IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

VOLUME : 1 ISSUE : 3 Print / Issue Publication Date: 08-Oct-2016

ISSN : 2455-2143

Indexed In

Google SCHOLAR OPEN ACCESS doi Crossref

WWW.IJEAST.COM

editor@ijeast.com

International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 1, Issue 3, ISSN No. 2455-2143, Pages 103-106
Published Online December - January 2016 in IJEAST (http://www.ijeast.com)

SECURED MAIL

Abhishek Srivastava
GLBITM,
Greater Noida

Shivangi Singh
GLBITM,
Greater Noida

Shalini Gupta
GLBITM,
Greater Noida

Antu Rani
GLBITM,
Greater Noida

ABSTRACT - Today, since the world is going global, and trillions of data are transferred daily across networks, security is looming on the horizon as a potentially massive problem. The generic name for the collection of tools designed to protect data and to thwart hackers is Computer Security.

Keywords *- private and public key encryption, hashing, cipher, repudiation, integrity*

I. DIGITAL SIGNAL METHODOLOGY

Digital Signature is a method to encrypt a message (such as documents, contracts, notifications) which will be transferred, adopting data-exchanging protocol and data-encrypting algorithm. An abstract is produced in this procession. The Hongjie Zhu is with the School of Information Science and Technology, abstract is like signature or seal which can be used by receiver to verify the identity of the sender. The functions of digital signature: Assuring data integrity. Once the message changes a little, the abstract will change a lot for hash function's peculiarity, so that avoids the message being distorted. (2) Anti-deniability. Using public key Cryptography algorithm, the sender can't deny that he has sent the message for he has the private key. (3) Avoiding receivers forging message that is claimed to be from the sender. Public Key Encrypting Scheme: As the base of digital signature technology, public key encrypting technology should be introduced first in the following content. In the traditional cryptography system, the cipher code used in the process of encrypting plain text into cipher text and in the inverse process is the same. This method is called symmetric cryptography technology. Public key encrypting scheme is a kind of asymmetric.

Cryptography technology - It resolves the difficult problems in application. Its basic idea: the keys of the two parties are different. Every user has a key pair. One is private key which is saved by the user himself, another one is issued in public places such as internet for downloading or enquiring. Public key algorithm is very slow (with contrast to private algorithm). It is designed for a little data, but not for much data. It is usually used together with hash function in digital signature.

Principle and Procession of Digital Signature

Digital signature technology is realized by public key Research on Digital Signature in RSA Digital signature encrypting technology combining with data decomposition function. Decomposition function is hash function. Firstly, the message is made into abstract with help of hash function

Hash Function & Algorithm

Hash algorithm is an algorithm which is used to compute a data fingerprint of a data block. It is a one-way function which satisfies the following conditions: 1) Can receive data with any length2) Can produce abstract with fixed length3) Can compute abstract easily4) cannot compute message from abstract5) It is impossible to find two different messages which have same abstract. Hash function can make short abstract with fixed length for the binary data with any length. The popular hash algorithms are MD5, Secure Hash Algorithm (SHA, having all kinds of security level.) and so on.

RSA Algorithm

RSA is a commonly used scheme for digital signatures. In a broad outline of the RSA approach, the message to be signed is input to a hash function that produces a secure has h code of fixed length. This hash code is then encrypted using the sender's private key to form the signature. Both the signature and the message are then concatenated and transmitted. The recipient takes the message and produces a hash code. The recipient also decrypts the signature using the sender's public key. If the calculated hash code matches the decrypted signature, the signature is accepted as valid. This is because only the sender knows the private key, and thus only the sender could have produced a valid signature. The signature generation and verification using RSA is identical to the schemes.

Algorithm to Be Used

Message will be encrypted by the row column transformation algorithm in this algorithm m the given message, numbers, symbols, or expressions, should get arranged in rows and columns by deciding the no. of element we want in one row which should *be set as key*. In this units of plaintext (which are commonly characters or groups of characters) are shifted according to an r regular system, so that the cipher text constitutes

International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 1, Issue 3, ISSN No. 2455-2143, Pages 103-106
Published Online December - January 2016 in IJEAST (http://www.ijeast.com)

a permutation of the plaintext. That is, the order of the units is changed (the plaintext is reordered). Mathematically objective is function is used on the character s' positions to encrypt and an inverse function to decrypt. Here for example

W . . . E . . . C . . . R . . . L . . . T . . . E

E. R. D. S. O. E. E. F. E. A. O. C.

A . . . I . . . V . . . D . . . E . . . N.

