ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

A Survey on Emergence of Citation Network using Knowledge Core

M.Kirithika Devi, Dr. R.Kavitha

Assistant Professor, Department of Computer Science and Engineering, Chendhuran College of Engineering and Technology, Lena Vilakku, Pilivalam (po), Pudukkottai, Tamil Nadu, India

Professor, Department of Computer Science and Engineering, Velammal College of Engineering and Technology, Madurai to Rameshwaram High Road, Viraganoor, Madurai, Tamil Nadu, India

ABSTRACT: The major aims of this method a secure multi-owner data sharing scheme. It implies that any user in the group can securely share data with others by the un trusted cloud. This scheme is able to support dynamic groups. Efficiently, specifically, new granted users can directly decrypt data files uploaded before their participation without contacting with data owners. User revocation can be easily achieved through a novel revocation list without updating the secret Keys of the remaining users. The size and computation overhead of encryption are constant and We present a secure and privacy-preserving access control to users, which guarantee any member in a group to anonymously utilize the cloud resource. Moreover, the real identities of data owners can be revealed by the group manager when disputes occur. We provide rigorous security analysis, and perform extensive simulations to demonstrate the efficiency of our scheme in terms of storage and computation overhead. Cloud computing provides an economical and efficient solution for sharing group resource among cloud users. Unfortunately, sharing data in a multi-owner manner while preserving data and identity privacy from an untrusted cloud is still a challenging issue, due to the frequent change of the membership.

I. INTRODUCTION

AN enduring empirical observation on the connectivity among intellectual artifacts (e.g. patents, academic articles, etc.) is that, described as networks, they often hold a core-periphery structure – A group of densely interconnected artifacts contains the *core* knowledge to the field, which is commonly cited by *peripheral* artifacts.

This article is primarily interested in explaining how the core-periphery structure emerges over time in citation networks. A citation network is viewed as collection of artifacts created by individual authors in a decentralized manner. In creating an artifact, each author both makes citations to existing artifacts and creates new knowledge in the present artifact, in order to maximize their own utility. Both citation and knowledge creation are costly activities to the author. We furthermore incorporate the heterogeneity of artifacts: Knowledge stored in an existing artifact is only partially useful for creating a new artifact. In this line, our analysis reveals an explicit relation between artifact heterogeneity and the characteristics of knowledge core.
After all, our research delivers two streams of insights on the growth of citation networks:

- *Network structure*: What is the architecture of the citation network formed by decentralized, self interested knowledge contributions?
- *Knowledge creation*: How much knowledge does one create and how does the level of knowledge creation change over time? As an answer to the first question, our model produces a *core-periphery* landscape for the citation network. As for the second question, we characterize the optimal individual knowledge creation as a decreasing function of time. Compared to conventional statistics-based modeling of Networks.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

II. EXISTING SYSTEM

Several security schemes for data sharing on untrusted servers have been proposed. In these approaches, data owners store the encrypted data files in untrusted storage and distribute the corresponding decryption keys only to authorized users. Thus, unauthorized users as well as storage servers cannot learn the content of the data files because they have no knowledge of the decryption keys However, the complexities of user participation and revocation in these schemes are linearly increasing with the number of data owners and the number of revoked users, respectively. By setting a group with a single attribute, Lu et al. proposed a secure provenance scheme based on the cipher text-policy attribute-based encryption technique, which allows any member in a group to share data with others. However, the issue of user revocation is not addressed in their scheme. presented a scalable and fine-grained data access control scheme in cloud computing based on the key policy attribute-based encryption (KP-ABE) technique. Unfortunately, the single owner manner hinders the adoption of their scheme into the case where any user is granted to store and share data.

III. PROPOSED SYSTEM

This paper, we propose a secure multi owner data sharing scheme, named Mona, for dynamic groups in the cloud. By leveraging group signature and dynamic broadcast encryption techniques, any cloud user can anonymously share data with others. Meanwhile, the storage overhead and encryption computation cost of our scheme are independent with the number of revoked users. In addition, we analyze the security of our scheme with rigorous proofs, and demonstrate the efficiency of our scheme in experiments.

