

## Securing Cloud Emails Using Two Factor Authentication Based on Password/Apps in Cloud Computing

Kayigana Virgile and Huiqun Yu

*Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai 200237, China*

*Shanghai Key Laboratory of Computer Software Evaluating and Testing, Shanghai 201112, China*

*kayigana@yahoo.com, yhq@ecust.edu.cn*

### **Abstract**

*Cloud computing has been extending extensively, involving multiple users on the service it provides. Cloud based email is one of services provided by cloud computing with number of users expanding on yearly basis. Cloud computing faces concerning problem of security and privacy mainly based on its working environment. Identity authentication is one reliable method in cloud environment to identify users requesting for cloud services, currently user authentication in cloud computing is based on the credentials possessed by the user mostly username & password. However, this method has vulnerability of being compromised by illegitimate users when users' passwords have been revealed or cracked. The paper proposes a more secure structure, i.e the two-factor authentication which authenticates by requiring the username/password pair also needs a second factor to completely grant access. The second factor is App stored in users' smart devices. Username & password/Apps method will continue to allow users to set up passwords for their accounts, it will make a random compulsory code to be entered through App on smart phone devices hence keep privacy of cloud emails. Testing has been done with help of Python program environment. Comparing to recent related proposed schemes, our proposed scheme has advantage of high unbreakable security feature with low implement cost.*

**Keywords:** *App, multifactor authentication, Biometric authentication, illegitimate users, cloud emails, Identity authentication, Python programming, DES algorithm*

### **1. Introduction**

Cloud computing is a recent hot computer technology topic. Cloud computing gets its interesting name coming from of its operating features; yet by knowing much about cloud characteristics everyone realizes that its technology isn't that new, only environment of implication had expanded. According to [1] The "Cloud" in cloud computing has many meanings driving through everything; from computing power to computing infrastructure, applications, business processes to personal collaboration which can be delivered to the user as a service wherever and whenever it is need. Cloud computing offers a good opportunity to improve productivity and lessens costs. Majority of the services are closely related to data storage or email flow; hence it became a target for malicious users to take advantage of cloud environment in order to get access of data which do not belong to them. Cloud computing is joined by multiple users with different purposes; some users go out in the cloud to attack cloud servers in order to steal information stored there by victims therefore cloud computing comes up with uncountable security risks such as data security; data confidentiality; data

availability and so on. Previous papers have proposed several authenticating methods of overcoming data privacy problem.

Ali A. Yassin, *et al.*, [2], Hua-Hong Zhu, *et al.*, [3] and kshay A. Pawle, *et al.*, [4] proposed authentication methods in cloud that are all related to Biometric methods which are effective, they have ability of restricting password guessing, promises security of data stored in cloud, yet they have also downside as they would restrict many users and companies to use cloud facilities because they require expensive instruments & sophisticated systems (fingerprint scanner, voice recognition machine) which not everyone nor every company could afford to buy and wander around with; as consequence it does minimize number of users willing to use cloud facilities by using above mentioned methods. By minimizing number of users the cloud computing cost will increase which it is not what we want by choosing the cloud environment. In addition there is speculation that authentication processing time is long if identification of many users is done at same time due to fingerprint/voiceprint matching process in database. Cloud based emails in the cloud rely on username/password authentication method. Actually good passwords are believed to be random enough if consisted by uppercase letter; lowercase letters, numbers and signs. However strong passwords are easy to be forgotten by human beings. Hence human beings prefer passwords related to their hobbies, birthdays, favorite actor and so on; this makes it easier for malicious users to guess & crack victim password. To make matter worse nearly all people use same password for multiple email accounts whenever one email account is compromised other mail accounts fall under privacy breach. The above challenges pose question on what good method could be less costly and effective to address people concerns about password guessing or being cracked problem. This paper proposes how to identify legitimate users using a two factor authentication (password/App) method.

