International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 9 Issue 5 ǁ 2021 ǁ PP. 29-40

Keystroke with Data Leakage Detection for Secure Email Authentication

Nila Saradha.R , Rishitha.R , Shahnas.M , Sneka.M .
Department of Information Technology, A.V.C. College of Engineering, Mannampandal, Mayiladuthurai.

Abstract:
In this project, design the system for mail application to register their details such as user name and password. At the time of password typing, time is calculated for typing whole password. So hackers are difficult to extract details. Also propose data allocation strategies have improved the probability of identifying leakages. And even if we had to hand over sensitive data, in proposed work implement secret key sharing method. Key will be verified before accessing the shared mail information. This will avoid the unwanted and malicious access of email data.

Date of Submission: 12-05-2021 Date of acceptance: 25-05-2021

I. INTRODUCTION:

1.1 Network Security:

A key area in security research is authentication, the determination of whether a user should be allowed access to a given system or resource. The important aspect of authentication is confidentiality and integrity. Also, for protecting any resource adequate authentication is the first line of defense. Here, for protection of resource we use authentication as a service. It is important that the same authentication technique should not be used in every situation. A complication is that users may have many passwords for Bank, network and web sites. The large number of passwords increases interference and it is lead to forgetting or confusing passwords. The acceptability of any authentication scheme greatly depends on its robustness against attacks as well as its resource requirement both at the client and at the server end. It means authentication scheme require processing at client and sever end. Due to the proliferation of mobile and hand-held devices the resource requirement has become a major factor. The implicit passwords main application is the protection of critical resources and systems. Nowadays users can access any information including banking and corporate database with the use of mobile phones.

However, our proposal can also be used in other scenario where confidentiality and integrity are the major security requirements. We propose our Authentication System for banking using Implicit Password. in which the scheme allows any image to be used and it does not need artificial predefined click regions with well-marked boundaries – a password can be any arbitrarily chosen sequence of points in the image with some finer differences. In IPAS, the server has the piece of information i.e. password at the time of authentication and at the time of registration, the user give this information to the server in an implicit form. Implicit password is particularly suited for mobile phones and portable computers, although it may be implemented for any computer.

To put it simply, authentication is the process that confirms a user's identity. Traditionally, this is done through a username and password. The user enters their username, which allows the system to confirm their identity; this system relies on the fact that (hopefully) only the user and the site's server know the password. The website authentication process works by comparing the user's credentials with the ones on file. If a match is found, the authentication process is complete.

1.1.1 Password Authentication

While password authentication is the most common way to confirm a user's identity, it isn't even close to the most effective or secures method. Anyone with your credentials could access your account without your permission, and the system wouldn't stop them. Most passwords are weak, and hacking techniques can break them in less and less time.

1.1.2 Email Authentication:

Email authentication is a password less option that allows users to securely log in using just an email address. The process is very similar to signing in with a Facebook or Twitter account, but this method offers a universal approach.

- **The user clicks the login button.** This opens a mail to link that directs the person to pre-written email that includes an encrypted token.
- **The user sends the email.** The message already comes with a recipient address so the user doesn't need to enter any information.
- **The server verifies the request.** Using a combination of token-based security checks, the user's identity is verified.

1.1.3Biometric Authentication:

Biometric authentication includes any type of authentication method that requires a user's biology. While this may seem like new-age technology, you're probably already using it to unlock the screen on your smartphone. Fingerprint scanning is the most well-known form of biometric authentication, but face recognition tools are an increasingly popular choice for developers.

Of course, hackers have a much more difficult time replicating a users' biological characteristics, but it is important to note that these authentication processes are often less secure than you'd initially assume. Small fingerprint scanners on smartphones only record portions of your fingerprint, for instance. Multiple images of part of a fingerprint are much less secure than a single, clear image. Remember, too, that biometric authentication can't be changed or altered if a user's fingerprints have been compromised. While biometric authentication holds a lot of promise, it's now most useful as an additional login tool to bolster another system.

1.2 Authorization

Once a user has been authenticated, the authorization process determines what permissions they have. Permissions are what the user is able to do and see on your website or server, and without them every user would have the same abilities and access to the same information. Permissions are crucial for a few a reasons:

- They prevent a user from accessing an account that isn't theirs.
- They restrict free accounts from getting premium features.
- They ensure internal accounts only have access to what they need.