ADVANTAGES:

- We propose a secure multi-owner data sharing scheme. It implies that any user in the group can securely share data with others by the untrusted cloud.
- We provide secure and privacy-preserving access control to users, which guarantees any member in a group to anonymously utilize the cloud resource.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

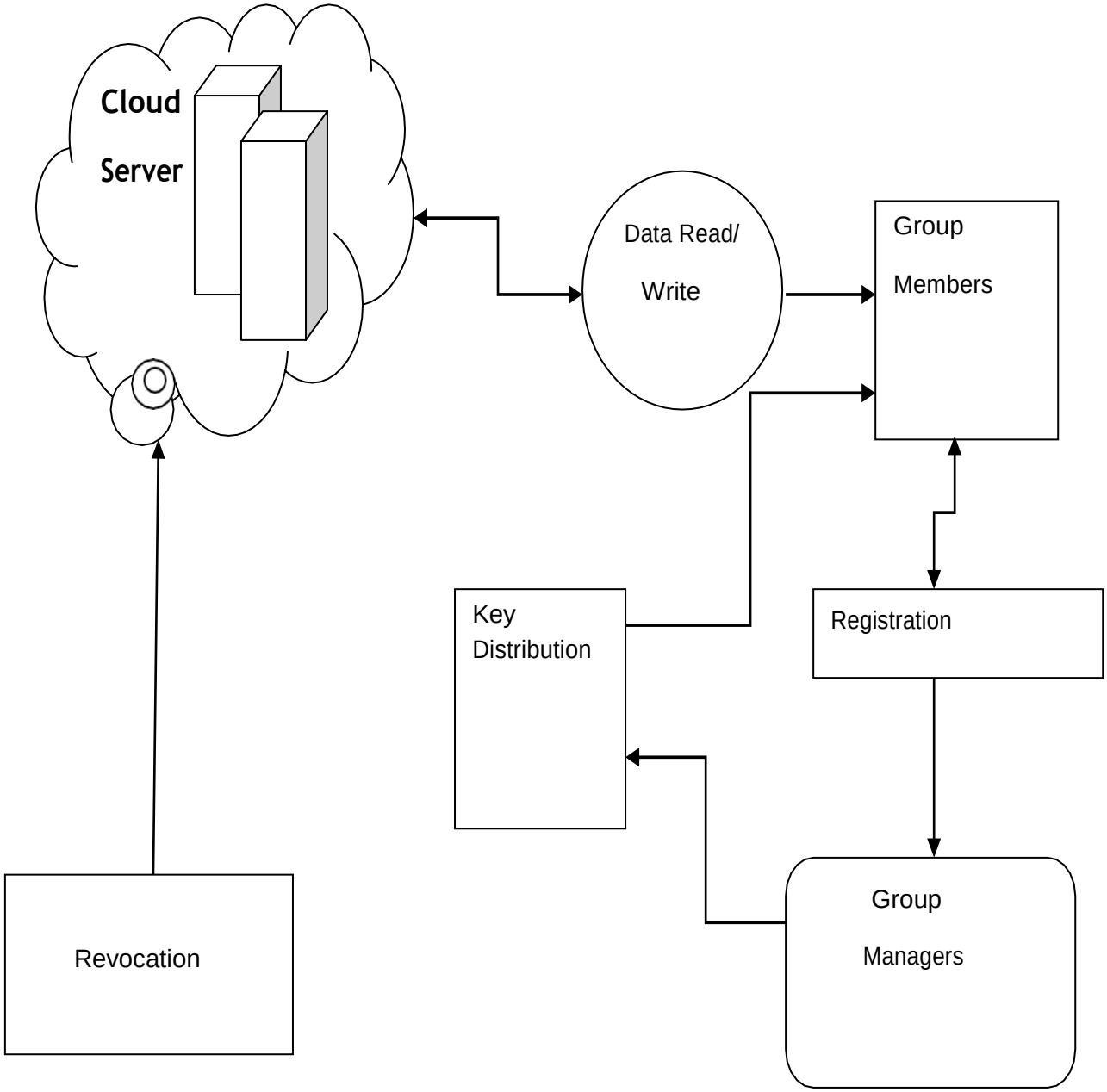
International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

SYSTEM ARCHITECTURE:



ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

IV. LITERATURE SURVEY

1) Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing
Shucheng Yu_, Cong Wang†, Kui Ren†, and Wenjing Lou_

This paper addresses this challenging open issue by, on one hand, defining and enforcing access policies based on data attributes, and, on the other hand, allowing the data owner to delegate most of the computation tasks involved in fine grained data access control to untrusted cloud servers without disclosing the underlying data contents. We achieve this goal by exploiting and uniquely combining techniques of attribute-based encryption (ABE), proxy re-encryption, and lazy re-encryption. Our proposed scheme also has salient properties of user access privilege confidentiality and user secret key accountability.

2) Plutus: Scalable secure file sharing on untrusted storage
MaheshSan Francisco, CA, USA
March 31–April 2, 2003

This paper has introduced novel uses of cryptographic primitives applied to the problem of secure storage in the presence of untrusted servers and a desire for owner managed key distribution. Eliminating almost all requirements for server trust (we still require servers not to destroy data – although we can detect if they do) and keeping key distribution (and therefore access control) in the hands of individual data owners provides a basis for a secure storage system that can protect and share data at very large scales and across trust boundaries.