## 2. Literature Review

Statistics given in [5] proves that users are aware of the cloud based emails privacy, up to 83% respondents are aware that their emails are stored in the cloud, 85% respondents feel their mails are not safe (still vulnerable) in the cloud, 58% organizations plan to move a significant portion of their IT to the cloud within couple of years. Obviously these numbers emphasize cloud based emails service still has a mountain to climb in order to satisfy users on security perspective. Until now many papers devoted their research time to address data privacy in cloud computing, very few papers have presented solution on the cloud based email privacy and its security problem [6, 7]. Ahmad-Reza Sadeghi, *et al.*, in [8] propose a method involving encryption to be applied on cloud platform. The idea is good but it is done at Cloud service provider, it does not have anything to do with user himself/herself, users will not feel as safe as if method had involved them, there have been question raised about if service providers would perform encryption as promised, what if the encryption is not done properly?

On the other hand statistics in [9] proves that in 2012, 90% of smartphone owners access the same email account on mobile and desktop, this implies that bulk of email owners have smartphones with internet service available. Currently there is a two factor authentication method practiced in online banking systems brought to cloud email authentication. The authentication involves a user to log in the system using his/her username & password and provide code sent by text message to his/her mobile device as second factor. There isn't an argument about the effectiveness of the method; however it increases many other downside facts if put in practice for cloud emails.

It needs user to provide a phone number for authentication to cloud provider, sending text message increases costs, service might not be covered when trying accessing email where

mobile telecommunication is not covered for example in a foreign country. In addition after giving away the phone number, user could be targeted by spam calls/ identification information stolen once got in hands of malicious attackers. Another concern is if the SIM card is lost, legitimate user will be restricted to use own email, and password retrieve could be more complicated. The idea of bringing App technology in email authentication will help overcome issues raised, will serve greatest number of people. App usually stands for application that runs on computers in order to perform a precise task directly for the user. Computer applications run on system software, App normally will not need phone numbers to accomplish function it needs to provide, it requires a device with an application installed on a correct operating system, with correct username & password. This adds another layer to the account security so that hackers will have difficult time in accessing cloud emails even if they might possess username & password. Smartphone typically use Android and iOS operating system. Google and Apple provide many kinds of App softwares to facilitate users on their daily use. Development of users using smartphone need to be exploited and put in practice to help securing cloud emails.

### 3. Proposed Scheme

This section describes cloud design, where two-factor authentication structure is proposed. The proposed cloud structure has two major advantages as follows.

- 1) The goal of proposed scheme is to maintain data privacy although user's password might be compromised. Proposed scheme involves a compulsory passcode read from user's cell phone App. Once the App is developed, it would be used by a great number of users hence reduce working rate at the side of cloud provider.
- 2) Many users will be able to use proposed authentication method without barrier as most of them possess smartphone/tablets. Cell phone technology is improving nowadays so that 80% [10] worldwide population have mobile phones, that makes it 5 billion mobile phones in which 1.08 billion are smart phones and the number is set to increase in coming days.

The use of proposed algorithm requires user to activate cloud email App on his/her cell phone. Steps indicating how to accomplish activation are given below:

- a) Create account
- b) Is account created? If yes continue to next step; if no go back to first step
- c) Input username & password in the App, with a security code given by cloud provider
- d) Activate App
- e) Read MAC address of smart device. It is usually hexadecimal number we name it  $h$ .  $h$  will be encrypted using DES algorithm and stored in database for future use. Email passcode will be sent to device with MAC address  $h$ .

DES [11] is a block cipher, it operates on plaintext blocks of a given size (64-bits) and returns cipher text blocks of the same size. Thus DES results in a permutation among the 2 to the 64th power possible arrangements of 64 bits, each of which may be either 0 or 1. The App needs to maintain the following characteristics:

**Connectivity:** the App gets coding numbers for authentication only when the handheld device is online. It shouldn't show any numbers unless email access & connectivity is required by user.

