by a cloud service provider. Those organizations that have signed to the cloud's services will transfer information to the cloud. The cloud, as per demands from a company can delete log information and perform log rotation. Before the cloud can delete or rotate log information it required proof from the requester that the latter is allowed to form such demands. The logging clients generate such proof. However, the proof may be given by the logging client to any entity that it needs to authorize. When Logging cloud is logged, Logs encrypted as well as shared is to be stored on cloud.

4) Log Monitor Log monitor are hosts that can be used to monitor and review log information. They'll generate queries to retrieve log information from the cloud. These monitors can perform additional analysis as per demands. They'll additionally raise the log cloud to delete log information for good, or rotate logs. We assume that the organization maintains the log generators and also the logging client. The log monitor may be maintained by organization or may be a separate entity. The logging client also can play the role of a log monitor. We tend to develop our model that the log monitor could be a separate entity that's trust by the logging client. The logging client and log monitor operate freely of every other, they'll communicate in asynchronous manner. This implies that if a logging client needs to send some information to the log monitor (vice versa), the sender can't expect the receiver to be on-line to receive the info. The logging cloud facilitates this communication by receiving and servicing applicable requests.

These are main parts of our system. Mostly we use the Log generators of lower configurations. Logging client and Log monitor is one and same system. And Logging Cloud is use as private cloud to storage purpose.

The most contributions are the design of the various parts of the system and developed encrypted protocols to maintain confidentiality and integrity problems during maintaining, storing, and retrieving log records at the honest however curious cloud provider and in transit. Disadvantage of existing system is that it can't show the privacy and confidentiality with log file storage and retrieval. The client uploads the information in batches and every batch is delimited by start of log record and end of log record. The cloud provider gets log records from its authenticated clients. The throughout upload logging client must authenticate to cloud to prove the client had obtained previous authorization from the logging cloud to use the services. However, it can't wish the identity of the logging client to be connected to any of its transactions include the authentication method. For this purpose we developed four protocols for upload, retrieval, deletion of log data on cloud with secure manner that steps are as follows.

A. Upload log Generation

We have uploaded log records from Clients. The log of Google Chrome is generated from number of clients. So, the log data can be store at the cloud when upload. This upload is done by the logging client. To Retrieved log details from the cloud by the logging monitor can send a request of retrieve to the logging cloud. Any attacker can try to use the upload data to retrieve the log data. The log data must be deciphered if and only if the corresponding decryption key is available.

B. Upload logs data

The entity that requires to upload the log data sends request message. In the request message consists of the upload of related to the desired log data. With the help of communication channel the logging monitor can send log data to the logging cloud. In the upload message none value is use individually or in a group can be tied to the logging client. The logging cloud and logging monitor send predefined formatted message to the logging cloud in order to the upload or retrieve any piece of information.

C. Retrieve Log Data

This Retrieve Log data is use to download or retrieve log data from cloud. The logging cloud can retrieve the data from its storage location and sends that data over the channel to the

requester. The cloud provider does not require authenticating the requester of logging client. This is required for quality of the log batches has been encrypted; the retrieved data is useful only to those who have the valid decryption keys to encrypt logs.

D. Delete Logs

The requester sends delete message to the logging cloud to delete log data. The cloud provides the response to requester as challenges to the requester. The authorization proves to delete by presenting a correct delete tag.

IV. ALGORITHM

1. Blowfish Algorithm for Encryption of Log record:

1. **Blowfish Algorithm:**
1. Divide input x into two 32-bit halves: xL, xR.
2. Then, for i = 1 to 16:
xL = xL XOR Pi
xR = F(xL) XOR xR
Swap xL and xR
3. After the sixteenth round, swap xL and xR again to undo the last swap.
4. Then, xR = xR XOR P17 and xL = xL XOR P18.
5. Finally, recombine xL and xR to get the ciphertext.

2. SHA-1 is used for MAC

SHA1 stands for \Secure Hashing Algorithm. It is the improvement upon the original SHA0. SHA1 is currently the most widely used SHA hash function, although it will soon be replaced by the newer and potentially more secure SHA2 family of hashing functions. It is currently used in a wide variety of applications. SHA1 outputs a 160bit digest of any sized file or input. It uses a 512 bit block size and has a maximum message size of 2^{64}-1bits:

SHA1 Algorithm Description:

1. Padding

(a) Pad the message with a single one followed by zeroes until the _nal block has
448 bits.

