VOL. 12, NO. 21, NOVEMBER 2017 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

DISTRIBUTED ENVIRONMENT: SECURITY MECHANISM DESIGN STRUCTURE AND SECURITY IMPLEMENTATION FOR DATA DISTRIBUTION

Jai Pratap Dixit[1], Neelendra Badal[2] and Syed Qamar Abbas[3]
[1]Department of Information Technology, Ambalika Institute of Management and Technology, Lucknow Uttar Pradesh, India
[2]Department of Computer Science and Engineering Kamla Nehru Institute of Technology, Sultanpur, Uttar Pradesh, India
[3]Department of Computer Science, Ambalika Institute of Management and Technology, Lucknow Uttar Pradesh, India
E-Mail: jpdixit.iiita@gmail.com

ABSTRACT

Distributed Environment is that can evolve their different behaviors based on their changes in data distribution area. In this paper, we discuss security mechanism design issues and propose security metrics issues also in the context of distributed environment. A key premise with design layouts of distributed environment is that in order to detect their changes, authentication and information must be collected by different approaches of monitoring in environment. How design approaches should be done, what steps should be monitored, and the impact of monitoring may have on the security mechanism of the design issues in target system need for carefully considered. Conversely, the impact of security mechanism design layouts on the securing of data distribution environment. We propose a different design issues in security metrics that can be used to quantify the impact of different monitoring on the distributed security mechanism issues of the target distributed environment.

Keywords: access authentication, security issues, cryptography, authentication.

INTRODUCTION

Now a days Security aspects in different Distributed Environment play important roll. It defines a way in which the each components of the distributed system interact with each component in such a way mapped with network [3] including client server model. Research has been significantly using different mechanisms with protecting the data with their clients. Partition Technique includes the DBMS redefinition. Data streams clustering technique are highly helpful to handle data and outlier detection.

Distributed system security [21-22] in term of different objectives of database models based on classification, access control, attacks, and system failures. Distributed system is multiple redundant within multiple devices and data transferring between devices with different channels. Cluster Security based on domain knowledge for certification of cluster vulnerabilities. Authentication, cryptographic techniques, access control [6] is many developments towards the generation of secure and trusted distribution environments.

A distributed computer system can be described as a collection of clients and servers communicating by exchange of messages.

- System is running in an open environment
- Need to communicate with other heterogeneous systems

Security issues: Access authorization, one must be authorized to access the distributed computer system: Message security [2].

Passing message confidentially between nodes using some cryptographic techniques: Mutual authentication [4]

Two way authentication, both parties authenticate each other's identity suitably

Categories of access authorization

- Discretionary access control
- Access control matrix (ACM)
- Implemented via access control list or capability list or both.
- Mandatory access control[11]
- Represented as information flow among communicating entities.

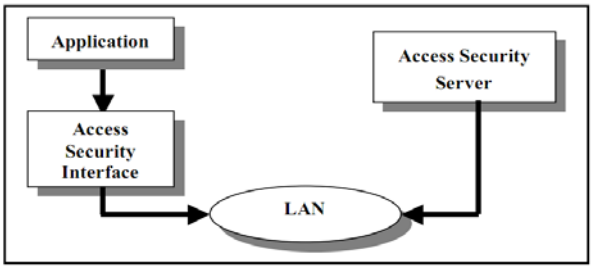

Figure-1. Access procedure.

The access security system software may operate at each of the network stations as an independent application (Figure-1**)**. The various applications will receive the access security services [7] via an access security interface which must be incorporated in each application where these services are required.

SECURITY MECHANISM

There are the following Security Mechanism used in distributed Environment.

- Encryption
- Authentication
- Authorization
- Auditing

Design based on platform architecture Design an architecture as a platform

- to support different security mechanisms
- For a carrier class internet server running on a *clustered* system.

Providing mechanisms to protect the system against:

- External attacks: originating from Internet,
- Internal attacks: attacks originating from Intranet.

Providing mechanisms for efficient
- Detection
- Reaction

GLOBUS SECURITY ARCHITECTURE

Global Architecture explains the overall security aspects within distributed system (Figure-2.)

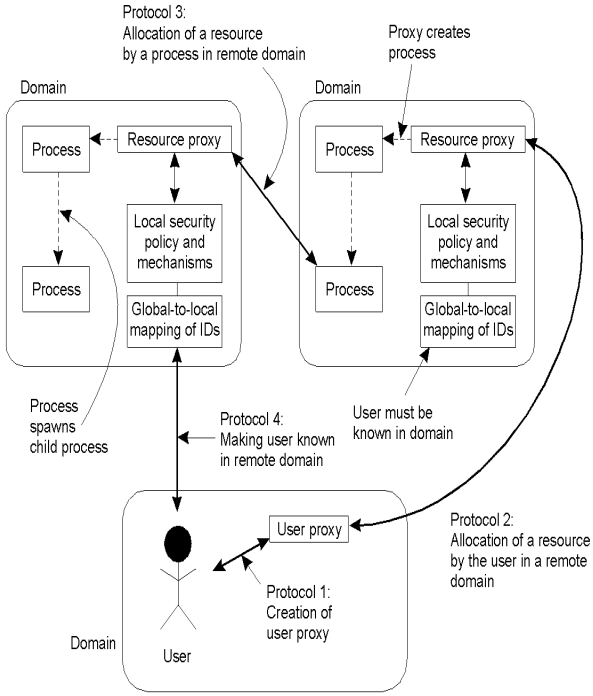

Figure-2. The Globus security architecture.