**Convenience:** this App is user-friendly with a relatively small storage space occupied by application itself.

**Standardization:** the App should have ability to operate on mostly used operating system of smart devices to ensure it serves many users as it should.

**Security:** the App has ability to read handheld devices MAC address in order to differentiate them. MAC address is then stored in database in encrypted form. Authentication coding numbers sent to/ from handheld device must also be encrypted using DES algorithm.

**Scalability:** allowing users to manipulate their email security settings; including removing device not needed or adding a newly bought device. It should give user ability to report loss device for cancelation, and password retrieval.

### 3.1 Proposed Scheme Work Flow

After the App has been activated user will go through credential authentication as follows:

- Stage 1: User  $X$  requests for service (The first state is illustrated by user making service requisition to cloud service provider server. This could be done through computer, tablet or cell phone);
- Stage 2: Cloud provider verifies if user is already a member of its services
  - If YES-> user is asked to provide username and its corresponding password
  - If NO-> user is asked to register and pay for any service fee required (could be free). After registration user will be given (can choose) a username and will be asked to provide password for this username, this information will be stored in cloud database for future reference. Note that cloud service provider will make sure that the username does not exist in its previous stored usernames, if existed an error prompt will show up asking user to pick another username. After registration go to next step.
- Stage 3: Upon log in the email account; user must have been registered already; at least must have a username. Let  $M$  defines a function determining if user is a subscriber of cloud service. If exists the result is equal to 1 if false the result is equal to -1.

$$\left\{ \begin{array}{l} M(X) > 0 \text{ go to the next stage} \\ M(X) < 0 \text{ go back to stage 1} \end{array} \right.$$

- Stage 4: User is asked to provide credentials (pair of username/password). Let username be  $u$  ; password be  $p$

$$\left\{ \begin{array}{l} \text{if } (u \wedge p) == 1 \text{ go to the next stage} \\ \text{if } (u \wedge p) \neq 1 \text{ go back to stage 1} \end{array} \right.$$

- Stage 5: Let passcode provided by the App be  $A$  ; passcode entered by user be  $A_u$  ,  $A$  is sent to device with MAC address  $h$  ; if  $A$  comes from device with different MAC address  $h'$  , access will be denied.

$\text{if } (A = A_u) == 1 \text{ grant access}$

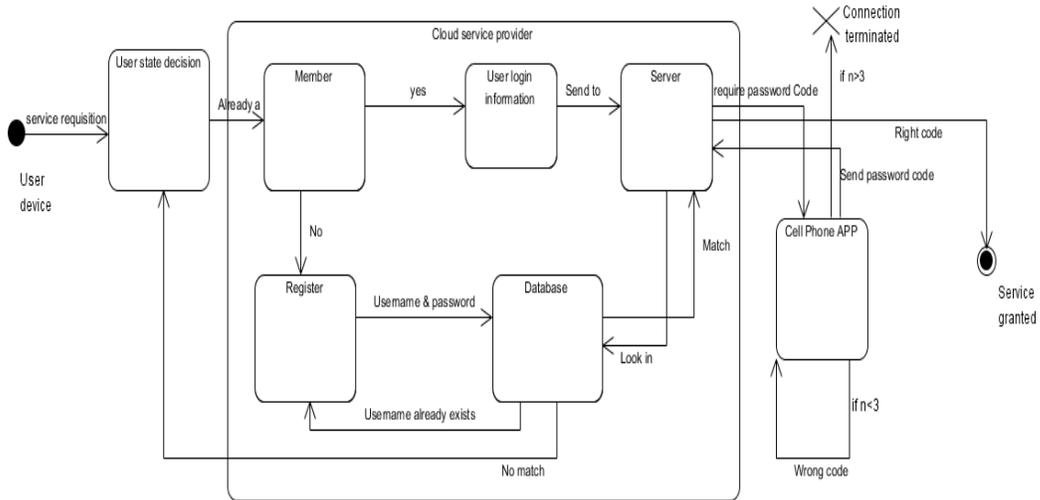
Let  $n = 0$  be an incrementing recurrence number counting how many times passcode is being entered

$$\left\{ \begin{array}{l} \text{if } (A = A_u) \neq 1 \text{ retry to enter new code} \\ n = n + 1 \\ \text{if } n = 3 \text{ deny access rights} \end{array} \right.$$

