Mobile Security (OTP) by Cloud Computing

Indrajit Das

Department of Computer Science & Engineering and Information Technology
Meghnad Saha Institute of Technology, Kolkata, West Bengal, India

Ria Das

Tata Consultancy Services Limited, Kolkata, West Bengal, India

Abstract- Cloud services have grown very quickly over the past couple of years, giving consumers and companies the chance to put services, resources and infrastructures in the hands of a provider. There is a big security concern when using cloud services. Security is very important in cloud computing since people and companies store confidential data in the cloud. It must also be easy to use the services provided, since cloud services have so many users with different technical background. Since the control of services and data needed for the everyday-run of a corporation is being handled by another company, further issues needs to be concerned. The consumer needs to trust the provider, and know that they handle their data in a correct manner, and that resources can be accessed when needed. This paper focuses on authentication and transmission encryption in cloud services. The current solutions used today to login to cloud services have been investigated and concluded that they don't satisfy the needs for cloud services. They are insecure, complex or costly. It can also be concluded that the best encryption algorithm to use in a cloud environment is AES, which is secure algorithm. This paper have resulted in an authentication and registration method that is both secure and easy to use, therefore fulfilling the needs of cloud service authentication. The method use a regular mobile phone to generate one time passwords that is used to login to cloud services. All of the data transmissions between the client and the server have been configured to use AES encryption. The conclusions that can be drawn is that the security proposal implemented in this paper functions very well, and provide good security together with an ease of use for clients that don't have so much technical knowledge.

Keywords – AES, Mobile One Time Password, One Time password, Personal Identification Number

I. INTRODUCTION

Cloud computing have developed rapidly over the past few years. However, cloud services also present a couple of issues. Since the resources are put under another provider, the customer will have no control over the situation. You don't know how your data is treated in the cloud, how sensitive data is encrypted, how the provider’s handle redundancy and backup of your data, can the resources always be accessed etc.

One of the bigger issues is the security part and one of the most important parts for a company that is thinking of moving services to the cloud. They need to know that their data is safe, both at the provider’s site and during transmissions between the host and server. Furthermore, the authentication procedure must be very secure; the best encryption algorithms in the world will not protect the data if someone has figured out your password.

Since cloud computing is a quite new subject, most of the cloud providers have not yet tighten up their security and still use insecure or complicated login methods. The authentication part of cloud computing must be easy and flexible for the millions of user that it has, but at the same time be very secure to protect the data that it stored in the cloud. At the same time the encryption method used during transmissions must also be very secure and since the cloud's vast amount of users, a fast algorithm that doesn't require much computer power and processing.

A. *Problem*

The most common login form used today, not only for cloud services, is to use static passwords. Many can agree that static password have a lot of security problems. Static passwords are often very easy to crack, since users prefer non-complex passwords. The users also rarely change their passwords or use the same password to access multiple services. Therefore, different cloud providers have lately started with *one time password* with *two-factor authentication.* The problem with their solutions is that it cost money, for the user or the provider, it can be complicated to use, or that the user have to carry a separate authentication device with him at all time.

One of the main concerns regarding cloud services is the security part, and is one large factor to why companies and customer hesitate to migrate their services into the cloud. At the same time, the security must be easy for the customers to understand and appeal to all kinds of people with different technical knowledge. And lastly, the security solutions should be very cheap or free of charge to implement, both for providers and customers, to attract more people to the cloud. So, in conclusion, for cloud services to grow even more, it needs a simple and cheap security solution.

B. Approach chosen to solve the problem

This paper propose that by using the user's mobile phone as an authentication device presenting a one time password for the user, and assuming most people always carry their phone with them, the problem with a separate authentication device for two-factor authentication is solved. As the mobile gives the user a one time password, the problem with static passwords for logins is also solved.

Furthermore, by using open source code in both the mobile phone and at the authentication server, the security solution is absolutely free of charge. That solves the problem of providing a free of charge security solution.