The environment consists of multiple administrative domains.
- Local operations are subject to a local domain security policy only [10].
- Global operations require the initiator to be known in each domain where the operation is carried out.
- Operations between entities in different domains require mutual authentication [11].
- Global authentication replaces local authentication.
- Controlling access to resources is subject to local security only [8].
- Users can delegate rights to processes.
- A group of processes in the same domain can share credentials.

Problems facing designers of distributed systems:

Widely varying modes of use: The system components are subject to wide variations in workload [9] (e.g. some web pages have millions of hits a day and some may have no hits). Some applications have special requirements for high communication bandwidth and low latency. Wide range of system environments: A distributed system must accommodate heterogeneous hardware, operating systems, and networks [8-15] (wireless networks operate at a fraction of the capacity and much higher error rates than present day LANs). Internal problems: Non-synchronized clocks, concurrency problems, many modes of hardware and software failures involving the individual components of the system. [23] External threats: Attacks on data integrity, ensuring confidentiality, denial of service.

DISTRIBUTED SYSTEM ARCHITECTURE

In a typical application, the server is concurrent and can handle several clients simultaneously. In [29] Servers may in turn be clients of other servers. For e.g. a web browser (client) may contact a web server, which invokes a servlet that communicates with a database server (may be Oracle or an LDAP server). Another example may be a client that communicates with an application server which communicates with a database server.

Services provided by multiple servers

Services may be implemented as several server processes in separate host computers interacting as necessary to provide a service to client processes. The data on which the service is based may be partitioned among the servers or each server may maintain replicated copies of the data. E.g. the web is an example of partitioned data where each web server manages its own set of web pages. Replication is used to increase performance and reliability [1] to improve fault-tolerance. It provides multiple consistent copies of data on different servers. e.g. the web service provided at altavista.digital.com is mapped onto several servers that have the database replicated in memory.

Distributed security is hard to achieve

Many layers to fit together: applications, middleware, OS, hardware, network Exposed by nature, Heterogeneous environment: Variety of Hardware, Software: OS, Middleware, Networking technologies

Challenges in distributed security

Implement coherent distributed security: Many layers to fit together: Applications, Middleware, OS, Hardware, Network, Heterogeneous environment: variety of Hardware, Software: OS, Middleware, Networking technologies, Integration of different security solutions: System management, manually managed, it is the open door for misconfigurations and inconsistencies

Need for a new security approach

No real time applications, Security policy based upon login and passwords, Running for short period of time (days) before each reboot, No pre-emptive security.

Why the need for a security framework?

Abstracting the underlying security algorithms and mechanisms

- Reducing development time
- Minimizing the risk of creating subtle, but dangerous security vulnerabilities by reusing security tested software

Maximize investment on developing security mechanisms

Access control approach on cluster computing

Current security approach in cluster computing:
- Generally based on user privileges (login, password),
- Life time: a session of several hours,
- Scope: limited range of operations according to the application's nature.

Our target application
- One user only,
- Life time: months if not years,
- Scope: wide range of operations, from upgrading software to managing information in database.

VARIATION ON THE CLIENT SERVER MODEL

Network computers

Eliminate the need for storing the operating system and application software on desktop PCs and instead download these from a remote file server. Applications are run locally but the files are managed by a remote file server. Since all the application data and code is stored by a file server, users may migrate from one network computer to another. The processor and memory capacities of a network computer can be constrained in order to reduce its cost. If a disk is provided, it holds only a minimum of software. The remainder of the disk is used as cache storage holding copies of software and data files recently downloaded from servers. The falling PC prices have probably rendered the network computer a non-starter.

Thin clients

Thin client refers to a layer of software that supports a window-based GUI on the local computer while executing application programs on a remote computer. This architecture has the same low management and hardware costs as the network computer, but instead of downloading application code into the user's computer, it runs them on a compute server - a powerful computer (typically a multiprocessor or a cluster computer) that has the processing power to run several applications concurrently.

DESIGN REQUIREMENT FOR DISTRIBUTED SYSTEM

Performance issues

Responsiveness: Interactive apps require a fast and consistent response. The speed at which the response is obtained is determined not just by the server and network load and performance, but also by the delays in all the software components involved, i.e., the operating system, the middleware services (such as remote method invocation support like naming) and the application code itself providing the service.

Throughput

This is the rate at which computational work is done (number of users serviced per second) and is affected by the processing speeds and at clients and servers and by data transfer rates.

Balancing computational loads

On heavily loaded servers it is necessary to use several servers to host a single service and to offload work (e.g. an applet in the case of a web server [8-9-11]) to the client where feasible.

Processes in a distributed system (e.g. client-side and server-side processes) interact with each other by passing messages, resulting in communication (message passing [14-16] and coordination (synchronization and ordering of activities [27]) between processes. Each process has its own state. There are two significant factors affecting process interaction in distributed systems such as Communication performance [10] is often a limiting characteristic and no single global notion of time since clocks on different computers tend to drift.

Three approaches for protection against security threats.