1.3 Password authentication method

Smart- card-centered password authentication is likely one of the most handy and typically used two-factor authentication mechanisms. This technology has been greatly deployed in quite a lot of varieties of authentication applications which incorporate far off host login, on-line banking and entry manipulate of constrained vaults, activation of protection contraptions, and lots of extra. A sensible-card situated password authentication scheme includes a server S and a customer A (with identity IDA). In the beginning, S securely issues a smart-card to A with the wise-card being personalized with admire to IDA and an initial password. This segment is referred to as the registration segment and is applied best as soon as for each customer. In a while, A can access S within the login-and-authentication phase, and this section will also be implemented as commonly as wanted. Nonetheless, in this section, there could have more than a few sorts of passive and active adversaries in the communication channel between A and S. They may be able to eavesdrop messages and even alter, dispose of or insert messages into the channel. The protection intention of the scheme in this segment is to be certain mutual authentication between A and S. In detailed, the purchaser is required to each have the sensible-card and comprehend the password with a purpose to carry out the wise-card-established password authentication effectively with server S. In other words, the scheme must furnish two-factor authentication.

There are any other necessities/residences that are fascinating in observe. For instance, A could want to exchange password occasionally. Conventionally, this requires A to have interaction with S and S has to keep a password database for its purchasers. In this paper, we recommend the thought of letting a change the password at will without interacting with or notifying S (at the same time making certain two-factor authentication), and also casting off any password database on the server side. Beneath are the reasons. Lots of the present methods require the server to keep a database for the passwords or derived values of the passwords of its purchasers. The derived values of the passwords may also be received through using a password-founded KDF (key derivation operate) which takes a password and a known random price called salt and practice a hash operate or a block cipher for a number of iterations. Nevertheless, this procedure not simplest introduces scalability concern to the server but also makes the systems suffer from disastrous loss when the server is compromised and the password database is stolen via adversaries.

Present programs also undergo from other skills security vulnerabilities. One outstanding difficulty is safety towards offline guessing attack (often referred to as offline dictionary assault). The reason of offline guessing attack is to compromise a customer's password through exhaustive search of all possible password

values. In a password-established atmosphere, passwords are viewed to be brief and human memorizable, and the corresponding password house is so small that an adversary is in a position to enumerate all possible values within the area within some cheap period of time. For example, most of the ATM deployments use PINs (personal identification numbers) of simplest fthis to 6 digits long, so the password space has no a couple of million possible values. Hence, an additional security requirement for wise-card-established password authentication is security towards offline guessing attack. In particular, compromising a patron's sensible-card must not allow an adversary to launch offline guessing attack in opposition to the patron's password. In observe the adversary may just steal the wise-card and extract the entire information stored in it through reverse engineering. This concept is paying homage to password-founded authentication protocols.

The difference is that for password-situated authentication protocols, the focal point is on stopping adversaries from getting any useful know-how about the password from the transcripts of protocol runs under the idea that the 2 speaking parties should not compromised, at the same time for intelligent-card-headquartered password authentication schemes, we extra require that the consumer's password will have to stay relaxed even after the client's clever-card is compromised.

1.4 Web security

The Internet is a dangerous place! With great regularity, we hear about websites becoming unavailable due to denial of service attacks, or displaying modified (and often damaging) information on their homepages. In other high-profile cases, millions of passwords, email addresses, and credit card details have been leaked into the public domain, exposing website users to both personal embarrassment and financial risk.

The purpose of website security is to prevent these (or any) sorts of attacks. The more formal definition of website security is the act/practice of protecting websites from unauthorized access, use, modification, destruction, or disruption. Effective website security requires design effort across the whole of the website: in your web application, the configuration of the web server, your policies for creating and renewing passwords, and the client-side code. While all that sounds very ominous, the good news is that if you're using a server-side web framework, it will almost certainly enable "by default" robust and well-thought-out defense mechanisms against a number of the more common attacks. Other attacks can be mitigated through your web server configuration, for example by enabling HTTPS. Finally, there are publically available vulnerability scanner tools that can help you find out if you've made any obvious mistakes.

1.5 Common attacks/vulnerabilities

1.5.1 Click jacking:

In this attack, a malicious user hijacks clicks meant for a visible top-level site and routes them to a hidden page beneath. This technique might be used, for example, to display a legitimate bank site but capture the login credentials into an invisible <iframe> controlled by the attacker. Click jacking could also be used to get the user to click a button on a visible site, but in doing so actually unwittingly click a completely different button. As a defense, your site can prevent itself from being embedded in an iframe in another site by setting the appropriate HTTP headers.