C. Goals

The goal with this paper is to implement a working authentication solution, which can be used in cloud services. The authentication method will be two-factor authentication with a mobile phone as the authentication device, which presents the user a password that is only valid one time for a certain amount of time. The password will only be given to the user after a successful 4-digit PIN input in the mobile phone software.

II. BACKGROUND

Cloud computing is a universal word for anything that involves distributing hosted services over the web or Internet. It can be an internet-based computing infrastructure that allows users to access different level of IT resources remotely through internet based client-side software as if it were installed locally in users own computer. Where the IT resources include server, storage, service, application, network and so on. These resources are associated in a large computer network which is owned by a company (Both privately and publicly). Cloud computing also provides services to others devices (such as smart-phones) on demand over the Internet [1] [2]. Companies, business organizations, academic or commercial researchers and any individual can be user of cloud computing. The main cloud service providers are Amazon, Salesforce and Google. Examples of large and well reputed IT firms that are dynamically involved in cloud computing are Microsoft, Fujitsu, Hewlett Packard (HP), IBM, Dell and VMware [1].

A. Existing problems in cloud computing

Cloud computing has turned into a standard information technology operation for many small or large businesses. It offers many considerable advantages, including probable expenditure savings. There are, however, major risk and disadvantages related with cloud computing.

Its dislocated nature is a benefit in many cases however can also be disadvantageous because the user has no supreme control over the software applications including secret data. Client has to depend on the provider to update, upgrade maintain and administer it. The user does not have direct access to the software to fix the problems while something goes wrong in any application and must rely on the service provider. The user can experience significant problems when the cloud provider is uncaring or incapable to fix the problem quickly.

Cloud computing can also mean big risks in the integrity, privacy areas and also greatly in users authentication. Using a cloud system, company’s susceptible data and information will be stored on third-party servers, and user will possibly have very inadequate understanding or control regarding this information. If the provider has insufficient security, or a violation of encryption systems or procedures are performed for different reasons, thus compromised company’s private and confidential data. This could have devastating consequences, and could cause lawful problems for company if third party private information (for example, customer information) is negotiated.

There are several problems in cloud computing and this paper is mainly focused on authentication based security issues in cloud computing and how it can be mitigated, the remaining part of the paper describes about this.

B. Authentication

In general authentication is the act of creating or validating something (or someone) as authentic and claims made about the topic are true. This might engage proving the identity of a person, guarantee that a product is what it’s wrapping and tagging claims to be, tracing the origins of a relic, or assuring that a computer program is a trusted one.

As cloud computing is a web based application so it might get these aforesaid attacks. In order to protect the cloud computing services from authentication attacks, it must need a very secure and strong authentication system therefore secure authentication in cloud computing is significantly important.

C. *Encryption*

Encryption is the core basis in cryptography system. It can be defined as the process of transforming information (usually plaintext) using cipher algorithm to make it unreadable to anyone without using the inverse decryption process [8].

D. *AES (Advance Encryption Standard)*

Advanced Encryption Standard (AES) is a symmetric key encryption technique which is securing files, e-mails, secure communication for LANs, hard drives, operating systems and other same related data. It works on multiple network layers simultaneously. [9]

E. *Two factor authentication OTP*

A one time password (OTP) is just what the names implies, a password that is only valid for one login. The benefit of OTPs is that it offers much higher security than static passwords, in expense of user friendliness and configuration issues.

OTPs is immune against password sniffing attacks, if an attacker use software to collect your data traffic, video records you when you type on your keyboard, or use social engineering, it doesn't matter since the password that the attacker gets hold on will not be valid to use. [10]

F. *Two factor authentication*

In two factor authentication a user has to supply two terms in order to authenticate himself. The user must have *something you know* used together with *something you have*. For example, when a user logins to a web page he writes his static password (*something you know*), and a series of random numbers from an authentication device (*something you have*). [11]

The most common implementation of this is when a person withdraws money from an ATM. The user has a bank card that he puts in to the machine, and a PIN code must then be entered before withdrawal is possible. [11] *Over the recent years, three factor authentication has also been introduced. This kind of authentication also needs "something you are", like a fingerprint or a voice print, together with the password and the physical token. [12]*

Two factor authentication together with OTP is much safer than static passwords, when looked at from an access attack perspective, such as sniffing, password cracking and social engineering. However, it cannot protect against two common attacks [13]: Man-in middle attack and Trojan attack.