Authentication process in python programming is as follows:  
 username2=raw\_input('Login Enter your username: ')//Stage 2  
 password2=raw\_input('Please enter your password:')  
 login={username2:password2}  
 if username2 in authentication.keys(): //// Stage 3

```

password3=authentication[username2]
if password2==password3: //// Stage 4
    print"
    print'username and password match'
print("Now enter a digital code from you Cell App")
n=0
while(n<3):
    import random
    import math
    A =int(math.pow(10,6)*random.random())
    if A <100000:
        continue
    print'Please enter the following 6 digit code numbers:%s'% A   ////Stage 5
    A_u =int(raw_input("Enter code:"))
    n=n+1
    if( A_u == A ):
        print"Log in is completed"
        print'Login successful:'
        break
    if (n==3):
        print"Exceeded log in chances access is denied"
else:
    print'Your username and password provided don\'t match'
else:
    print"
    print'Username doesn\'t exist please register'
Proposed method schematic is as illustrated in below figure
    
```



**Figure 1. Proposed Scheme**

#### 4. Performance Evaluation

The following is the table consisting of results extracted from Python programming. The program initiates proposed scheme Username/password with second factor which is code sent to cellphone & tablet through App. Initially it asks user to register, login & enter random code provided by machine, it creates a list in database to recall username & corresponding passwords. At authentication phase a username is entered with a corresponding password, if they do not match; access to email will be denied, if username & password match; then user is asked to input a six digit code with 3 chances in case s/he mistyped one of the code. If user tries 3 times without entering the correct code access will be denied. Table 1 illustrates results extracted from a database of 100 users registered in database by doing 10 different attempts to log in using different username & passwords. The time unit in below table is seconds.

The average time it takes computer to verify if pair of username & password is a match is  $2.32201 \times 10^{-5}$  s. The average time to verify the code and get it right is 7.80635768 s

**Table 1. Python Programming Result for 100 Users in Database**

Times	Time to verify username/ password	time to verify digit code numbers
1	0.0000269	9.32624
2	0.00002566	7.7694098
3	2.60908E-05	8.61158
4	2.56631E-05	7.226473
5	2.65185E-05	8.41378
6	2.73739E-05	8.51021
7	0.00001112	6.54624
8	0.000016681	5.14636
9	0.000022669	10.19741
10	2.35245E-05	6.315874
average	2.32201E-05	7.80635768

It is worth mentioning that the time given in this table it is when a user gives credentials at a normal speed as a human (do not type username and come back 3 minutes later to input password). We conclude that authentication time at normal speed can be done in 10 seconds for 100 users in database. By assuming that a hacker had already known the username and password let us examine what are chances of guessing code sent to App. In this method we ask a user to type a number of six digit code. From probability we know that six digit numbers in total are 900000 numbers (100 000-999999). Probability of one number to be the code is  $1.11 \times 10^{-6}$ ; hacker has probability of 0.00000111 to guess right the number which is relatively small. In addition storage space used is relatively small; the file holding 100 usernames with corresponding passwords is a 4KB file. Below is table when the program is run whereas there are 200 users in database.