3) SiRiUS: Securing Remote Untrusted Storage
Eu-Jin Goh_, Hovav Shacham†, Nagendra Modadugu, Dan Boneh‡ Stanford
University

This paper presents SiRiUS, a secure file system designed to be layered over insecure network and P2P file systems such as NFS, CIFS, OceanStore, and Yahoo! Briefcase. SiRiUS assumes the network storage is untrusted and provides its own read-write cryptographic access control for file level sharing. Key management and revocation is simple with minimal out-of- band communication. File system freshness guarantees are supported by SiRiUS using hash tree constructions. SiRiUS contains a novel method of performing file random access in a cryptographic file system without the use of a block server.

4) Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing
Rongxing Lu†, Xiaodong Lin‡, Xiaohui Liang†, and Xuemin (Sherman) Shen†

In this paper proposed scheme is characterized by providing the information confidentiality on sensitive documents stored in cloud, anonymous authentication on user access, and provenance tracking on disputed documents. With the provable security techniques, we formally demonstrate the proposed scheme is secure in the standard model.

5) Ciphertext-Policy Attribute-Based Encryption: An Expressive, E_cient, and Provably Secure Realization
Brent Waters _
University of Texas at Austin

This Paper presents a new methodology for realizing Ciphertext-Policy Attribute Encryption (CP- ABE) under concrete and non interactive cryptographic assumptions in the standard model. Our solutions allow any encryptor to specify access control in terms of any access formula over the attributes in the system. In our most e_cient system, ciphertext size, encryption, and decryption time scales linearly with the complexity of the access

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

formula. The only previous work to achieve these parameters was limited to a proof in the generic group model.

6) Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data
Vipul Goyal¤ Omkant Pandey*y* Amit Sahai*z* Brent Waters *x*

This Paper presents more sensitive data is shared and stored by third-party sites on the Internet, there will be a need to encrypt data stored at these sites. One drawback of encrypting data, is that it can be selectively shared only at a coarse-grained level (i.e., giving another party your private key). We develop a new cryptosystem for ¯ne-grained sharing of encrypted data that we call Key-Policy Attribute-Based Encryption (KP-ABE). In our cryptosystem, ciphertexts are labeled with sets of attributes and private keys are associated with access structures that control which ciphertexts a user is able to decrypt. We demonstrate the applicability of our construction to sharing of audit-log information and broadcast encryption.

7) Revocation and Tracing Schemes for Stateless Receivers ?
Dalit Naor1, Moni Naor2??, and Je® Lotspiech1

This paper provide a general traitor tracing mechanism that can be integrated with any Subset-Cover revocation scheme that satis¯es a \bifurcation property". This mechanism does not need an a priori bound on the number of traitors and does not expand the message length by much compared to the revocation of the same set of traitors.
The business model, IT as a service (ITaaS), is used by in-house, enterprise IT organizations that offer any or all of the above services.

Using software as a service, users also rent application software and databases. The cloud providers manage the infrastructure and platforms on which the applications run. End users access cloud-based applications through a web browser or a light-weight desktop or mobile app while the business software and user's data are stored on servers at a remote location. Proponents claim that cloud computing allows enterprises to get their applications up and running faster, with improved manageability and less maintenance, and enables IT to more rapidly adjust resources to meet fluctuating and unpredictable business demand.

- Autonomic computing — Computer systems capable of self-management.
- Client–server model — *Client–server computing* refers broadly to any distributed application that distinguishes between service providers (servers) and service requesters (clients).
- Grid computing — "A form of distributed and parallel computing, whereby a 'super and virtual computer' is composed of a cluster of networked, loosely coupled computers acting in concert to perform very large tasks."
- Mainframe computer — Powerful computers used mainly by large organizations for critical applications, typically bulk data processing such as census, industry and consumer statistics, police and secret intelligence services, enterprise resource planning, and financial transaction processing.
- Utility computing — The "packaging of computing resources, such as computation and storage, as a metered service similar to a traditional public utility, such as electricity."
- Peer-to-peer — Distributed architecture without the need for central coordination, with participants being at the same time both suppliers and consumers of resources (in contrast to the traditional client–server model).
- Cloud gaming - Also called On-demand gaming is a way of delivering to games to computers. The gaming data will be stored in the provider's server, so that gaming will be independent of client computers used to play the game.

Characteristics

- **Agility** improves with users' ability to re-provision technological infrastructure resources.

- **Application programming interface** (API) accessibility to software that enables machines to interact with

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

cloud software in the same way the user interface facilitates interaction between humans and computers. Cloud computing systems typically use REST-based APIs.