- Protection against invalid operations
- Protection against unauthorized invocations
- Protection against unauthorized users

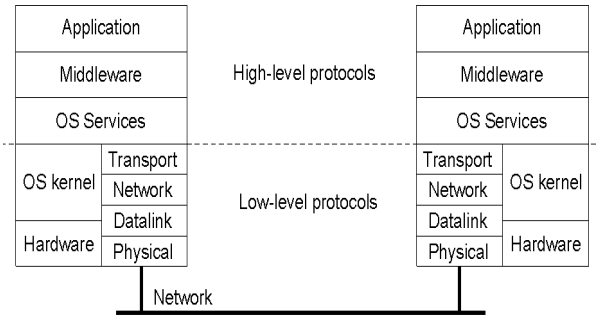

Figure-3. The logical organization of a distributed system into several layers.

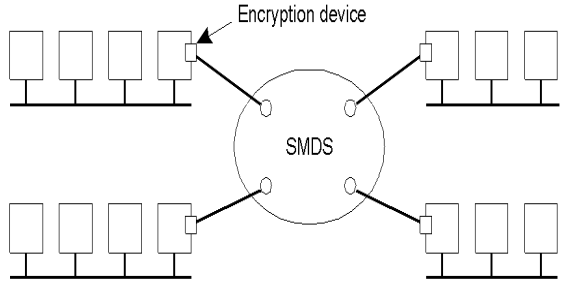

Figure-4. Several sites connected through a wide-area backbone service.

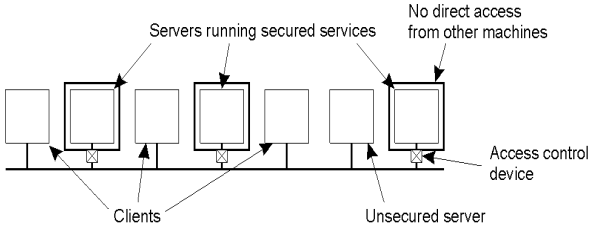

Figure-5. The principle of server running secure services.

SECURITY MECHANISM APPROACHES IN DISTRIBUTED SECURITY

A cryptographic technique uses private key and public key systems. Both provide integrity and authenticity of messages in addition to secrecy, symmetric encryption: private key, asymmetric encryption: private key & public key, Implemented using intricate algorithm like MD5, AEDS, and DES. [12-13]

Distributed authentication protocol: Maintain three basic properties: authenticity, integrity, and freshness.

Mutual authentication protocol: Characterized by whether a third-party authentication server is assumed and by how the freshness of messages is guaranteed.

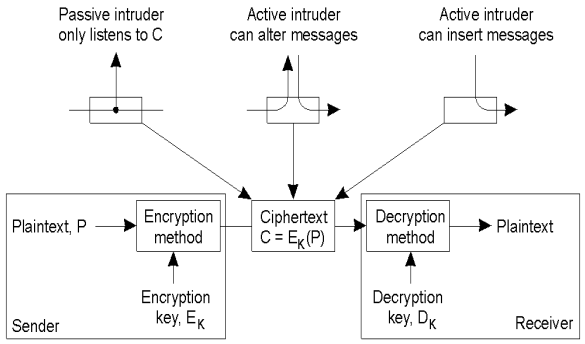

Figure-6. Intruders and eavesdroppers in communication.

ALGORITHM

In this approach we use updated algorithm of Advanced Encryption Standard (AES), which use specification for the encryption of electronic data, Triple Data Encryption Algorithm (TDEA or Triple DEA), become a key block, applies the Data Encryption Standard (DES) algorithm and The MD5 algorithm is used hash function producing a 128-bit hash value. It can still be used as a checksum to verify data integrity for implementation [30].

AES ALGORITHM PROCEDURE

- For encryption and decryption of attached documents and files.
- Firstly the original text i.e. clear text is converted into bytes and then for the AES algorithm to perform encryption, we need to generate Key and IV using the derived bytes and the symmetric key.
- Using Memory Stream and Crypto Stream the clear text is encrypted and written to byte array and finally the byte array is converted to Base64String and returned which is the final outcome i.e. the corresponding encrypted text.
- After that the encrypted text i.e. ciphers text is converted into bytes and then similar to the encryption process here too we will generate Key and IV using the derived bytes and the symmetric key.
- Using Memory Stream and Crypto Stream the cipher text is decrypted and written to byte array and finally the byte array is converted to Base64String and returned, which is the decrypted original text.

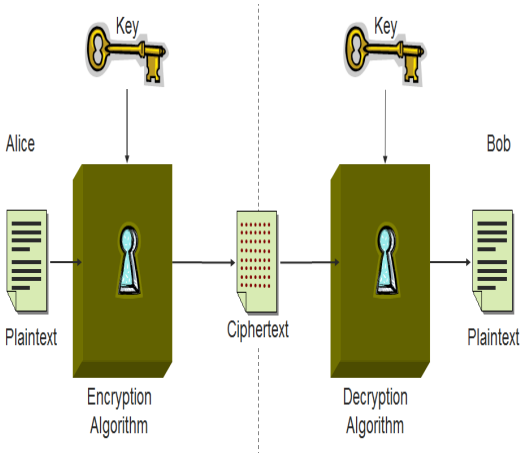

Figure-6(a). Encryption & decryption key process.