(b) Append the size of the original message as an unsigned 64 bit integer.

2. Initialize the 5 hash blocks (h0,h1,h2,h3,h4) to the specific constants defined in the SHA1 standard.

3. Hash (for each 512 bit Block)

(a) Allocate an 80 word array for the message schedule

i. Set the _rst 16 words to be the 512 bit block split into 16 words.

ii. The rest of the words are generated using the following algorithm.
Word[i-3] XOR word[i-8] XOR word[i-14] XOR word[i-16] then rotated 1 bit to the left.

(b) Loop 80 times doing the following.

i. Calculate SHAfunction() and the constant K (these are based on the current round number.

ii. e=d

iii. d=c

iv. c=b (rotated left 30)

v. b=a

vi. a = a (rotated left 5) + SHAfunction() + e + k + word[i]

I Add a,b,c,d and e to the hash output.

4. Output the concatenation (h0,h1,h2,h3,h4) which is the message digest.

3. Shamir's Secret Sharing Algorithm for Sharing:

It divides data D into n pieces in such a way that D is easily reconstruct from any k pieces, but even complete knowledge of k – 1 pieces reveals absolutely no information about D. This technique enables the construction of robust key management schemes or cryptographic schemes that can function securely and reliably even when misfortunes destroy half the pieces and security breaches expose all but one of the remaining pieces. We generalize the problem to one in which the secret is some data D (e.g., the safe combination) and in which non-mechanical solutions (which manipulate this data) are also allowed.

Our goal is to divide D into n pieces D1, ..., Dn in such a way that:
(1) knowledge of any k or more Di pieces makes D easily computable;
(2) knowledge of any k – 1 or fewer Di pieces leaves D completely undetermined (in the sense that all its possible values are equally likely). Such a scheme is called a (k, n) threshold scheme. Efficient threshold schemes can be very helpful in the management of cryptographic keys.
In order to protect data we can encrypt it, but in order to protect the encryption key we need a different method (further encryptions change the problem rather than solve it). The most secure key management scheme keeps the key in a single, well-guarded location (a computer, a human brain, or a safe). This scheme is highly unreliable since a single misfortune (a computer breakdown, sudden death, or sabotage) can make the information inaccessible. An obvious solution is to store multiple copies of the key at different locations, but this increases the danger of security breaches (computer penetration, betrayal, or human errors). By using a (k, n) threshold scheme with n = 2k – 1 we get a very robust key management scheme.

V. MATHEMATICAL MODEL:

Design of Secure Log Management over Cloud is of P Class because:

1) Problem can be solved in polynomial time.

2) And always produce strong results.

Let S be the set of Inputs, Functions and Outputs S = {I,F,O}where I represents input log file and encryption keys which is input to log files, F represents the set of functions that are performed on the input. O is the Set of output.

Inputs:

I1= Log File

I2=Encryption Keys

Functions:

F1= Log File Preparation for Secure Storage

F2= Secret Sharing Module

F3= Upload, Retrieval and Deletion of Log Data

Output:

O1= Retrieve Encrypted File

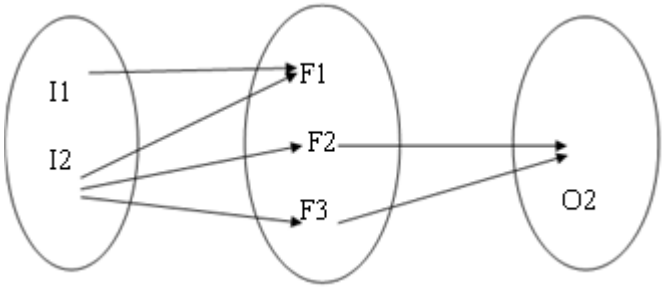
Sets:

I = I1, I2

F= F1, F2, F3

O=O1

Venn diagram:



VI. PERFORMANCE AND RESULT

The Performance is done to find out the correctness of Log Management over Cloud with other existing log management system. When we measures specific to Log management over Cloud with respect to the security, we have provide material which strongly suggests that Log management over Cloud is profitable to maintain security of the logs.