- **Cost** is claimed to be reduced and in a public cloud delivery model capital expenditure is converted to operational expenditure.

  This is purported to lower barriers to entry, as infrastructure is typically provided by a third-party and does not need to be purchased for one-time or infrequent intensive computing tasks. Pricing on a utility computing basis is fine-grained with usage-based options and fewer IT skills are required for implementation (in-house).The e-FISCAL project's state of the art repository contains several articles looking into cost aspects in more detail, most of them concluding that costs savings depend on the type of activities supported and the type of infrastructure available in-house.

- **Device and location independence** enable users to access systems using a web browser regardless of their location or what device they are using (e.g., PC, mobile phone). As infrastructure is off-site (typically provided by a third-party) and accessed via the Internet, users can connect from anywhere.
- **Virtualization** technology allows servers and storage devices to be shared and utilization be increased. Applications can be easily migrated from one physical server to another.
- **Multitenancy** enables sharing of resources and costs across a large pool of users thus allowing for:
  - **Centralization** of infrastructure in locations with lower costs (such as real estate, electricity, etc.)
  - **Peak-load capacity** increases (users need not engineer for highest possible load-levels)
  - **Utilization and efficiency** improvements for systems that are often only 10–20% utilized.
- **Reliability** is improved if multiple redundant sites are used, which makes well-designed cloud computing suitable for business continuity and disaster recovery.[30]
- **Scalability and elasticity** via dynamic ("on-demand") provisioning of resources on a fine- grained, self-service basis near real-time, without users having to engineer for peak loads.
- **Performance** is monitored, and consistent and loosely coupled architectures are constructed using web services as the system interface.
- **Security** could improve due to centralization of data, increased security-focused resources, etc., but concerns can persist about loss of control over certain sensitive data, and the lack of security for stored kernels. Security is often as good as or better than other traditional systems, in part because providers are able to devote resources to solving security issues that many customers cannot afford. However, the complexity of security is greatly increased when data is distributed over a wider area or greater number of devices and in multi-tenant systems that are being shared by unrelated users. In addition, user access to security audit logs may be difficult or impossible. Private cloud installations are in part motivated by users' desire to retain control over the infrastructure and avoid losing control of information security.
- **Maintenance** of cloud computing applications is easier, because they do not need to be installed on each user's computer and can be accessed from different places.

Virtualization

Virtualization (or **virtualisation**) is the creation of a virtual (rather than actual) version of something, such as a hardware platform, operating system (OS), storage device, or network resources.

While a **physical computer** in the classical sense is clearly a complete and actual machine, both *subjectively* (from the user's point of view) and *objectively* (from the hardware system administrator's point of view), a **virtual machine** is *subjectively* a complete machine (or very close), but *objectively* merely a set of files and running programs on an actual, physical machine (which the user need not necessarily be aware of).

Virtualization can be viewed as part of an overall trend in enterprise IT that includes autonomic computing,

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

a scenario in which the IT environment will be able to manage itself based on perceived activity, and utility computing, in which computer processing power is seen as a utility that clients can pay for only as needed.

The usual goal of virtualization is to centralize administrative tasks while improving scalability and overall hardware-resource utilization. With virtualization, several operating systems can be run in parallel on a single central processing unit (CPU). This parallelism tends to reduce overhead costs and differs from multitasking, which involves running several programs on the same OS.

ALGORITHM
AES:
Definition

The Advanced Encryption Standard (AES) is an encryption algorithm for securing sensitive but unclassified material by U.S. Government agencies and, as a likely consequence, may eventually become the de facto encryption standard for commercial transactions in the private sector. (Encryption for the US military and other classified communications is handled by separate, secret algorithms.)In January of 1997, a process was initiated by the National Institute of Standards and Technology (NIST), a unit of the U.S. Commerce Department, to find a more robust replacement for the Data Encryption Standard (DES) and to a lesser degree Triple DES. The specification called for a symmetric algorithm (same key for encryption and decryption) using block encryption (see block cipher) of 128 bits in size, supporting key sizes of 128, 192 and 256 bits, as a minimum. The algorithm was required to be royalty-free for use worldwide and offer security of a sufficient level to protect data for the next 20 to 30 years. It was to be easy to implement in hardware and software, as well as in restricted environments (for example, in a smart card) and offer good defenses against various attack techniques. The entire selection process was fully open to public scrutiny and comment, it being decided that full visibility would ensure the best possible analysis of the designs. In 1998, the NIST selected 15 candidates for the AES, which were then subject to preliminary analysis by the world cryptographic community, including the National Security Agency. On the basis of this, in August 1999, NIST selected five algorithms for more extensive analysis. These were:

- MARS, submitted by a large team from IBM Research
- RC6, submitted by RSA Security
- Rijndael, submitted by two Belgian cryptographers, Joan Daemon and Vincent Rijmen
- Serpent, submitted by Ross Andersen, Eli Biham and Lars Knudsen
- Two fish, submitted by a large team of researchers including Counterpane's respected cryptographer, Bruce Schneier

Implementations of all of the above were tested extensively in ANSI C and Java languages for speed and reliability in such measures as encryption and decryption speeds, key and algorithm set-up time and resistance to various attacks, both in hardware- and software-centric systems. Once again, detailed analysis was provided by the global cryptographic community (including some teams trying to break their own submissions). The end result was that on October 2, 2000, NIST announced that Rijndael had been selected as the proposed standard. On December 6, 2001, the Secretary of Commerce officially approved Federal Information Processing Standard (FIPS) 197, which specifies that all sensitive, unclassified documents will use Rijndael as the Advanced Encryption Standard. Also see cryptography, data recovery agent (DRA)**RELATED GLOSSARY TERMS:** RSA algorithm (Rivest-Shamir-Adleman), data key, greynet (or graynet), spam cocktail (or anti-spam cocktail), fingers canning (fingerprint scanning),munging, insider threat, authentication server, defense in depth, nonrepudiation

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

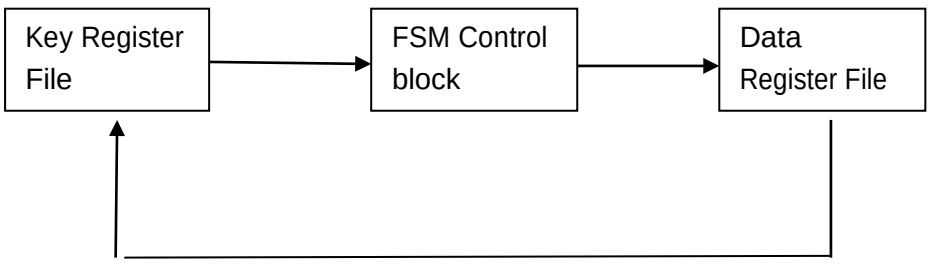
Explanations

AES is based on a design principle known as a Substitution permutation network. It is fast in both software and hardware. Unlike its predecessor, DES, AES does not use a Feistel network.AES has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits, whereas Rijndael's can be specified with block and key sizes in any multiple of 32 bits, with a minimum of 128 bits. The block size has a maximum of 256 bits, but the key size has no theoretical maximum.AES operates on a 4×4 column-major order matrix of bytes, termed the *state* (versions of Rijndael's with a larger block size have additional columns in the state). Most AES calculations are done in a special finite field. The AES cipher is specified as a number of repetitions of transformation rounds that convert the input plaintext into the final output of ciphertext. Each round consists of several processing steps, including one that depends on the encryption key. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same encryption key.

High-level description of the algorithm

1. Key Expansion—round keys are derived from the cipher key using Rijndael's key schedule
2. Initial Round
    1. AddRoundKey—each byte of the state is combined with the round key using bitwise xor
3. Rounds
    1. SubBytes—a non-linear substitution step where each byte is replaced with another according to alookup table.
    2. ShiftRows—a transposition step where each row of the state is shifted cyclically a certain number of steps.
    3. MixColumns—a mixing operation which operates on the columns of the state, combining the four bytes in each column.
    4. AddRoundKey
4. Final Round (no MixColumns)
    1. SubBytes
    2. ShiftRows
    3. AddRoundKey

Diagrams



ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

Examples

In this appendix, twenty examples are provided for the MAC generation process. The underlying block cipher is either the AES algorithm or TDEA. A block cipher key is fixed for each of the currently allowed key sizes, i.e., AES-128, AES-192, AES-256, two key TDEA, and three key TDEA. For each key, the generation of the associated sub keys is given, followed by four examples of MAC generation with the key. The messages in each set of examples are derived by truncating a common fixed string of 64 bytes. All strings are represented in hexadecimal notation, with a space (or a new line) inserted every 8 symbols, for readability. As in the body of the Recommendation, K1 and K2denote the sub keys, M denotes the message, and T denotes the MAC. For the AES algorithm examples, Then is 128, i.e., 32 hexadecimal symbols, and K denotes the key. For the TDEA examples, Tlen is 64, i.e., 16 hexadecimal symbols, and the key, K, is the ordered triple of strings, (Key1, Key2, Key3). For two key TDEA, Key1 = Key3. D.1 AES-128

For Examples 1–4 below, the block cipher is the AES algorithm with the following 128 bit key: K
2b7e1516 28aed2a6 abf71588 09cf4f3c.