1.5.2 Denial of Service (DoS):

DoS is usually achieved by flooding a target site with fake requests so that access to a site is disrupted for legitimate users. The requests may simply be numerous, or they may individually consume large amounts of resource (e.g., slow reads or uploading of large files). DoS defenses usually work by identifying and blocking "bad" traffic while allowing legitimate messages through. These defenses are typically located before or in the web server (they are not part of the web application itself).

1.5.3 Directory Traversal (File and disclosure):

In this attack, a malicious user attempts to access parts of the web server file system that they should not be able to access. This vulnerability occurs when the user is able to pass filenames that include file system navigation characters (for example, ../../). The solution is to sanitize input before using it.

1.5.4 File Inclusion:

In this attack, a user is able to specify an "unintended" file for display or execution in data passed to the server. When loaded, this file might be executed on the web server or the client-side (leading to an XSS attack). The solution is to sanitize input before using it.

1.5.5 Command Injection:

Command injection attacks allow a malicious user to execute arbitrary system commands on the host operating system. The solution is to sanitize user input before it might be used in system calls.

II. LITERATURE SURVEY:

2.1 Title: A Novel Behaviour Profiling Approach to Continuous Authentication for Mobile Applications
Authors: Alotaibi, Saud, Abdulrahman Alruban, Steven Furnell, and Nathan L. Clarke
Present a novel behavioural profiling approach to user identity verification as part of mobile application security. This work presented a novel behavioural profiling approach to verifying the user in terms of mobile application security and providing robust user identification. In this proposed work, three supervised machine learning algorithms were selected to evaluate the proposed approach and to determine the ideal classifier based on EER value. The experimental results show that the significance of this research lies in having successfully applied continuous user verification for mobile applications in a manner that fulfils both security and usability requirements. Although the authentication decision is based on action resolution, the experimental results are still promising. Making an authentication decision on each user action might lead to an unusable system which does not present transparent authentication.

Each participant engaged in the study for at least 1 month, during which time they were all simply asked to use their device as normal. For the purpose of the data collection, a code was developed to extract log files from a backup file from the participants' devices by utilising the Android Debug Bridge (ADB), which is a command line tool that allows communication between a connected Android device and a computer. The backup file was extracted and a code run on SQLite to extract a log file from the backup file extracted for each application. During the data collection phase, applications were selected and collected. Some applications, such as Facebook, online mobile banking, and Chrome, are fully encrypted, and there was no means of collecting user data without compromising the user's privacy by asking the participant to root his/her device.

2.2 Title: Double serial adaptation mechanism for keystroke dynamics authentication based on a single password
Authors: Mhenni, Abir, Estelle Cherrier, Christophe Rosenberger, and Najoua Essoukri Ben Amara
Propose a double serial adaptation strategy that considers a single-capture-based enrollment process. When using the authentication system, the template of users and the decision/adaptation thresholds are updated. Indeed, the user introduces the password only once, when creating a new account. Thus, the reference is composed of a single sample. Afterwards, for each successful authentication, the reference is updated in a transparent way. Avoiding the enrollment phase, the growing window mechanism serves to increase the size of the reference to capture more intra-class variations. Once the size of the reference reaches 10 samples, the sliding window will be considered in order to limit the number of samples saved in the reference. Consider a preprocessing step which intends to eliminate the noise in the captured characteristics. Then use a single sample to create a user's reference while avoiding the tedious step of typing the same password several times in the enrollment phase. Use a GA-KNN verification method: It is based on the optimized combination of multi-distance metrics for the KNN classifier, which shows a better performance. This combination is ensured by vote parameters that are optimized by GA and updated during the use of the system. Propose to adapt the reference and the used thresholds over time. Hence, proposed method also considers the decision of the adapted thresholds criterion (user and time-dependent).

2.3 Title: User Authentication Using Keystroke Dynamics via Crowdsourcing
Authors: Foresi, Andrew, and Reza Samavi
As a consequence of the largely adopted password authentication method, keyboards have become an inevitable tool on many devices. However, the typing behaviour of users when using keyboards (known as keystroke dynamics) tends to be different even if the passwords are the same. This means that in the case where an intruder has access to another user's password, the uniqueness of both parties typing behaviors will prevent the intruder from being able to gain access to this user's account despite having direct access to their password. An authentication system that uses passwords as well as keystroke dynamics incorporates two out of three authentication techniques via something you know and something are, with the latter being your unique typing style. The major advantage of this approach is that this additional layer of authentication is achieved without imposing any additional effort to the users and with a minimal process running in the background to capture the keystroke dynamics.