Algorithm wise Analysis

1. Encryption with Blowfish

In this Module, Logging client has taken the log data from different log generators based on same network. Logging client had encrypted that log data and MAC generated by using SHA-1. For integrity, MAC is added to log records. And again encrypt that records with MAC to more secure.

1.1. Result Analysis of Encryption:

To encryption of Log files, we have taken some log history of system. From each system, We have taken different log files of different Sizes. When we applied Blowfish Encryption algorithm on that files, result was encrypted files which was not able to read by human being easily. As well as size of original files was larger than encrypted file.

Table 10.1: Result Analysis of First Module

No.	Number of Entries	Actual size(KB)	Blowfish	AES
1	19	92	37	90
2	35	247	45	112
3	48	321	45	164
4	54	320	45	164

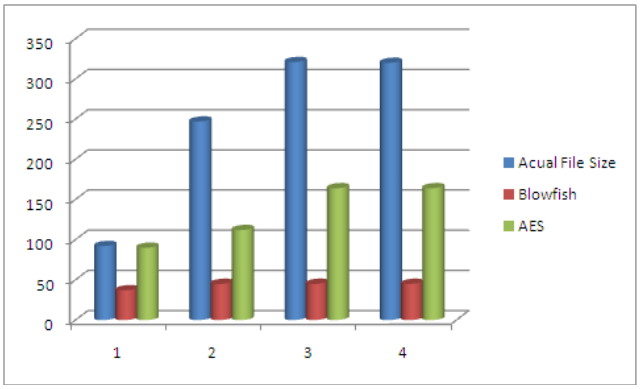


Figure 2: Comparison between Blowfish and AES

2. SHA-1 for Message Authentication Code

In this module, we have generated random sequence of code i.e. Message Authentication Code. This MAC is different for sender and receiver because to provide secure data while transmitting as shown in below. This MAC is also encrypted to provide more security. Secure Hash Algorithm 1 is selected because outputs a 160 bit digest of any sized file or input. It uses a 512 bit block size and has a maximum message size of 2^{64}-1bits.

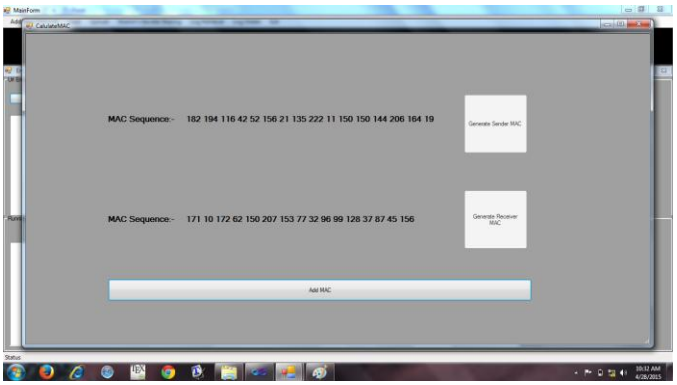


Figure 3: Add MAC for Sender and Receiver

ISSN: 2321-8169
3042 - 3048

3. Secret Sharing Module

In this Module, we have shared secret data in three parts to provide integrity and confidentiality. If we keep data on single storage, that can be get by third party. So to avoid this, we have used secret sharing scheme to protect against such a possibility. Hence, We have use Shamir`s secret sharing scheme. The idea behind these schemes is that at the end of a fixed period of time, the shares stored at each host change although the original secret stays the same. When any attacker make changes or delete the file over cloud, then we provide the recovery of secret facility in this module. To analyze the correctness of this module, we delete a secret file among three and then tried to recover the secret. It works greatly and produces the lost data from other two different files.