G. *Two factor OTP authentication in cloud sevices*

More and more companies and providers have lately seen the flaws of static passwords and started to add different authentication methods. Facebook recently launched a service where you can get an OTP sent to your mobile device [14], Google Apps started with two factor authentication for some of its users [15] and Amazon Web Services offers a time-based OTP solution in order to increase the security for its users. [16]

II. Proposed Security Solution

There are ways to have a secure and easy-to-use cloud service that can satisfy these criteria's:

- Provide better password solution for login procedures than the insecure method of static passwords.
- Provide better two-factor OTP authentication solution than those discussed above.
- Have an easy-to-understand registration system, which at the same time doesn't compromise the security.
- Use an encryption algorithm that is secure but also fast, to be able to serve the vast amount of cloud users.
- Offer a solution that is free of charge in order to attract more customers to the cloud services.
- In overall, the security solution for cloud services must be easy to use, but also be very secure in order to protect the customer's data and gain the trust of the customers.

The solution presented here will be free of charge for both the users and the provider, and at the same time easy and flexible for the clients to download, install and use.

A. *Authentication Solutions*

a. *Proposal 1 – Authentication with mOTP*

The authentication method used is two-factor authentication with a one-time password, based on [3] and [4] but with modifications. The user's mobile phone will work as the authentication device, in which the user have to enter a 4-digit PIN code to generate an OTP that can be used for login. This is done by a Java-application running on the phone. The OTP that is generated on the mobile phone is based on three components which will be hashed together with SHA-1:

- The 4-digit PIN code that the user enter.
- A secret random number that was created during device-initialization (Init-secret) that only exists on the user's mobile device.
- The current time

After hashing, the mobile phone will display the first six numbers of the hash that will be used as the OTP for login. Since time is part of the hash, the OTP is only valid for three minutes. The OTP will then be sent to the server during login. The server knows the Init-secret and the pin-code, which is stored in a database, and also the current time. Therefore the password can be verified by the server.

The following example shows a simplified view on how the authentication process is done:

1. A client wishes to log in to a personal account through a web browser, and surfs to the login page.
2. The client then starts an application on a mobile phone, and enters a PIN code.
3. After PIN input, an OTP is generated and displayed on the phone.
4. The client enters his username and the OTP at the login page, and sends the information to the authentication server.
5. The server denies or permits access for the client.

This solution offers greater benefits than other types of authentication solutions:

- The only crucial information sent over the network will be the username and the OTP. Since the OTP is only valid for one time during a period of three minutes it will be of no value for an attacker.
- The OTP needs a private PIN code to be generated on the mobile phone, a PIN code that only the user knows.
- The cost will be absolutely free for both user and provider, since this is an open source solution.
- No needs to carry any extra authentication device, the user only have to carry his mobile phone with him.
- Easy registration process where everything can be done from home, no need to order an external authentication device or get the device from a local office.
- Easy to remove users mobile phones from the authentication database.

b. Proposal 1 – OTP with Challenge – Response

This proposal is similar to the first proposal, except that one additional security parameter is added to the system, called a challenge.

This technique is very common in use among banks all over the world, when a customer of a bank wants to log in to his/hers online account through a web browser. The customer still uses a OTP to login, but before the OTP can be generated the customer will get a challenge from the authentication server that must be entered into the authentication device. This will add a little more security to the system.