Then reads off: WECRL TEERD SOEEF EAOCA IVDEN The cipher has broken this cipher text up into blocks of five to help avoid errors. This is a common technique used to make the cipher more easily readable. The spacing is not related to spaces in the plaintext and so does not carry any information about the plaintext. The message is written out in rows of a fixed length, and then read out again column by column, and the columns are chosen in some we write this into the grid as follows6 3 2 4 1 5 W E A R E D I S C O V E R E D F L E E A T O N C E Q K J E U EVLNE ACDTK ESEAQ ROFOJ DEECU WIREE In this way our message will get encrypted.

Software Requirement Specification

A System Requirements Specification (SRS) is a document where the requirements of a system that is planned to be developed are listed. The System Requirements Specification (SRS) document describes all data, functional and behavioral requirements of the software under production or development. Software requirements specification (SRS) is a comprehensive description of the intended purpose and environment for software under development. The SRS fully describes what the software will do and how it will be expected to perform. An SRS minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines how an application will interact with system hardware, other programs and human users in a wide variety of real-world situations. Parameters such as operating speed, response time, availability, portability, maintainability, footprint, security and speed of recovery from adverse events are evaluated. “SRS is an Agreement between developer and Client.” The software requirement specification document enlists all necessary requirements for project development. To derive the requirements we need to have clear and thorough understanding of the products to be developed. This is prepared after detailed communications with project team and the customer.

II. SOFTWARE DEVELOPMENT LIFE CYCLE

Generally, we are using SDLC for building database system. The six stages of the SDLC are designed to build on one another, taking the outputs from the previous stage, adding additional effort, and producing results that leverage the previous effort and are directly traceable to the previous stages. This top- down approach is intended to result in a quality product that satisfies the original intentions of the customer. Too many software development efforts go away when the development team and customer personnel get caught up in the possibilities of automation. Instead of focusing on high priority features, the team can become mired in a sea of “nice to have” features that are not essential to solve the problem, but in themselves are highly attractive. This is the root cause of a large percentage of failed and/or abandoned development efforts, and is the primary reason the development team utilizes the Waterfall SDLC.

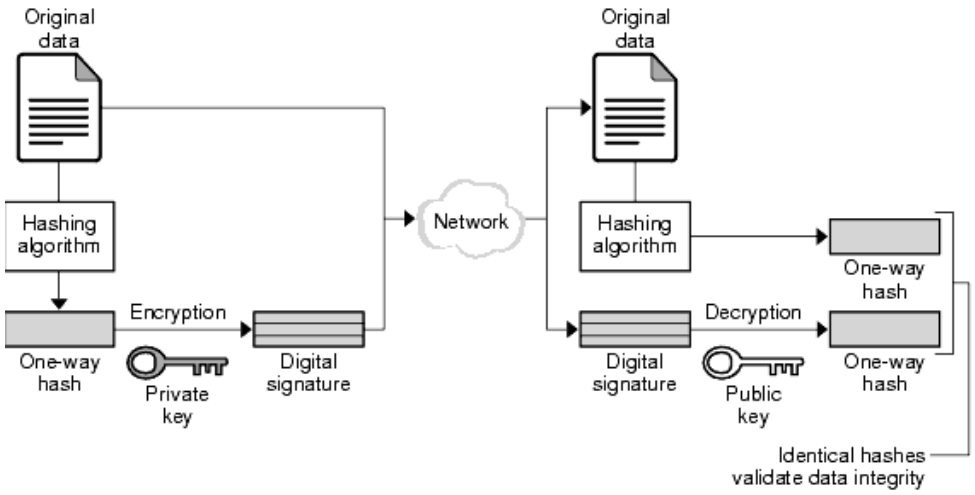

Figure 1

International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 1, Issue 3, ISSN No. 2455-2143, Pages 103-106
Published Online December - January 2016 in IJEAST (http://www.ijeast.com)

2.1. Planning Stage

The most critical section of the project plan is a listing of high-level product requirements, also referred to as goals. All of the software product requirements to be developed during the requirements definition stage flow from one or more of these goals. The minimum information for each goal consists of a title and textual description, although additional information and references to external documents may be included. The outputs of the project planning stage are the configuration management plan, the quality assurance plan, and the project plan and schedule, with a detailed listing of scheduled activities for the upcoming Requirements stage, and high level estimates of effort for the out stages.