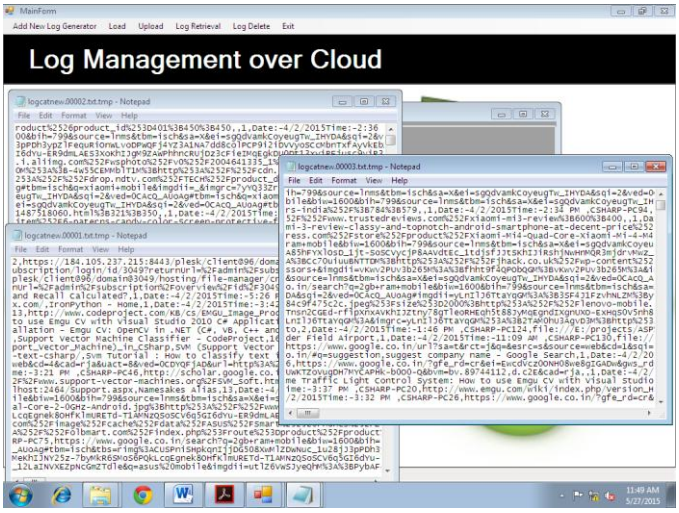


Figure 4: Secrete Sharing Algorithm to share

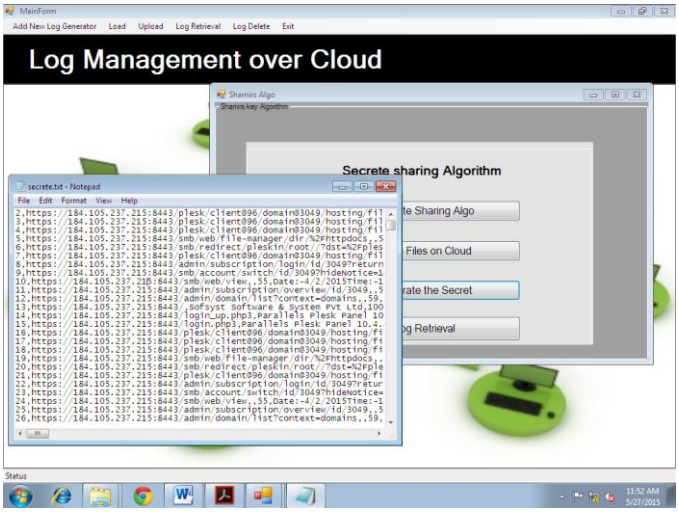


Figure 5: Secret of File to recover

4. Complete performance

We have developed this system with these algorithms to improve previous systems. First We have selected Blowfish algorithm because this is high secured and takes less space than other as we have already seen in Encryption with Blowfish. Secure Hash Algorithm 1 is selected because outputs a 160 bit digest of any sized file or input. It uses a 512 bit block size and has a maximum message size of 264-1bits. And finally, we have used Shamir`s secret Sharing algorithm because If we keep data on single storage, that can be get by third party. So to avoid this, we have used secret sharing scheme to protect against such a possibility. Hence, We have use Shamir`s secret sharing scheme. With the help of these algorithms we have developed this complete system. In system, We have taken history file of google chrome as a input. These history files are logs of different systems. Then we have encrypted that records and generated MAC for sender as well as for receiver to provide security during transmission. For high security, we have encrypted that MAC and records also. Finally, we have stored that information on cloud with secret sharing in 3 parts. When records are stored on cloud that is in encrypted format and when we retrieved records then decryption is done. According to the above , our system is more secured, takes less time as well as takes less space to manage Logs over Cloud.

VI. Conclusion

Log records are very important to any organization. Log is the activities or event which are done by user. This logs are most important to attacker. This logs keep secured is very important to organization. And organization wants to keep records for longer time with less cost. We have achieved this with the help of some algorithms such as Blowfish encryption algorithm to encrypt logs then SHA-1 for Message Authentication Code to provide confidentiality while transmitting data from sender to receiver and vice versa. Finally, Shamir`s Secrete Sharing Algorithm is used to keep encrypted logs on various location. Because, if attacker tries to get data, he or she can get attacked on some data which is not useful to them. If attacked on some data that can be recover by this algorithm. Hence, in this way we have maintain the Log management over Cloud with high secured, it includes Log collection, transmission, stored, and retrieval of Log with authentication.

VII. ACKNOWLEDGMENT

I thank to my project guide and PG Coordinator Prof. Imran R. Shaikh, And Our Head of Computer Department Prof. S.R. Durugkar and our faculty member of Computer engineering, S.N.D.College of Engineering, Babhulgaon. Without their importance this development could not be completed.