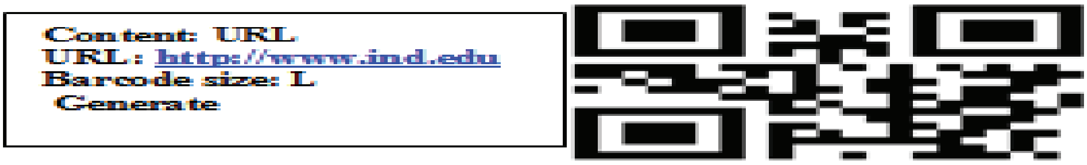
Different providers use different methods to implement this in a system. Following is a simple example on one method on how this is done, based on the login procedure at a bank:

1. A client wishes to log in to a personal account through a web browser, and surfs to the login page.
2. At the login page, a challenge is presented to the client. This can be a text string, image (see proposal 3), a random number etc.
3. The client enters the challenge into an authentication device, and then the personal PIN code into the same device.
4. From these two values, an OTP is generated, based on an algorithm, and presented to the client. (Often time or a counter is also added to the algorithm).
5. The client enters the OTP and sends it over the Internet to the authentication server.
6. The server denies or permits access for the client.

The benefits with this solution is the same as for proposal 1, but with an extra security feature in the form of a challenge

c. Proposal 1 – Optical Challenge – Response

In theory, this solution is similar to *proposal 2*; the client will get a challenge and give a OTP response to that challenge. The difference is that the system use two-dimensional bar-codes as challenges and responses, instead of the usual text-strings or number-combinations. The bar-codes can store information that is processed by different software's. In the figure below, the bar-code contains a URL to http://www.hh.se. The bar-code was generated using [5].



Content: URL
URL: http://www.ind.edu
Barcode size: L
Generate

Figure 1. Bar Code generated

This solution is not yet commercially spread, but research is being made. For example, [6] has a working system up and running where the client use a web-cam and the camera on a mobile phone to successfully authenticate with barcodes that contain authentication information.

The following example is one way to authenticate users with the use of bar-codes and cameras, based on the work in [6]. In order for this to work, the user needs to install an application on the mobile phone that can process the images. Furthermore, the mobile phone must have a camera, and the user needs to connect a web-cam to a computer. As you will notice, the procedure for authentication is the same as in *proposal 2*, but with images instead of text.

1. A client wishes to log in to a personal account through a web browser, and surfs to the login page.
2. The client is presented with a challenge in the form of a 2-dimensional bar-code. The bar-code is calculated by the server from an amount of random numbers.
3. The client starts an application on a mobile phone, and takes a photo of the bar-code, i.e. takes a photo of the computer screen.
4. From the taken picture, the phone application calculates a hash response based on the challenge. The response will be in the form of a bar-code and will be displayed in the mobile phone.
5. The client holds the phone in front of a web-cam, which will capture the response bar-code.
6. The response code is sent to the authentication server which will deny or permit access for the client.

The benefits of this solution is the same as for proposal 2, but with extra security since the bar-codes might bring more security than by sending crucial information as a text string.

B. Registration Solutions

How can cloud providers in a flexible and cheap way register clients to their authentication databases? In order for a user to log in to the cloud service using the proposed security solution, three statements is needed in a database on the server:

1. The mobile phone's Init-Secret value
2. A unique user name
3. A 4-digit PIN-code

However, there are problems with this solution. All of this information will be sent over an insecure network where someone might sniff the packet. Even if the packet is encrypted through TLS/HTTPS, there is a possibility that a hacker can decrypt the information and get full access to the user's account, especially since he don't have any time limit and can decrypt the packet offline. One more point, even though all the packets will be encrypted, the user will not be protected against Man-in-the-middle attacks, that can direct the user to a page with a fake certificate where the attacker can gather all of the information that the user provide during the registration part. [7]

To protect against Man-in-the-middle attacks, people need education on how to spot a fake certificate, and is out of the scope of this paper.