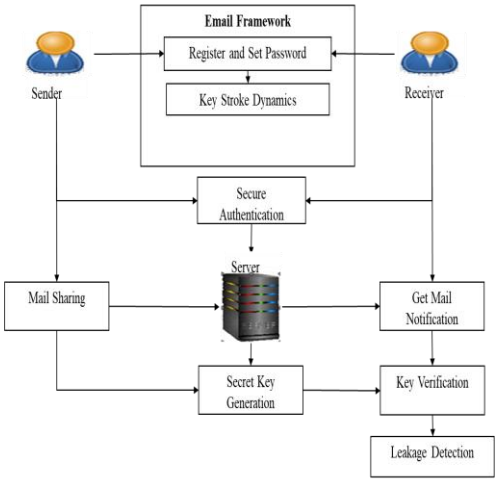
When a user types on a keyboard we are able to collect 3 distinct features; key hold time, key flight time and key latency time, all measured in milliseconds. Key hold time refers to the time from when a key is pressed until when it is released. Key flight time refers to the time from when a key has been pressed until the time when the next key in the password has been pressed. Key latency refers to the time between when the previous key was released to when the next key was pressed. A negative key latency indicates that the next key was pressed before the previous key has been released. Each of these features can be calculated from recording the timestamp vector (in milliseconds) of each key press and key release as a user types their password. For a password of length n, there will be n key hold times, n-1 flight times, and n-1 latency times. During the

enrollment phase when a user enters their password, key presses and key releases are recorded and a pattern profile is calculated from their entries. Calculate the average and standard deviation for each feature and require that on consecutive login attempts, at least 75% of the features recorded fall within 2.5 standard deviations of the mean times for each feature.

2.4 Title: A novel security scheme for behavioral authentication systems based on keystroke dynamics
Authors: Salem, Asma, and Mohammad S. Obaidat
Propose KSD as a second authentication scheme. Beside the traditional authentication by password, the users have to type specific suggested text to collect their typing rhythms. Android operating system–based touch screen devices will be used to conduct our experiments and study. Keystroke dynamics is one of common biometric-based systems, in which each user could be authenticated behaviorally in a unique and distinguished way. In this work, a user verification and identification system for touch screen mobile smart devices is proposed. The system examines the keystroke dynamics as a behavioral authentication means. The study proposed a prototype for a keyboard application developed for collecting timing and non timing information from keystroke dynamics. In addition, we proposed a complex password combination, which consists of text, numbers, and special characters. Strengthening access control using artificial neural networking and machine learning models is suggested. We present a unique approach for combining timing and non timing features together, as in our work we include several non timing features such pressure, size, and position in addition to the duration time feature. Different experiments will be implemented using timing and non timing features. Multiple comparative analyses are provided. MLP consists of multiple layers of nodes in a network, which has fully connected nodes. Except for the input nodes, each node is a neuron (or processing element) characterized by specific nonlinear activation function. MLP utilizes the well-known supervised learning technique algorithm, back propagation model, for training and testing. MLP is a modification of the standard linear perceptron, which can distinguish data that are not linearly separable as in our case for typing behavior of the users.13,14,24 MLP is a NN model that learns nonlinear function mappings. It is capable of learning a rich variety of nonlinear decision surfaces such as image, voice and typing behavior characteristics.

2.5 Title: An adaptive typing biometric system with varying users model
Authors: Ferrari, Carlo, Daniele Marini, and Michele Moro
Introduces a typing biometric system that continuously adjourns the users models for taking into account both short term amd long term modification in their habits. The system relies on a reduced space features that mainly uses the hold time and both the keyUp-keyDown and keyDown-keyUp time for some selected keys. An Adaptive Continuous Biometric Authentication Scheme, recomputes each user model according to his/her most recent typing history in a temporal sliding window of fixed dimension. Measures from continuous typing have to be properly summarized in order to efficiently support the check against the acquired models (built in the enrolment phase) and they are used to compute actual values of the so-called feature vectors. Matching indeed becomes feasible because users representation is moving from the space signals to a generally new space (the feature space) that is generally more robust with respect to all those randomness arising at different levels from data acquisition environment, to sensors transduction, and signal digital conversion and representation. On the other hand, the notion of feature space is not unique with respect to all the biometries being really dependent from the chosen one. The performance of a specific systems depends on the way the feature space is built: a richer one can results in a higher precision at lower computation speed and energy saving, while a barer one could give an almost useless system. When behavioral biometrics is used for continuous identification and recognition, the amount of raw data is really high and different feature vector definitions do affect the formation and updating of the user model. The use of a behavioral biometric parameter saves computational resources and energy and it does not require additional hardware. In order to lower the sensitiveness of the systems to expected changes in the way a user is typing, due to boredom, tiredness or mood, each genuine user model is continuously recomputed while data from typing are recorded. At the same time, also the model of the other enrolled users are recomputed before matching, resulting in a better system.