2.2. Design Stage

The design stage takes as its initial input the requirements identified in the approved requirements document. For each requirement, a set of one or more design elements will be produced as a result of interviews, workshops, and/or prototype efforts. Design elements describe the desired software features in detail, and generally include functional hierarchy diagrams, screen layout diagrams, tables of business rules, business process diagrams, pseudo code, and a complete entity-relationship diagram with a full data dictionary. These design elements are intended to describe the software in sufficient detail that skilled programmers may develop the software with minimal additional input. When the design document is finalized and accepted, the RTM is updated to show that each design element is formally associated with a specific requirement. The outputs of the design stage are the design document, an updated RTM, and an updated project plan.

2.3. Development Stage

The development stage takes as its primary input the design elements described in the approved design document. For each design element, a set of one or more software artifacts will be produced. Software artifacts include but are not limited to menus, dialogs, and data management forms, data reporting formats, and specialized procedures and functions. Appropriate test cases will be developed for each set of functionally related software artifacts, and an online help system will be developed to guide users in their interactions with the software. The TNP will be updated to show that each developed artifact is linked to a specific design element, and that each developed artifact has one or more corresponding test case items. At this point, the RTM is in its final configuration. The outputs of the development stage include a fully functional set of software that satisfies the requirements and design elements previously documented, an online help system that describes the operation of the software, an implementation map that identifies the primary code entry points for all major system functions, a test plan that describes the test cases to be used to validate the correctness and completeness of the software, an updated RTM, and an updated project plan.

2.4. Testing Stage

During the integration and test stage, the software artifacts, online help, and test data are migrated from the development environment to a separate test environment. At this point, all test cases are run to verify the correctness and completeness of the software. Successful execution of the test suite confirms a robust and complete migration capability .During this stage, reference data is finalized for production use and production users are identified and linked to their appropriate roles. The final reference data (or links to reference data source files) and production user list are compiled into the Production Initiation Plan. The outputs of the integration and test stage include an integrated set of software, an online help system, an implementation map, a production initiation plan that describes reference data and production users, an acceptance plan which contains the final suite of test cases, and an updated project plan. During the installation and acceptance stage, the software artifacts, online help, and initial production data are loaded onto the production server. At this point, all test cases are run to verify the correctness and completeness of the software. Successful execution of the test suite is a prerequisite to acceptance of the software by the customer. After customer personnel have verified that the initial production data load is correct and the test suite has been executed with satisfactory results, the customer formally accepts the delivery of the software. The primary outputs of the installation and acceptance stage include a production application, a completed acceptance test suite, and a memorandum of customer acceptance of the software. Finally, the PDR enters the last of the actual labor data into the project schedule and locks the project as a permanent project record. At this point the PDR "locks" the project by archiving all software items, the implementation map, the source code, and the documentation for future reference.

III. SYSTEM DESIGN

The design is a solution, the translation of requirement into ways of meeting them. The design will determine the success of the system. Based on the proposed system objectives, the major modules are identified and the operations to be carried out are determined. In the design phase of the system the user interaction screen, data base tables, inputs, outputs and screen are designed by using all the necessary fields in a compact manner. The redundancy and duplication of fields are avoided. System design involves first logical design and then physical constructions of the system. After logical design, a detailed specification of the system, which describes the inputs, outputs, files are developed. During the design phase of the system the following factors are considered. Data Flows – the movement of

International Journal of Engineering Applied Sciences and Technology, 2016
Vol. 1, Issue 3, ISSN No. 2455-2143, Pages 103-106
Published Online December - January 2016 in IJEAST (http://www.ijeast.com)

data into, around and out of the system. Data Stores- temporary and permanent collection of data. Processors- activities to accept manipulate and deliver data and information Procedures- methods and routines to achieve the intended results .The importance of software design can be stated with a single word quality. Design is placed where quality is fostered in software development. Design is the only way whose requirements are actually translated into a finished software product or system.