MD5 Algorithm procedure

- Hashing the encryption and decryption key using MD5.
- Used MD5 hash generator as the result is a 128 bit byte array which is a valid length for the TripleDES encoder.
- Hash functions map binary strings of an arbitrary length to small binary strings of a fixed length. Cryptographic # function has the property which state that computationally infeasible to find two distinct inputs that #function to the same value; that is, hashes of two sets of data should match if the corresponding data also matches. Small changes to the data result in large, unpredictable changes in the hash.
- The hash size for the MD5 algorithm is 128 bits.
- The Compute Hash methods of the MD5 class return the hash as an array of 16 bytes. Some MD5 implementations provide 32-character, hexadecimal / formatted # function. To interoperate with such implementations, format the return value of the Compute Hash methods as a hexadecimal value.

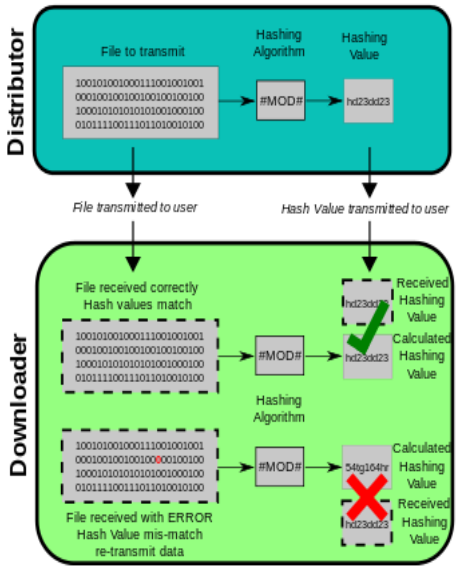

Figure-6(b). MD 5 key management.

TripleDES algorithm procedure

- Used to encode and decode the message string.
- TripleDES uses three successive iterations of the DES algorithm. It can use either 2 or 3- 56-bit keys.
- A newer symmetric encryption algorithm, Advanced Encryption Standard (AES), is available. Consider using the AES class and its derived classes instead of the TripleDES class [17] Use TripleDES only for compatibility with legacy applications and data.
- This algo supports to key lengths from 128 bits to 192 bits in increments of 64 bits pattern.
- Decryption and Encryption can be handled in the same way; use Create Decryptor instead of Create Encryptor. The same key (Key) and initialization vector used to encrypt the file must be used to decrypt it.
- Triple DES uses a "key bundle" that comprises three DES keys, K_1, K_2 and K_3, each of 56 bits (excluding parity bits). The encryption algorithm is:
- ciphertext = $E_{K3}(D_{K2}(E_{K1}(\text{plaintext})))$
- I.e., DES encrypts with K_1, DES *decrypt* with K_2, then DES encrypt with K_3.
- Decryption is the reverse:
- plaintext = $D_{K1}(E_{K2}(D_{K3}(\text{ciphertext})))$
- I.e., decrypt with K_3, *encrypt* with K_2, and then decrypt with K_1.
- Each triple encryption encrypts one block of 64 bits of data.
- In each case the middle operation is the reverse of the first and last. This improves the strength of the algorithm when using keying option 2, and provides backward compatibility with DES with keying option 3.

Key Procedure

- The standards define three keying options:
- **Keying option 1::** All three keys are independent.
- **Keying option 2::** K_1 and K_2 are independent, and $K_3 = K_1$.
- **Keying option 3::** All three keys are identical, i.e. $K_1 = K_2 = K_3$.
- Keying option 1 is the strongest, with $3 \times 56 = 168$ independent key bits.
- Keying option 2 provides less security, with $2 \times 56 = 112$ key bits. This option is stronger than simply DES encrypting twice, e.g. with K_1 and K_2, because it protects against meet-in-the-middle attacks.
- Keying option 3 is equivalent to DES, with only 56 key bits. It provides backward compatibility with DES, because the first and second DES operations cancel out
- Each DES key is nominally stored or transmitted as 8 bytes, each of odd parity,[13] so a key bundle requires 24 bytes for option 1, 16 for option 2, or 8 for option 3.

Generating the private and public key requires four steps:

a) Choose two very large prime numbers, *p* and *q*
b) Compute $n = p \times q$ and $z = (p - 1) \times (q - 1)$
c) Choose a number *d* that is relatively prime to *z*
d) Compute the number *e* such that $e \times d = 1 \bmod z$

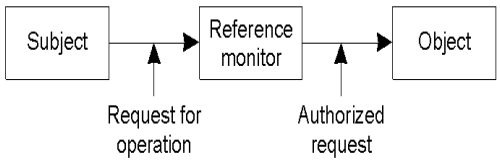

Figure-7. General model of controlling access to objects.

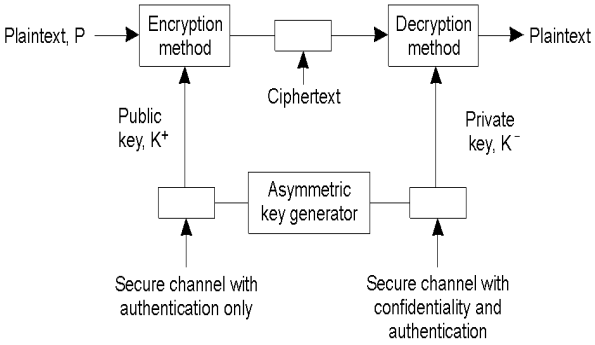

Figure-8. Public-key distributions.