III. PROPOSED ARCHITECTURE WITH MODULE EXPLANATIONS:



MODULES DESCRIPTION :

EMAIL FRAMEWORK CONSTRUCTION

- A mail server is an application that receives incoming e-mail from local users and remote senders and forwards outgoing e-mail for delivery.
- A computer dedicated to running such applications is also called a mail server.
- In this module we can create the framework like as mail server.
- This framework contains server and multiple users. Server can maintain all user details.
- Users easily upload the files in inbox and also share the data anywhere and anytime.
- This framework enable for provide key stroke authentication and leakage detection process.

USER ENROLMENT

- In this Email application User has to register the appropriate details in the Email server database for using the authentication process.
- These details include user name, address, email id, contact number, primary password, confirm password and keystroke value.
- The key stroke value analyzed during password typing.
- Keystroke duration threshold and user details are stored in the server database.

KEYSTROKE AUTHENTICATION

- Anonymous access is the most common web site access control method, which allows anyone to visit the public areas of a website while preventing unauthorized users from gaining access to a critical features and private information of web servers.
- The user verification phase analyzes the mail id, password, keystroke value to the server.
- During password verification, key stroke time for password will be calculated and matched with database.
- During password verification, key stroke time for password will be calculated and matched with database.
- User should enter the password with the specified time, otherwise they will not allow to access application.

DATA SHARING

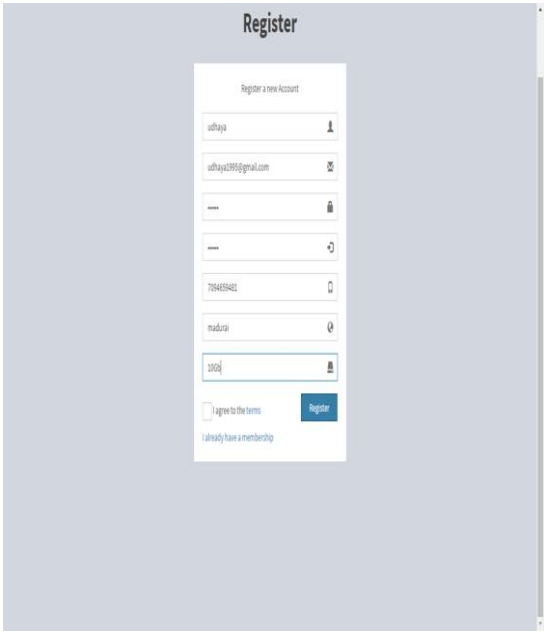
- User can share the message to another user in secure email environment.
- Once completion of authentication process they will be allow to compose the mail.
- Then add the recepient detail to communicate.
- Receiver also creates account with key stroke authentication method.
- Authorized users are allowed to access this application.

DATA LEAKAGE DETECTION

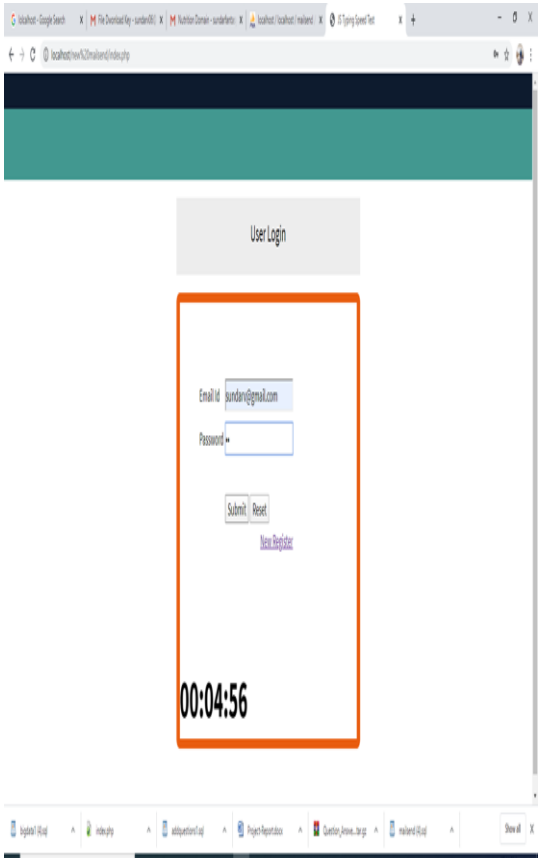
- The Mail is being sent to authorized user and unauthorized user.
- As the unauthorized user receives the mail, the system detects that the mail has been send to the unauthorized user using key verification process.
- Receiver want to verify their secret key before accessing mail content.
- Here, on the user side, if the unauthorized user accesses that mail, the mail does not display the contents of the mail.