**Table 2. Python Programming Result for 200 Users in Database**

Times	Time to verify username/ password	time to verify digit code numbers
1	1.88198E-05	10.60922266
2	2.4808E-05	7.632074229

3	3.50733E-05	9.419320621
4	5.261E-05	8.087254775
5	1.88198E-05	5.308053866
6	1.62535E-05	6.124746039
7	1.96753E-05	6.711450762
8	2.22416E-05	6.404326767
9	2.27E-05	5.776995138
10	1.75E-05	7.347533617
average	2.48507E-05	7.342097847

From results perspective it is obvious that time spent on identification is barely changing, time is almost same; the file holding 200 usernames with corresponding passwords is 8KB file, which is 2 times larger than the previous one. By calculations suppose that if there were 200 million users in the database the file would be  $400000 \text{ KB} \cong 400 \text{ MB} \leq 1 \text{ GB}$ . This storage is really acceptable to hold credentials of all users comparing to nowadays computer storage capacity.

## 5. Comparison & Benefit Over Related Work

Many papers did research on cloud computing security and privacy, some authentication methods were proposed. From the beginning cloud computing evolved username/password method to identify cloud users; method is easy to use and to implement, however its security suffers from many attacks which reveal user credentials. An idea of two factor authentication has seen the light. A two factor authentication composed of username/password and simple codes sent by text message to user's mobile device as second factor, method is frequently used in E-banking authentication system. Recently proposed schemes largely involve biometric method. Voiceprint-Biometric Template Design and Authentication method in [2], anonymous Password Authentication Scheme by using digital signature and fingerprint method [3]. In [4] Face Recognition System (FRS) on Cloud Computing for User Authentication is also given. Methods mentioned above are effective in terms of authenticating users; they rely on people physiology characteristics which cannot be common to anyone. These characteristics are hard to forge and steal therefore they have ability of restricting password guessing; protect credentials to be compromised as a result promise security of data stored in cloud. Specialized instruments to extract physiology samples (fingerprint; palm; voice, face) are required by mentioned techniques; their compatibility is little more complicated to be put in practice in order so serve many users, from this view biometric machines are not universal. Our proposed scheme has advantage of letting user remain anonymous while accessing cloud service while it also lets user to select preferable password, even if password can be accidentally revealed account will still be safe because of second factor put in place. Second factor is sent on user portable device through App with less or probably no cost. User will be able to manage devices used to receive a second factor; second factor is very hard to guess nor break and it is unknown neither to cloud service provider nor user self. According to experiment done through python; authentication time is very short (less than 10 seconds overall). Comparing to traditional Id: password authentication method our proposed scheme has advantage of high unbreakable security feature with low implement cost. TABLE 3 describes security features cloud based email can provide. Those features include self-select password/ user being anonymous while trying accessing email, credential storage, more importantly if it is unbreakable and can serve many users worldwide.

**Table 3. Security Features**

Feature	Definition
F1	The user can select his password voluntary in the registration phase.
F2	No one can detect the user's identity except the service provider. (anonymity)
F3	Can change password if user needs.
F4	User has ability to change and choose second device to use for authentication.
F5	Second factor is random & unknown to anyone
F6	Cost of authentication instrument
F7	Authentication execution time
F8	Can reach many people
F9	high unbreakable security
F10	Credential storage

Above Table with 10 security features will help us compare our proposed authentication scheme to previous authentication scheme in [2-4] and traditional Id/password method with summary table below. Yes illustrate that mentioned method can provide the feature, while no illustrates otherwise.

**Table 4. Comparison of Authentication Schemes**

	Our scheme	The methods in [2],[3],[4]	Traditional Id/password method
F1	yes	yes	yes
F2	yes	yes	yes
F3	yes	yes	yes
F4	yes	no	n/a
F5	yes	no	n/a
F6	cheap	expensive	cheap
F7	short	long	short
F8	yes	no	yes
F9	yes	yes	no
F10	small	large	small