FRAME WORK FOR SECURITY MECHANISM ISSUES

Design an algorithm for performing the security-mechanism at data partitioned level and transformation of partitioned data. The algorithm should be such that an opponent cannot defeat its purpose. We design a simple schema, tables indexes, constrains with using redefinition procedure [28]. We create different data sets and Outliers detection which helps in clustering for providing new optimistic results

Security mechanisms in middle ware for data distribution and it's during data modification and data transformation. Generate the modification approaches secret information to be used within the algorithm for securing the data.

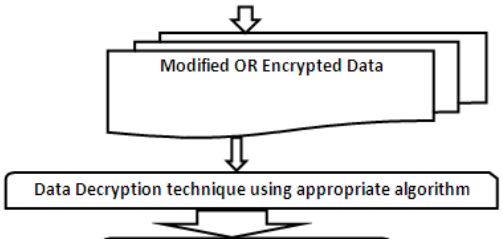

Figure-9. Data modification security technique.

Application of data web services [24] in secure manner and it's Monitoring in distributed environment. Develop methods for the distribution and sharing of the secret information.

After modification technique, the data could be used for data mining procedure in distributed environment [5] also. And it is very easy to obtain the original data after modification. In the proposed work, the data transformation technique is used for appropriate numerical attributes. Data protection is based on altered or manipulated technique so that data remain even after the procedure.

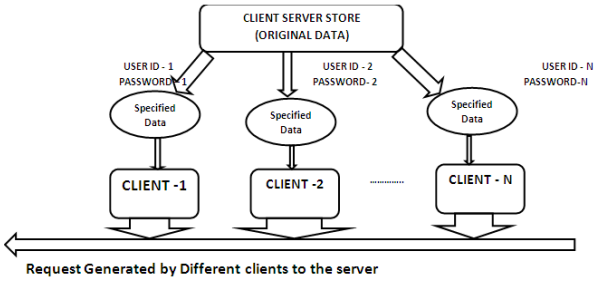

Figure-10. Client site security architecture.

Specify a mechanism for the client, used by the different principals who make use of the security algorithm for secret information to obtain a particular security mechanism metrics [19-21].

A useful means of classifying security mechanism in terms of event detection, security recovery, authentication, access control, confidentially and Integrity services [18-20-22].

However, it is feasible to prevent the success of this security [26] mechanism, usually by means of encryption (in fig. no-11). Thus, the emphasis in dealing with passive decryption is on prevention rather than detection.

RESULT

Implementation Work is thesis are shown as snaps shots. This research implementation work as online portal www.jnsdistributedsecurity.com., with user panel and admin panel for data distribution in secure manner.

There are the following steps implemented in Research work.

For admin Login ID jpdixit.iiita@gmail.com Login password: *****, For NEW USERS @ FIRST CREATE AN ACCOUNT AT THIS PORTAL

1. When user created a account they cannot login without admin approval. Firstly admin approved it than user can login in system.
2. At the time of use registration user decide two passwords, one is for login, second one for accessing file and email data.
3. When admin approved users can access the data distribution as secure manner.
4. Admin can create a Separate security password for data.
5. All file converted into ZIP format.
6. Users send any data to registered users only.
7. Encryption and Decryption algorithm updated MD5 & TrippleDES and AES Algorithms
8. Admin can only show the encrypted data
9. Email selection only via check box
10. Trying to security of the documents as time duration as particular date
11. No one can access documents on server

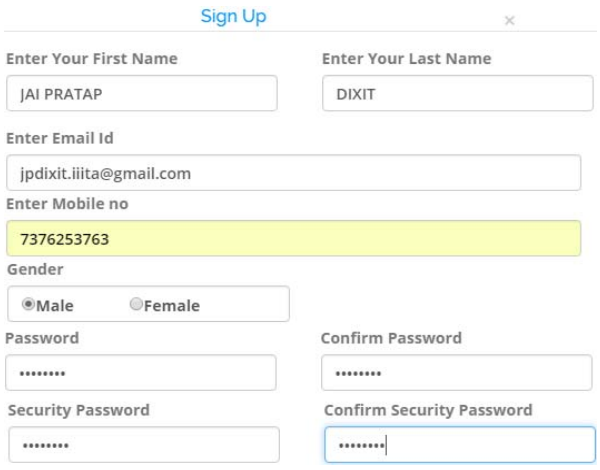

Figure-11. Registration form.

Figure-12 discribe the login process with different security password one is used for login and others one is used for the secureencrypted file accessing.

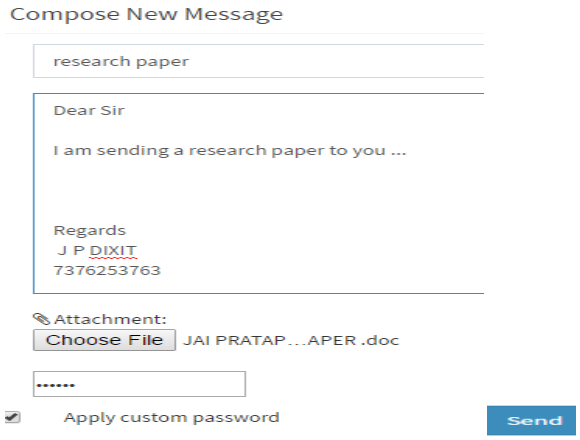

Figure-12. Message writing.

Figure-13 Message writingwith secure password (Figure-12) refers to the custom password by admin..