Subkey Generation CIPHK (0128) 7df76b0c 1ab899b3 3e42f047 b91b546f K1
fbeed618 35713366 7c85e08f 7236a8de

K2 f7ddac30 6ae266cc f90bc11e e46d513b

Example Explanations

The Advanced Encryption Standard (AES) specifies a FIPS-approved cryptographic algorithm that can be used to protect electronic data. The AES algorithm is a symmetric block cipher that can encrypt (encipher) and decrypt (decipher) information. Encryption converts data to an unintelligible form called ciphertext; decrypting the ciphertext converts the data back into its original form, called plaintext. The AES algorithm is capable of using cryptographic keys of 128, 192, and 256 bits to encrypt and decrypt data in blocks of 128 bits.

Feasibility Study

Feasibility study is the test of a system proposal according to its workability, impact on the organization, ability to meet user needs, and effective use of recourses. It focuses on the evaluation of existing system and procedures analysis of alternative candidate system cost estimates. Feasibility analysis was done to determine whether the system would be feasible.

The development of a computer based system or a product is more likely plagued by resources and delivery dates. Feasibility study helps the analyst to decide whether or not to proceed, amend, postpone or cancel the project, particularly important when the project is large, complex and costly.Once the analysis of the user requirement is complement, the system has to check for the compatibility and feasibility of the software package that is aimed at. An important outcome of the preliminary investigation is the determination that the system requested is feasible.

Technical Feasibility:

The technology used can be developed with the current equipments and has the technical capacity to hold the data required by the new system.

- This technology supports the modern trends of technology.
- Easily accessible, more secure technologies.

Technical feasibility on the existing system and to what extend it can support the proposed addition. We can add new modules easily without affecting the Core Program. Most of parts are running in the server using the concept of stored procedures.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

Operational Feasibility:

This proposed system can easily implemented, as this is based on JSP coding (JAVA) & HTML .The database created is with MySql server which is more secure and easy to handle. The resources that are required to implement/install these are available. The personal of the organization already has enough exposure to computers. So the project is operationally feasible.

Economical Feasibility:

Economic analysis is the most frequently used method for evaluating the effectiveness of a new system. More commonly known cost/benefit analysis, the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If benefits outweigh costs, then the decision is made to design and implement the system. An entrepreneur must accurately weigh the cost versus benefits before taking an action. This system is more economically feasible which assess the brain capacity with quick & online test. So it is economically a good project

Methodology Waterfall
Approach

While the Waterfall Model presents a straightforward view of the software life cycle, this view is only appropriate for certain classes of software development. Specifically, the Waterfall Model works well when the software requirements are well understood (e.g., software such as compilers or operating systems) and the nature of the software development involves contractual agreements. The Waterfall Model is a natural fit for contract-based software development since this model is document driven; that is, many of the products such as the requirements specification and the design are documents. These documents then become the basis for the software development contract.

There have been many waterfall variations since the initial model was introduced by Winston Royce in 1970 in a paper entitled: “managing the development of large software systems: concepts and techniques”. Barry Boehm, developer of the spiral model (see below) modified the waterfall model in his book *Software Engineering Economics* (Prentice-Hall, 1987). The basic differences in the various models is in the naming and/or order of the phases.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

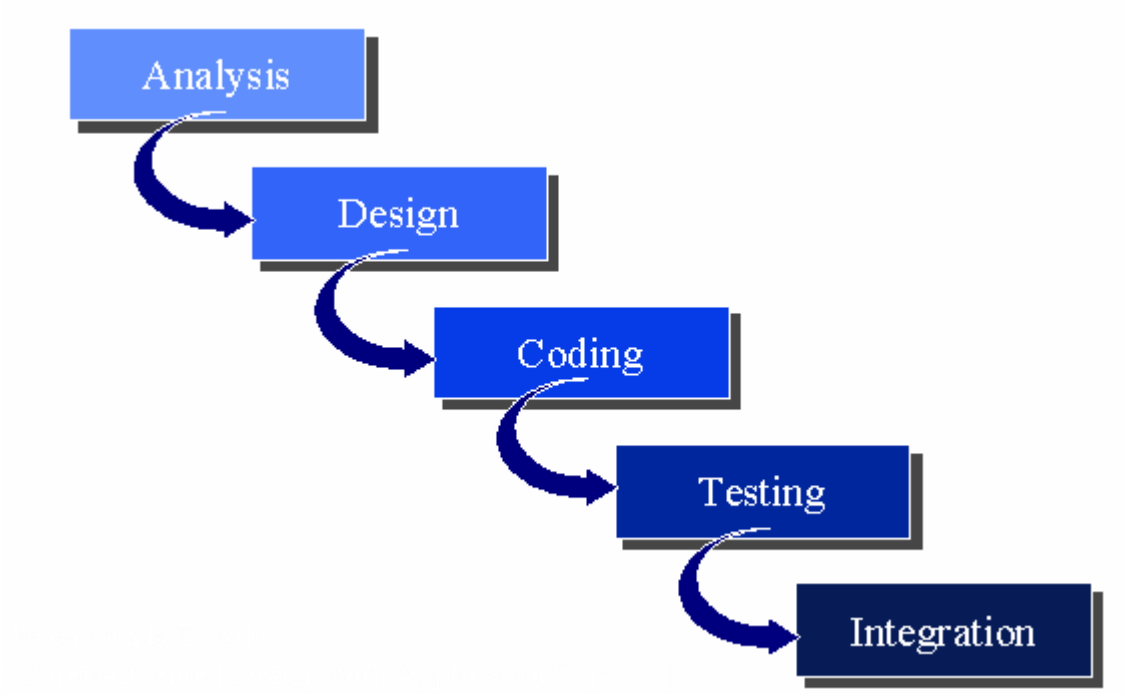
International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