In order to provide a safe method to register to a service, these guidelines must be followed:

1. The Init-Secret can NEVER at ANY point travel over the network.
2. Since this is a cloud provider that customers will access through a web browser, the registration should be simple
3. The registration customers should be able to do the whole registration process over the Internet, not like the bank’s system where the user go to the local office to get the authentication device. That will not be flexible and possible for a cloud solution. Given the proposals discussed below and their pros and cons, the best registration proposal for this kind of security system, with respect to cost and flexibility, will be *Proposal 3*.

a. Proposal 1

The Init-secret is generated by pressing 25 random numbers on the mobile phone with the client software installed. The server can present this 25 random numbers to the user at the registration page, telling him to use those numbers to generate the Init-secret. The server uses the same numbers on the server side to generate the Init-secret. So both the server and the client have generated the same Init-secret but it has not been sent over the network.

Pros

- The Init-secret is not sent over the network.

Cons

- Instead the random numbers will be sent, which can be sniffed and the attacker can generate the same Init-secret, leading back to the starting point.

b. Proposal 2

Reprogram mOTP and put in a static Init-secret, then send the application to client's phone by email. The client only registers with username, PIN-code and an email address. The client must then wait for a file-attachment in an email.

Pros

- No crucial login information is sent over the network.
- If the encrypted username and PIN-code is cracked it will not matter, since a hacker can't login without the correct Init-secret. And today's best encryptions (AES, RC4) have not been cracked yet.
- The Init-secret will be safe inside an application. If someone tries to manipulate the application along the way it can be detected by hash-function.
- No configurations needed for the client, the application is ready-to-go.

Cons

- Harder to implement, more work for the server.
- If the process is not automated it can take a long time to get the server response.
- If a hacker already has access to the client's email-account, he will also have access to the application.
- Harder for users to install on the phone when they must first transfer the application from the computer. (If they don't access email via the mobile phone directly)

c. Proposal 3

Same as *Proposal 2* but instead of sending the application, the email contains an URL-link, that the user enter via the mobile phone, where the application can be downloaded. The page requires a login with username and PIN-code for extra security.

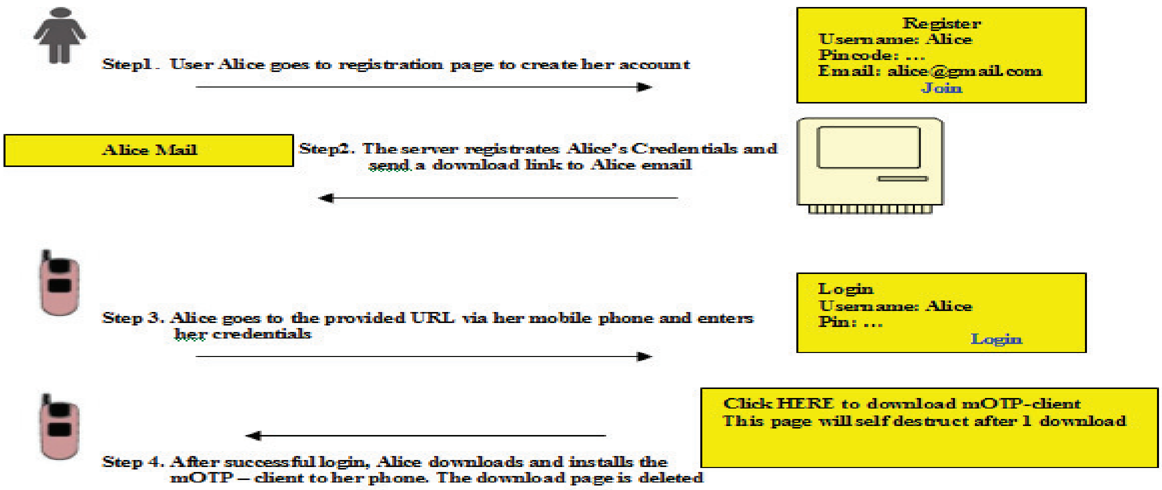

Step1. User Alice goes to registration page to create her account
Register
Username: Alice
Pincode: ...
Email: alice@gmail.com
Join
Alice Mail
Step2. The server registrates Alice's Credentials and send a download link to Alice email
Step 3. Alice goes to the provided URL via her mobile phone and enters her credentials
Login
Username: Alice
Pin: ...
Login
Click HERE to download mOTP-client
This page will self destruct after 1 download
Step 4. After successful login, Alice downloads and installs the mOTP – client to her phone. The download page is deleted

Figure 2. Registration Process

Pros

- No crucial login information is sent over the network.
- If the encrypted username and PIN-code is cracked it will not matter, since a hacker cant login without the correct Init-secret. And today's best encryptions (AES, RC4) have not been cracked yet.