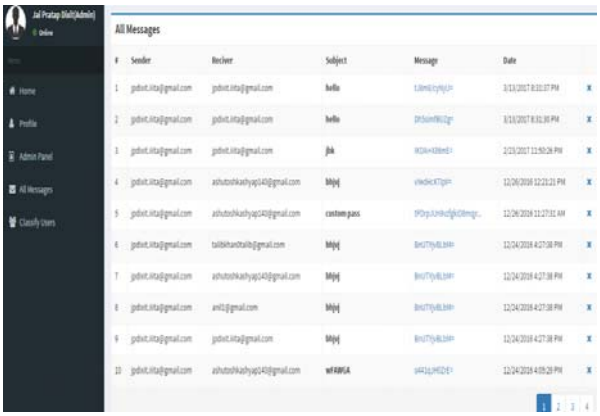

Figure-13. Message description details.

Encrypted message data Figure-14 are recorded by algorithm automatically. Which cannot be accessed by server hackers?

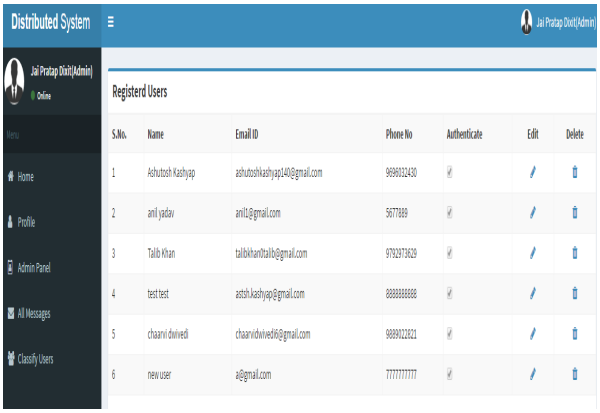

Figure-14. Admin panel for user authentication.

Via admin panel admin can authenticate the person in the system for authorized users (Figure-15).

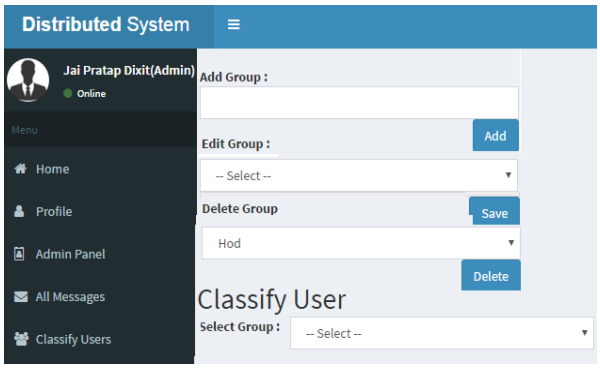

Figure-15. Group creation in security system.

Admin can classified the different users as category with in distributed system (Figure-16) Users sending a message

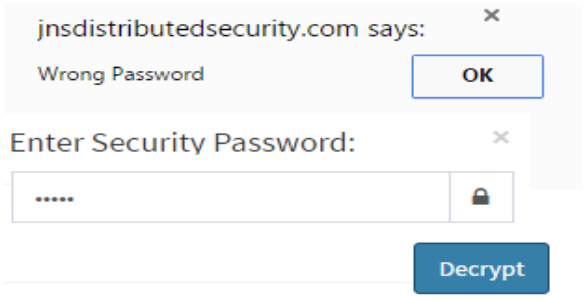

Figure-16. Security password verification.

When user opens a mail then it asked for a password for decryption of messages as Figure-17.

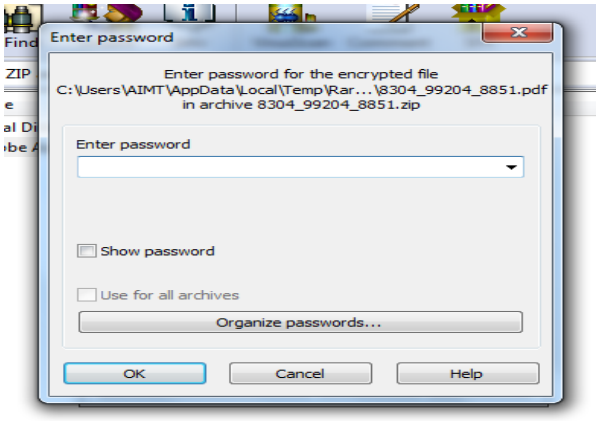

Figure-17. Security password.

In above diagram (Figure-18) all encrypted file become converted into a zip format and also again asked for the secure password for accessing a file.

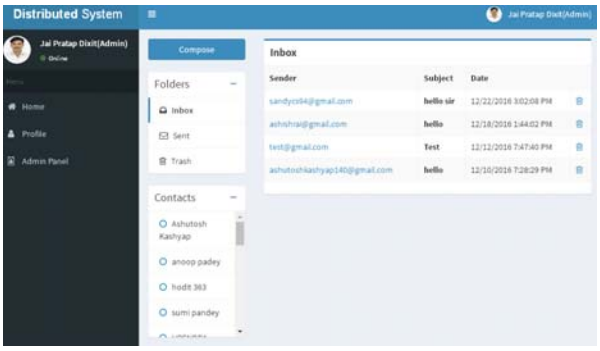

Figure-18. Admin pannel.

For sending a mail client choose a appropriate mails for data distribution as Figure-19.