The basic waterfall approach looks like the illustration below. Each phase is done in a specific order with its own entry and exit criteria and provides the maximum in separation of skills, an important factor in government contracting.



Example of a typical waterfall approach

While some variations on the waterfall theme allow for iterations back to the previous phase, "In practice most waterfall projects are managed with the assumption that once the phase is completed, the result of that activity is cast in concrete. For example, at the end of the design phase, a design document is delivered. It is expected that this document will not be updated throughout the rest of the development. You cannot climb up a waterfall." (Murray Cantor, Object-oriented project management with UML, John Wiley, 1998)

The waterfall is the easiest of the approaches for a business analyst to understand and work with and it is still, in its various forms, the operational SLC in the majority of US IT shops. The business analyst is directly involved in the requirements definition and/or analysis phases and peripherally involved in the succeeding phases until the end of the testing phase. The business analyst is heavily involved in the last stages of testing when the product is determined to solve the business problem. The solution is defined by the business analyst in the business case and requirements documents. The business analyst is also involved in the integration or transition phase assisting the business community to accept and incorporate the new system and processes.

V Model

The "V" model (sometimes known as the "U" model) reflects the approach to systems development where in the definition side of the model is linked directly to the confirmation side. It specifies early testing and preparation of testing scenarios and cases before the build stage to simultaneously validate the definitions and prepare for the test stages.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

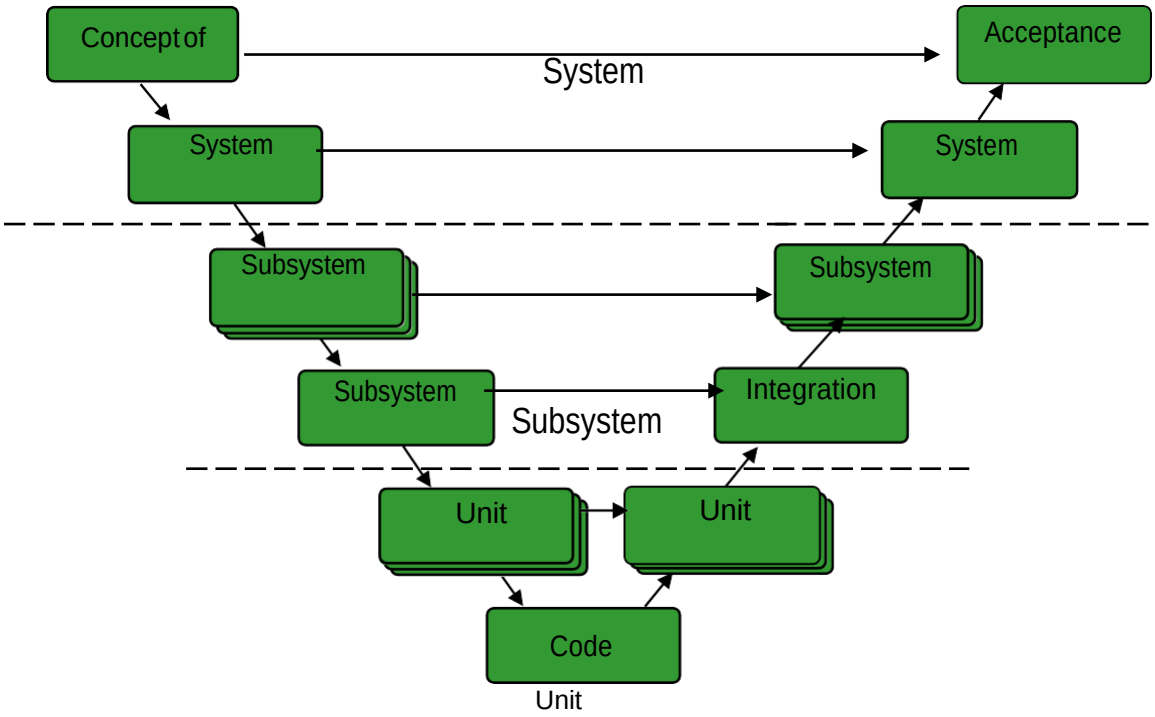
Vol. 8, Issue 10, October 2019

It is the standard for German federal government projects and is considered as much a project management method as a software development approach.