## 6. Conclusion

After activation of App, smart devices are being used to receive code. These devices have their unique characteristic to identify them a MAC address. Hacker may know the device user possesses but it is highly unlikely the hacker will know its MAC address, let it alone be able to falsify MAC address. As long as hacker does not physically steal device he/she will have no chance of accessing other users' emails. A big bonus is that statistics show that many people are using smart devices with internet thus our method can serve them. From results exploited from python with simple probability calculations, it is obvious that it will be less likely to guess random code. If the number of digits increases probability of guessing the code will diminish, on the other hand time verification after testing is very reasonable; it does not keep user waiting to confirm credentials. We took a close look at storage space of users' credentials which is relatively small. Space can be saved and allow other server storage to be used on other matters. As conclusion it is clear that authentication will be at least accurate and fast, it involves user himself/herself. It is user-friendly; users are familiar with the first step of input username and password so it wouldn't require additional training.

## ACKNOWLEDGEMENTS

This work was partially supported by the NSF of China under grants No. 61173048 and No. 61300041, Specialized Research Fund for the Doctoral Program of Higher Education under grant No. 20130074110015, and the Fundamental Research Funds for the Central Universities under Grant No. WH1314038.

## References

- [1] J. Hurwitz, R. Bloor, M. Kaufman and F. Halper, "Cloud Computing For Dummies", Wiley Publishing, Inc, (2010), pp. 8.
- [2] H.-H. Zhu, Q.-H. He, H.-H. Zhu, H. Tang and W.-H. Cao, "Voiceprint-Biometric Template Design and Authentication Based on Cloud Computing Security", International Conference on Cloud and Service Computing, (2011).
- [3] A. A. Yassin, H. Jin, A. Ibrahim and D. Zou, "Anonymous Password Authentication Scheme by using digital signature and fingerprint in cloud computing", Second International Conference on Cloud and Green Computing, (2012).
- [4] A. A. Pawle and V. P. Pawar, "Face Recognition System (FRS) on Cloud Computing for User Authentication", International Journal of Soft Computing and Engineering (IJSCE) ISSN, vol. 3, Issue 4, (2013) September, pp. 2231-2307.
- [5] T. Ayodele, "Information Intelligence, Infonetmedia Portsmouth, United Kingdom, Dennis Adeegbe IFREC IFREC, Osaka, Japan "Cloud Based Emails Boundaries and Vulnerabilities" Science and Information Conference, (2013) October 7-9, London, UK.
- [6] M. Willett and R. V. Solms, "A Framework for Assuring the Conformance of Cloud-based Email", the 8th International Conference for Internet Technology and Secured Transactions, (2013).
- [7] A. Joyia, A. Ghafoor, M. Sajjad and M. Q. Choudhary, "Secure and Privacy Enhanced Email System as a Cloud Service", IEEE, (2013).
- [8] A.-R. Sadeghi, T. Schneider and M. W. H. Görtz, "Institute for IT-Security, Ruhr-University Bochum, Germany", Token-Based Cloud Computing\* Secure Outsourcing of Data and Arbitrary Computations with Lower Latency" Trust and Trustworthy Computing Lecture Notes in Computer Science, vol. 6101, (2010), pp. 417-429.
- [9] (2014) February 5, <http://www.emailmonday.com/mobile-email-usage-statistics>.
- [10] (2012) October, <http://www.onbible.com/info/how-many-people-use-smartphones-in-the-world/>.
- [11] "NIST, U.S. National Institute of Standards and Technology", Federal Information Processing Standards Publication FIPS PUB 46-3. <http://csrc.nist.gov/publications/fips/archive/fips46-3/fips46-3.pdf>.

## Authors



**Kayigana Virgile** is pursuing Doctorate degree at East China University of Science and Technology in the Department of Computer Science and Engineering. His research interests include cloud computing security and mobile application security.



**Huiqun Yu** is a professor in the Department of Computer Science and Engineering at East China University of Science and Technology. He is a senior member of the IEEE and CCF, and a member of the

**ACM.** His research interests include software engineering, information security, formal methods, trustworthy computing and cloud computing.