IV. EXPERIMENTAL RESULTS:

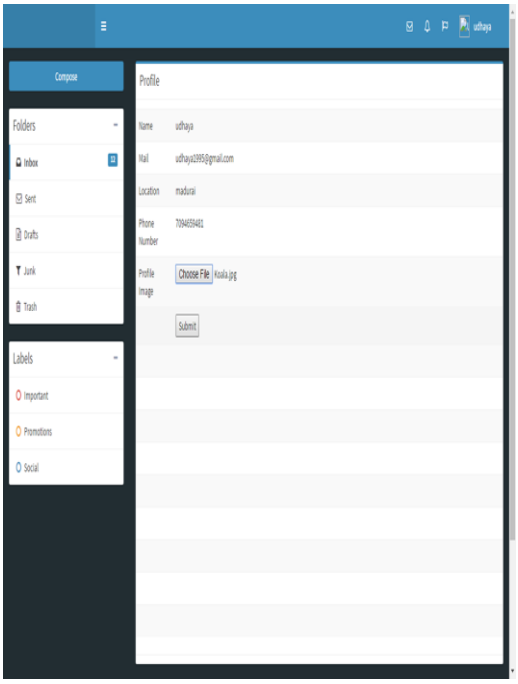
Register



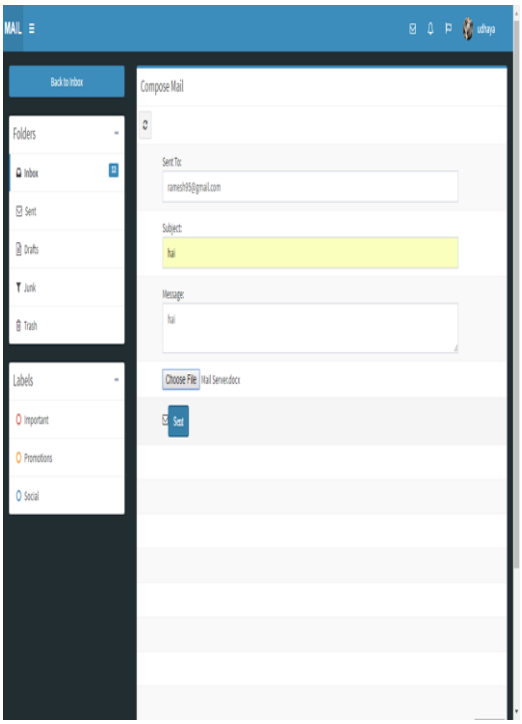
User Login With Key Stroke



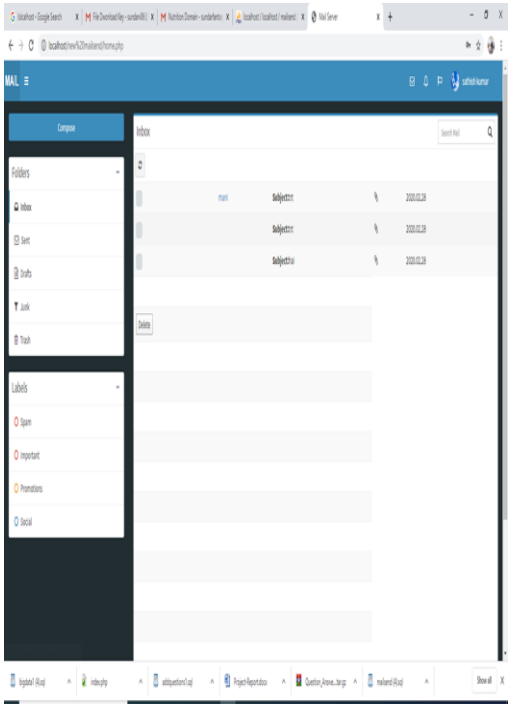
View Profile



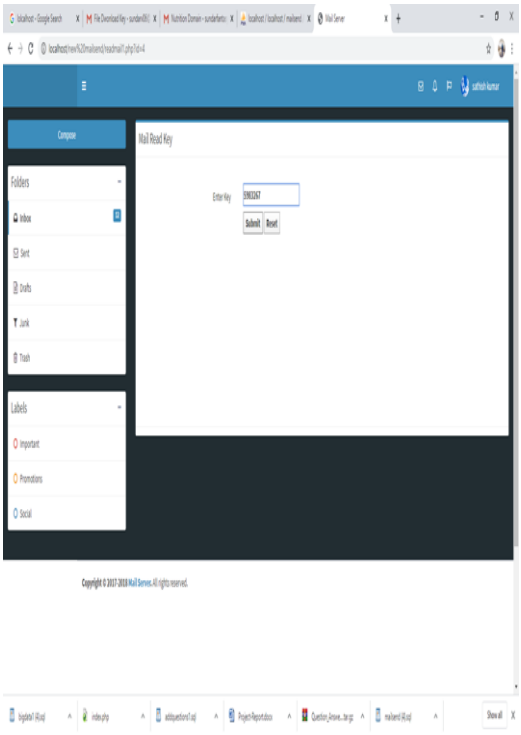
Compose Mail



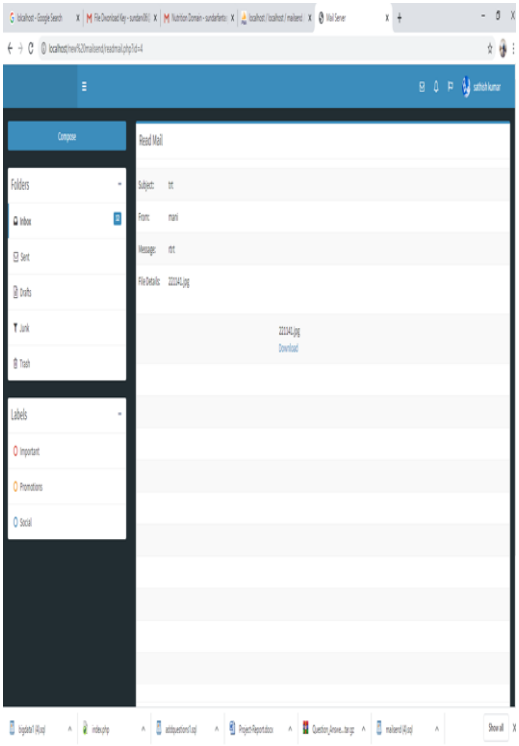
Receiver View Mail



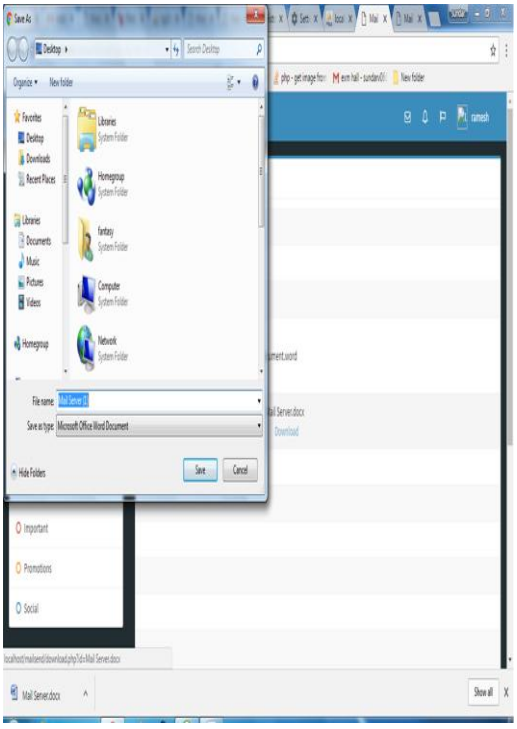
Key Verification



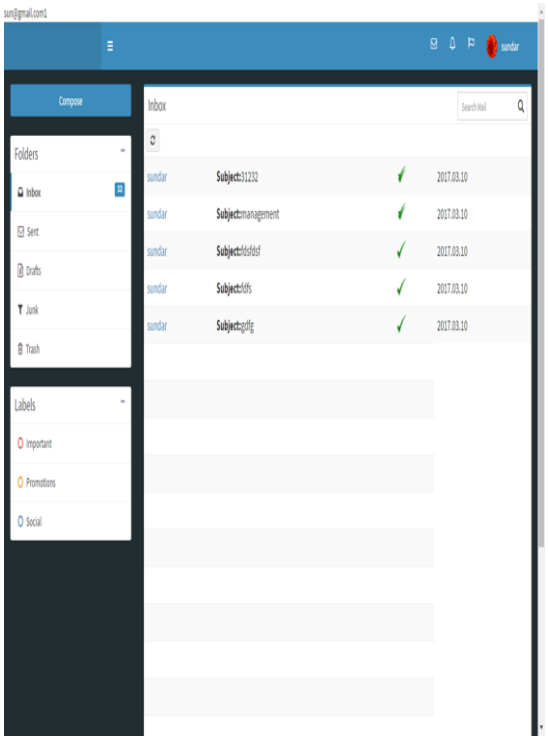
Read Mail



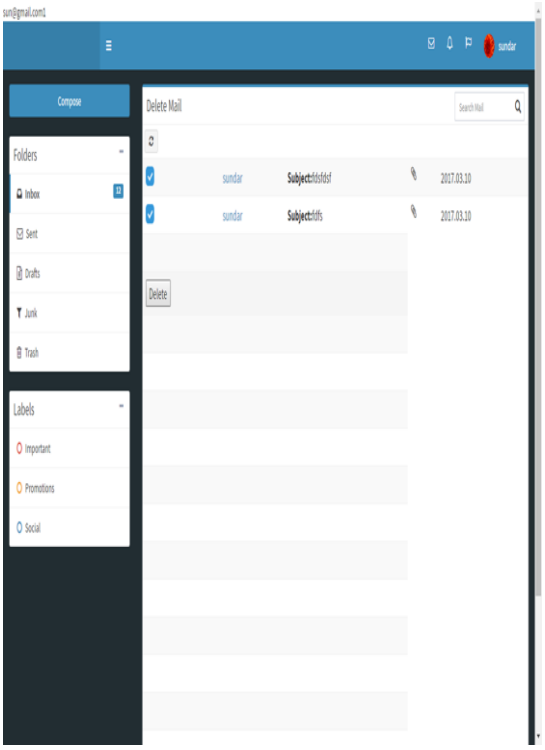
Download Files



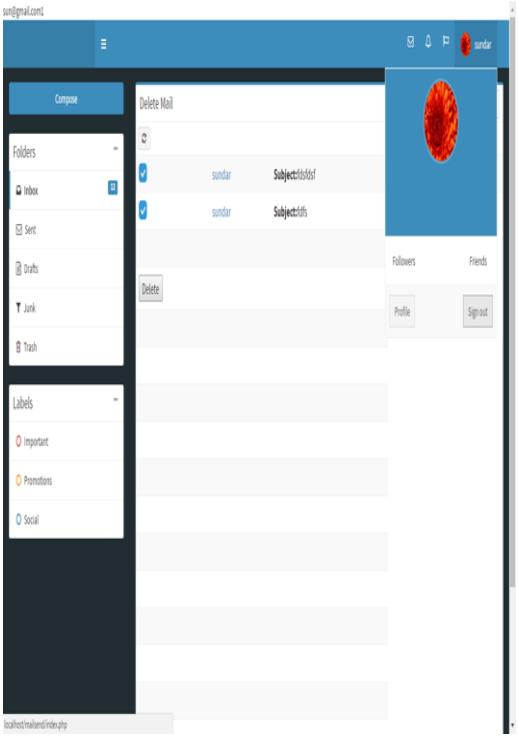
Inbox Checking



Mail Delete



Sign Out Process



V. CONCLUSION:

- To deal with the problem of Data leakage, we have presented implementation a variety of data distribution strategies that can improve the distributor's chances of identifying a leaker.