"The V Model, while admittedly obscure, gives equal weight to testing rather than treating it as an afterthought. Initially defined by the late Paul Rook in the late 1980s, the V was included in the U.K.'s National Computing Centre publications in the 1990s with the aim of improving the efficiency and effectiveness of software development. It's accepted in Europe and the U.K. as a superior alternative to the waterfall model; yet in the U.S., the V Model is often mistaken for the waterfall…

"In fact, the V Model emerged in reaction to some waterfall models that showed testing as a single phase following the traditional development phases of requirements analysis, high-level design, detailed design and coding. The waterfall model did considerable damage by supporting the common impression that testing is merely a brief detour after most of the mileage has been gained by mainline development activities. Many managers still believe this, even though testing usually takes up half of the project time." (Goldsmith and Graham, "The Forgotten Phase", Software development, July 2002)

As shown below, the model is the shape of the development cycle (a waterfall wrapped around) and the concept of flow down and across the phases. The V shows the typical sequence of development activities on the left-hand (downhill) side and the corresponding sequence of test execution activities on the right-hand (uphill) side.



Example of a typical V Model (IEEE)

The primary contribution the V Model makes is this alignment of testing and specification. This is also an advantage to the business analyst who can use the model and approach to enforce early consideration of later testing. The V Model emphasizes that testing is done throughout the SDLC rather than just at the end of the cycle and reminds the business analyst to prepare the test cases and scenarios in advance while the solution is being defined.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 10, October 2019

The business analyst's role in the V Model is essentially the same as the waterfall. The business analyst is involved full time in the specification of the business problem and the confirmation and validation that the business problem has been solved which is done at acceptance test. The business analyst is also involved in the requirements phases and advises the system test stage which is typically performed by independent testers – the quality assurance group or someone other than the development team. The primary business analyst involvement in the system test stage is keeping the requirements updated as changes occur and providing "voice of the customer" to the testers and development team. The rest of the test stages on the right side of the model are done by the development team to ensure they have developed the product correctly. It is the business analyst's job to ensure they have developed the correct product.

V. CONCLUSION

We design a secure data sharing scheme, Mona, for dynamic groups in an untrusted cloud. In Mona, a user is able to share data with others in the group without revealing identity privacy to the cloud. Additionally, Mona supports efficient user revocation and new user joining. More specially, efficient user revocation can be achieved through a public revocation list without updating the private keys of the remaining users, and new users can directly decrypt files stored in the cloud before their participation. Moreover, the storage overhead and the encryption computation cost are constant. Extensive analyses show that our proposed scheme satisfies the desired security requirements and guarantees efficiency as well.

REFERENCES

1. M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A.Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M.Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010.
2. S. Kamara and K. Lauter, "Cryptographic Cloud Storage," Proc.Int'l Conf. Financial Cryptography and Data Security (FC), pp. 136-149, Jan. 2010.
3. S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing," Proc. IEEE INFOCOM, pp. 534-542, 2010.
4. M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, "Plutus: Scalable Secure File Sharing on Untrusted Storage," Proc. USENIX Conf. File and Storage Technologies, pp. 29-42, 2003.
5. E. Goh, H. Shacham, N. Modadugu, and D. Boneh, "Sirius: Securing Remote Untrusted Storage," Proc. Network and Distributed Systems Security Symp. (NDSS), pp. 131-145, 2003.
6. G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage," Proc. Network and Distributed Systems Security Symp. (NDSS), pp. 29-43, 2005.
7. R. Lu, X. Lin, X. Liang, and X. Shen, "Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing," Proc. ACM Symp. Information, Computer and Comm. Security, pp. 282-292, 2010.
8. B. Waters, "Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization," Proc. Int'l Conf. Practice and Theory in Public Key Cryptography Conf. Public Key Cryptography, http://eprint.iacr.org/2008/290.pdf, 2008.
9. V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data," Proc. ACM Conf. Computer and Comm. Security (CCS), pp. 89-98, 2006.
10. D. Naor, M. Naor, and J.B. Lotspiech, "Revocation and Tracing Schemes for Stateless Receivers," Proc. Ann. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 41-62, 2001.