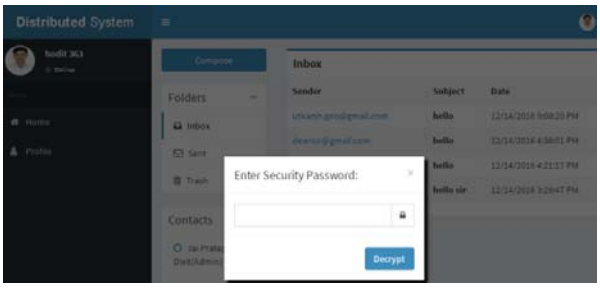

Figure-19. Security check encryption.

A secure password also asked at the time of email opening for message as above diagram Figure-20.

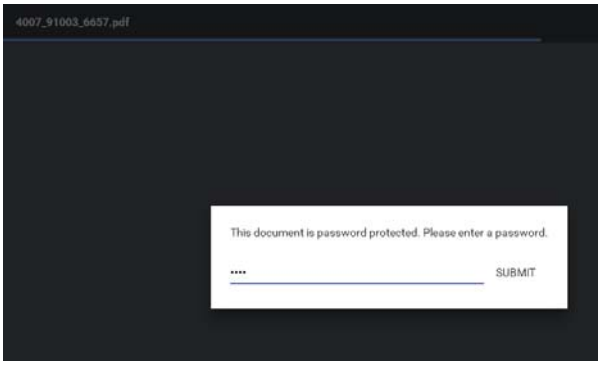

Figure-20. Security check password.

This Figure-21 File accessing secure password for particular file. it may be users or admin password as per sending critiera for data distribution for secure communication.

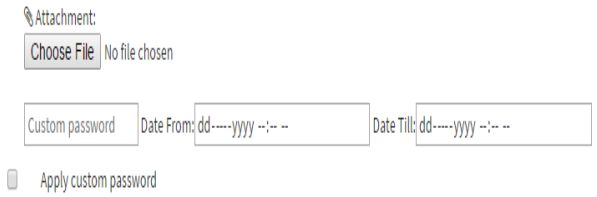

Figure-21. Time slot base secure mechanism.

Data can be secure by the time duration access for data distribution. It provide by the admin of the system Figure-21.

Conclusion and future scope

Here we Implemented the proposed research work design scheme based on the a novel approach of data distribution security mechanism in distributed environment at data partitioned level and its transformation of partitioned data in secure manner, middle ware security mechanism in data distribution during data modification and data transformation using efficient algorithm in security system for implementation. And also provide a proposal for application of data web services in secure manner and also it's monitoring in distributed environment.

The previous security mechanism was discussed the security challenges will be implemented in future using appropriate security mechanism.

There is little security support as a coherent solution into distributed applications developed for clustered servers.

Current encryption algorithm are sometimes expensive to execute and can be decrypted with some known information. A more efficient and secure algorithm is needed to achieve the security goal.

Techniques to identify the eavesdropper in the communication network quickly and correctly. The intelligent agent is a good practice whereas it must introduce a third party monitor to the distributed computing system. And that monitor can be a potential secure issue in terms of system attack.

As cloud computing has been widely used in recent years, the security issue becomes a harder problem since more computers in scattered locations join the system. Challenging work is proposed to ensure the safe communication among these endpoints. Data can be secure by the time duration access for data distribution. It provide by the admin of the system.

ACKNOWLEDGEMENT

I acknowledge my great gratitude and immense respect to Dr. Neelendra Badal [Assistant Professor- Department of CSE, Kamla Nehru National Institute of Technology, and Sultanpur] and Dr. S Q Abbas [Professor- Department of CSE, Ambalika Institute of Management and Technology] for their encouragement, inspiration and insightful suggestions. I would like to gratitude to Dr Alok Mishra for their valuable support.

REFERENCES

[1] H. Hamdi, M. Mosbah. 2009. A DSL framework for policy based Security of distributed systems. 3rd IEEE International Conference on Secure Software Integration and Reliability Improvements. pp. 150-158. Article (CrossRef Link).

[2] Adi Armoni. 2002. Data Security Management in Distributed Computer Systems. Informing Science. Vol. 5. Article (CrossRef Link).

[3] K. Boudaoud; N. Agoulmine; J.N De Souza. 2004. Distributed Network Security Management Using Intelligent Agents. Article (CrossRef Link).

[4] Mirtaheri S.L, Khaneghah E.M, Sharifi M, Azgomi M.A. 2008. The influence of efficient message passing mechanisms on high performance distributed scientific computing. Parallel and distributed Processing with Applications IEEE: 663-668. Article (CrossRef Link).

[5] Naqvi S.; Riguidel M. 2004. Security architecture for heterogeneous distributed Computing systems. Security technology, 38th international conference, IEEE Explore. Article (CrossRef Link).

[6] Edara U.R.; Subramanian N.; Dwivedi M.; Sinha A. 2010. A system for security assessment grid environment. 2010 IEEE 4th international conference. pp. 1-6. Article (CrossRef Link).

[7] Vijayarani Mining, Dr.A. Tamilarasi. 2010. Data Transformation Technique for Protecting Private Information in Privacy Preserving Data Mining. Advanced Computing: An International Journal (ACIJ). 1(1). Article (CrossRef Link)

[8] Domingo-Ferrer J & Mateo-Sanz J. M. 2002. Practical data-Oriented micro aggregation for statistical disclosure control. IEEE Transactions on Knowledge and Data Engineering. 14(1): 189-201, 2002.Article (CrossRef Link).