- Also we have implemented the concept of key stroke authentication for user authentication.
- In proposed email framework users register using their details with key stroke values.
- During login process, user can also verified using their password with key stroke values.
- This will enhance the process of authentication in email.

REFERENCES:

[1]. A. Shabtai, Y. Elovici and L. Rokach, "A Survey of Data Leakage Detection and Prevention Solutions," Springer Briefs in Computer Science, Springer, 2012.

[2]. J. Kumar and A. K. Singh, "Dynamic resource scaling in cloud using neural network and black hole algorithm," 2016 Fifth International Conference on Eco-friendly Computing and Communication Systems (ICECCS), Bhopal, 2016, pp. 63-67.

[3]. H. Taneja, Kapil and A. K. Singh, "Preserving Privacy of Patients based on Re-identification Risk," Fourth International Conference on Eco-friendly Computing and Communication Systems(ICECCS), 2015, pp. 448-454.

[4]. I. Gupta, and A. K. Singh, "Privacy and Security Architecture for Cloud Data," technical report, NIT Kurukshetra, India, 2016.

[5]. S. Chhabra and A. K. Singh, "Dynamic data leakage detection model based approach for MapReduce computational security in cloud," 2016 Fifth International Conference on Eco friendly Computing and Communication Systems (ICECCS), Bhopal, 2016, pp. 13-19.