- The Init-secret will be safe inside an application. If someone tries to manipulate the application along the way it can be detected by hash-function.
- No configurations needed for the client, the application is ready-to-go.
- Application downloaded and installed directly to the mobile phone, no need for user to connect the phone to the computer first.
- Extra security by requiring authentication to the download page.
- The download page can be deleted after 1 download, that way only one copy of the application/Init-secret will exist.
- Extra security by using two devices. If the client's computer is compromised and the traffic is being monitored by an attacker, perhaps your mobile phone will be safe.

Cons

- Harder to implement, more work for the server.
- If the process is not automated it can take a long time to get the server response.

IV.CONCLUSION

This paper have looked at the current security situation in cloud computing and how different cloud provider's have solved the issue of authenticating users that wish to use a service in the cloud. In a few cases static passwords have been used when logging in, and in other cases two-factor authentication with OTPs. In this paper we propose three different ways to securely and easy login to a cloud service using OTPs with the user's mobile phone as an authentication device. Furthermore, three different proposals for registrar new users to the cloud service have been made, that is secure and easy to use. With the authentication, registration and encryption method proposed and implemented in this paper, all of those factors are accomplished.

REFERENCES

[1] *" Analysis and Research of Cloud Computing System Instance",* S.Zhang & S.Zhang & X.Chen, Future Networks, 2010. ICFN'10. Second

[2] Intenational Conference. 22-24 Jan. 2010. page(s): 88

[3] *" The Comparism Between Cloud Computing and Grid Computing".* S.Zhang & X. Chen , Computer Application and System Modeling

[4] (ICCASM), 2010 International Conference. 22- 24 Oct . 2010. Page (s) : V11-72.

[5] http://motp.sourceforge.net/

[6] http://forums.tizag.com/showthread.php? t=10140

[7] http://zxing.appspot.com/generator/

[8] *" 2-clickAuth-Optical Challenge – Response Authentication",* A.Vapen , D. Byers , N. Shahmehri, 2010 International Conference on

[9] Availability , Reliability and Security.

[10] http://www.doityourself.com/video/Hacking-and-decrypting-SSL-and-TLS-traffic-27836970

[11] " Crytography and Security Services. Mechanisms and applications", M.Molollon, Cyertech Publishing, Newyork, 2007

[12] " Performance Analysis of Advance Encryption Standard (AES)", Y.X. Guizani & S. Bo Sun Hsiao- Hwa Chen Ruhai Wang,

[13] Global Telecommunications Conference, 2006, GLOBECOM '06.IEEE. page(s):1

[14] " A Novel Web Security Evaluation Model for a One – Time- Password System", B.Soh & A. Joy, WI 2003. Proceedings. IEEE/WIC

[15] International Conference, 13- 17 Oct. (2003),413

[16] " A Two- Factor Mobile Authentication Scheme for Secure Financial Transactions", R. Di Pietro & G. Me & M.A. Strangio, ICMB 2005. International Conference 11- 13 July (2005),28

[17] http://serchsecurity.techtarget.com/sDefinition/

[18] " Impersonation Attacks on Software- Only Two-Factor Authentication Schemes", T.Kwon, Communications Letters, IEEE Aug 2002. V6, page(s): 358

[19] " Facebook introduces one – time passwords" , D.Goodin

[20] http://googleenterprise.blogspot.com/2010/09/more-secure-cloud-for-millions-of.html

[21] http://aws.amazon.com/mfa/