[9] Samarati P. 2001. Protecting respondents' identities in Microdata release. IEEE Transactions on Knowledge and Data Engineering. 13(6): 1010-1027. 2001.Article (CrossRef Link).

[10] Mayur Sawant, Kishor Kinage, Pooja Pilankar, Nikhil Chaudhari. 2013. Database Partitioning: A Review Paper. International Journal of Innovative Technology

and Engineering (IJITEE) ISSN: 2278-3075, 3(5). Article (CrossRef Link).

[11] Wen Qi, Jie Song and Yu-bin Bao. 2010. Near-uniform Range Partition Approach for Increased Partitioning in Large Database. IEEE, 978-1-4244-5265-1/10. Article (CrossRef Link).

[12] Jie Song and Yubin Bao. 2010. NPA: Increased Partitioning Approach for Massive Data in Real-time Data Warehouse. IEEE, 978-1-4244-7585-8/10. Article (CrossRef Link).

[13] Eugene Wu and Samuel Madden. 2011. Partitioning Techniques for Fine-grained Indexing. IEEE 978-1 - 4244-8960-2/11, Article (CrossRef Link).

[14] Dr. S. Vijayarani1, Ms. P. Jothi. 2014. Hierarchical and Partitioning Clustering Algorithms for Detecting Outliers in Data Streams. International Journal of Advanced Research in Computer and Communication Engineering, ISSN 2278-1021 /: 2319-5940. 3(4). Article (CrossRef Link).

[15] Yogita Durga Toshniwal. 2012. Clustering Techniques for Streaming Data-A Survey in proc. of the IEEE. Article (CrossRef Link).

[16] Dr. S. Vijayarani, Ms.P. Jothi. 2014. Partitioning Clustering Algorithms for Data Stream Outlier Detection. International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization). 2(4). Article (CrossRef Link).

[17] Sudipto Guha, Adam Meyerson, Nine Mishra and Rajeev Motwani. 2003. Clustering Data Streams: Theory and practice. IEEE Transactions on Knowledge and Data Engineering. 15(3): 515-528. Article (CrossRef Link).

[18] George Coulouris, Jean Dollimore and Tim Kindberg. 2005. Distributed System-Concept and Design. 4th ed. London England: Addison-Wesley. Article (CrossRef Link).

[19] Andrew S Tenenbaum and Maarten van Steen. 2007. Distributed System: Principle and Paradigms. 2nd ed. Upper Saddle River, NJ, USA: Pearson Higher Education. Article (CrossRef Link).

[20] Firdhous. 2011. Implementation of Security in Distributed system-A Comparative Study. International Journal of computer System. 2(2). Article (CrossRef Link).

[21] Yuchong Hu, Yinlong Xu, Xiaozhao Wang, Cheng Zhan, and Pei Li. 2010. Cooprative Recovery of Distributed Storage System From Multiple Losses with network Coding. IEEE Journal on Selected Area in Communications. 28(2): 268-267. Article (CrossRef Link).

[22] Wei Li and Rayford B Vaughn. 2006. Cluster Security Research Involving the Modeling of Network Exploitations Using Exploitation Graphs. In: 6th IEEE International Symposium on Cluster Computing and the Grid Workshop, Singapore. pp. 26-36. Article (CrossRef Link).

[23] Ragib Hassan Suvda Myagmar, Adam J Lee, and William Yurcik. 2005. Toward a threat model for storage system. In: Proceeding of the 2005 ACM Workshop on storage security and Survivability, USA. Article (CrossRef Link).

[24] Theodoros K Dikaliotis, Alexandros G Dimakis and Tracey Ho. 2010. Security in Distributed storage system by communicating a logarithmic number of bits. in IEEE, ISIT, Austin, TX, USA. pp. 1948-1952 Article (CrossRef Link).

[25] Vijay Prakash, Manuj Darbari. 2012. A Review on Security Issues in Distributed Systems. International Journal of Scientific & Engineering Research. 3(9): 1ISSN 2229-5518 Article (CrossRef Link).

[26] M. Shehab, A. Ghafoor, E. Bertino. 2010. Secure collaboration in a Mediator free distributed environment. IEEE Transactions on Parallel and Distributed Systems. 19(10): 1338-1351, 2010.Article (CrossRef Link).

[27] S. Pallickara, J. Ekanayake, G. Fox. 2007. A scalable approach for the secure and authorized tracking of the availability of Entities in distributed systems. IEEE International Parallel and Distributed Processing symposium. pp. 1-10. Article (CrossRef Link).

[28] T. Xiaoyong, K. Li, Z. Zong, B. Veradale. 2011. A novel security-Driven scheduling algorithms for precedence-constrained tasks in heterogeneous distributed systems. IEEE Transactions on Computers. 60(7): 1017-1029. Article (CrossRef Link).

[29] X. Huang, Y. Xiang, A. Chonka, J. Zhou, R.H Deng. 2011. A Generic framework for three factor authentication: Preserving Security and privacy in distributed systems. IEEE Transactions on Parallel and Distributed Systems. 222(8): 1390-1397.Article (CrossRef Link).

[30] Dixit J .P. Badal Neelendra, Abbas S.Q. 2017. A Novel Approach: Distributed Security Mechanism of Data Distribution in Distributed Environment. In International Journal of Applied Engineering Research ISSN 0973-4562 (Accepted). pp. 1460-1468.