References :

[1] Indrajit Ray, Kirill Belyaev, Mikhail Strizhov, DieudonneMulamba, MariappanRajaram “Secure Logging

As a Service—Delegating Log Management to the Cloud" IEEE SYSTEMS JOURNAL, VOL. 7, NO. 2, JUNE 2013

[2] U.S. Department of Health and Human Services. (2011, Sep.). HIPAA—General Information [Online]. Available: https://www.cms.gov/hipaageninfo

[3] PCI Security Standards Council. (2006, Sep.) Payment Card Industry (PCI) Data Security Standard—Security Audit Procedures Version 1.1 [Online]. Available: https://www.pcisecuritystandards.org/pdfs/ pci−audit−procedures−v1-1.pdf

[4] Sarbanes-Oxley Act 2002. (2002, Sep.). A Guide to the Sarbanes-Oxley Act [Online]. Available: http://www.soxlaw.com/

[5] C. Lonvick, The BSD Syslog Protocol, Request for Comment RFC 3164, Internet Engineering Task Force, Network Working Group, Aug. 2001.

[6] D. New and M. Rose, Reliable Delivery for Syslog, Request for Comment RFC 3195, Internet Engineering Task Force, Network Working Group, Nov. 2001.

[7] M. Bellare and B. S. Yee, "Forward integrity for secure audit logs," Dept. Comput. Sci., Univ. California, San Diego, Tech. Rep., Nov. 1997.

[8] BalaBit IT Security (2011, Sep.). Syslog-ng—Multiplatform Syslog Server and Logging Daemon [Online]. Available: http://www.balabit.com/network-security/syslog-ng

[9] J. Kelsey, J. Callas, and A. Clemm, Signed Syslog Messages, Request for Comment RFC 5848, Internet Engineering Task Force, Network Working Group, May 2010.

[10] D. Ma and G. Tsudik, "A new approach to secure logging," ACM Trans. Storage, vol. 5, no. 1, pp. 2:1–2:21, Mar. 2009.

[11] U. Flegel, "Pseudonymizing unix log file," in Proc. Int. Conf. Infrastruture Security, LNCS 2437. Oct. 2002, pp. 162–179.

[12] M. Rose, The Blocks Extensible Exchange Protocol Core, Request for Comment RFC 3080, Internet Engineering Task Force, Network Working Group, Mar. 2001.

[13] B. Schneier and J. Kelsey, "Security audit logs to support computer forensics," ACM Trans. Inform. Syst. Security, vol. 2, no. 2, pp. 159–176, May 1999.

[14] J. E. Holt, "Logcrypt: Forward security and public verification for secure audit logs," in Proc. 4th Australasian Inform. Security Workshop, 2006, pp. 203–211.

[15] D. Dolev and A. Yao, "On the security of public key protocols," IEEE Trans. Inform. Theory, vol. 29, no. 2, pp. 198–208, Mar. 1983.

[16] A. Shamir, "How to share a secret," Commun. ACM, vol. 22, no. 11, pp. 612–613, Nov. 1979.

[17] G. R. Blakley, "Safeguarding cryptographic keys," in Proc. Nat. Comput. Conf., Jun. 1979, p. 313.

[18] R. Ostrovsky and M. Yung, "How to withstand mobile virus attack," in Proc. 10th Ann. ACM Symp. Principles Distributed Comput., Aug.1991, pp. 51–59.

[19] A. Herzberg, S. Jarecki, H. Krawczyk, and M. Yung, "Proactive secret sharing or: How to cope with perpetual leakage," in Proc. 15th Ann. Int. Cryptology Conf., Aug. 1995, pp. 339–352.

[20] D. L. Wells, J. A. Blakeley, and C. W. Thompson, "Architecture of an open object-oriented database management system," IEEE Comput., vol. 25, no. 10, pp. 74–82, Oct. 1992.

[21] K. Nørv°ag, O. Sandst°a, and K. Bratbergsengen, "Concurrency control in distributed object oriented database systems," in Proc. 1st East-Eur. Symp. Adv. Databases Inform. Syst., Sep. 1997, pp. 32–32.

[22] R. Droms, Dynamic Host Configuration Protocol, Request for Comment RFC 2131, Internet Engineering Task Force, Network Working Group, Mar. 1991.

[23] K. Kent and M. Souppaya. (1992). Guide to Computer Security Log Management, NIST Special Publication 800-92 [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-92/SP800-92.pdf