3.1 Input Design

The input design is the link that ties information system into the world of its users. Input design consist of developing specific procedures for data preparation, steps necessary to put the transaction data in the form that is usable for computer processing. Main objectives that guides in the input design stages are: Controlling the amount of Inputs Avoiding inordinate delay .The accuracy of the output depends on the accuracy of the input and its processing. Thus, for this proposed system, the input design is in the format of web s for the user in format of form for the administrator. Validation checks are to be built in the system to avoid any error entries from the users. Hence the input design is the process of converting user-oriented inputs to a computer based format. So, input interface design takes an important role in controlling the errors. Customized messages are given in place of system messages, while the data manipulation is being carried out. Enforcing integrity, data validation procedures are done in such a way that end-user is free such daily core. There for, the input interface design should be made in such a way that it can be easily understandable to the user by using meaningful and appropriate words.

3.2 Output Design

Computer output is the most important and direct source of information to the administrator. Efficient, intelligible output design should improve the systems relationship with the appraisal. A major form output, reports, is a hard copy from printer. When designing output, system analyst must accomplish the following. Determine what information should be present. Decide whether to display, print the information and select the output medium Arrange the presentation of information in an acceptable format Decide how to distribute the outputs.

IV. FUTURE SCOPE AND ENHANCEMENT

The development of this software has been done keeping in mind the future scope of this application. I find that it has good prospects in the future also. Modifiability: -This project may be modified in future because the ongoing developing projects is developing with bottom-up approach, it means it will develop using unit, and each unit will be integrated then it will be become sub-module and after integration of sub-module the main module will completed. So, it will easier task to develop the project in future. Portability: - The ongoing projects are compatible to almost all windows based system. It will work without any errors. The efficiency of the software will not be affected if will be transferred to other operating system. The project will be developed using windows-xp operating system. Reusability: -The reusability of the software is high because the developing project using Front-end as a PHP which is event-driven programming language. Therefore scope is enhanced; as it is able handle any kind of subject matter in future. All programs will be written in PHP and all databases will be stored in MY SQL so any experienced programmer can changed the software. Slight modification of program can affect this software. Authorized person of management should make so many changes in the program.

V. REFERENCES

[1] Herbert Scheldt. Java2 - The Complete Reference (7TH Edition).

[2] Phillip Hanna 2003. The Complete Reference 2.0 JSP

[3] Ivan Bayross 2003. Oracle 10g Developer Suite.

[4] Roger S. Pressman, 1982. Software Engineering: A Practitioner's Approach.

[5] Bryan Bashan, Kathy Sierra & Bert Bates, 2008 Head First Servlets & JSP**.**

[6] Sun Microsystems, API DOCS –JAVA, J2EE, Java Mail, Java Servlets, JSPs.

[7] Bellare M, Miner S K, 2008. A Forward-secure Digital Signature Scheme.

[8] Shimon Zhang, 2013. The Principle and Application of Internet Safety.

[9] Krawczyk H. 2008. Simple Forward-Secure Signatures from any Signature.

[10] Shaofang Yang, 2005. Java Programming Base.

[11] Malkin T, Micciancio D, Miner S, 2009.Effcient Generic Forward-secure

IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

ABOUT IJEAST

International Journal of Engineering Applied Science and Technology (IJEAST) is a peer-reviewed, open access journal that publishes high-quality research papers in the field of Engineering, Applied Science and Technology.

IJEAST aims to provide a platform for researchers, academicians, and professionals to share their innovative ideas, research findings, and practical experiences with the global scientific community.

FOCUS AREAS

- Engineering
- Applied Science
- Technology
- Innovation & Development
- Interdisciplinary Studies

PEER REVIEWED
All submissions are rigorously peer reviewed to ensure quality.

OPEN ACCESS
Free and unrestricted access to research for all.

GLOBAL REACH
Connecting researchers and professionals worldwide.

TIMELY PUBLICATION
We ensure a swift and efficient publication process.

For more information, visit our website
www.ijeast.com

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

editor@ijeast.com
www.ijeast.com
India

2455-2143