VOL. 11, NO. 15, AUGUST 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

CREATION OF A MAPPING SYSTEM TO GENERATE OBJECTS SIMILAR TO BIT-COINS

Arun Karthik, Ashwin Joseph and R. Sethuraman
Department of Computer Science and Engineering, Faculty Computer Science and Engineering, Sathyabama University, India
E-Mail: arun.ronin@gmail.com

ABSTRACT

The usage and conversion of images into numerical values - similar to QR Bar Codes with the intention of creating a substantially complex character value that would initially percolate, depending on, predominantly heavy processing, in order to generate characters with the end results matching a particular user to user feed. The ability to, for example, in the typical social media spectrum, that is, currently present in our eco-system, would be an ideal way to illustrate how this might, make better sense. Suppose the image, or images, such as your profile picture, would be used for the metamorphosis of that particular image to correspond to your phone number, this in turn would create a secure and stable way to re-define one's vulnerability assets on said mobile device - every time there's a change, in a particular image. What would be true - is the notion that, other people, in general, have access or possess with reasonable certainty - the ability to see and view - both your picture and phone number, as it would generally be in the public domain when you call, or inside an application. Though, this drawback could be easily overcome, with the help of Service Providers who, would be the only ones that would with reasonable flexibility be able to utilize someone's network - whether GSM, CDMA, etc. to provide access to confidentially secured and locked networks along with the image at hand.

Keywords: mapping system, bit-coins, coding, algorithm.

INTRODUCTION

The general use of the mapping system would be to generate a database, of characters, that would be supplied to an all-purpose use filter which would regulate itself within the system.

The heavy- process specific limitation to generate these intermediary character codes would require the usage of a large amount of time, and effort, which overall would limit the interception of the characters.

The security of the system could be enhanced, modified, customized, and personalized with the added help of an interaction between the password/pin/lock, that the specific user has set to further increase the amount or level of complexity required/needed to crack the entire process. The term of the overall process would be a subset of the security that would be necessary to initialize.

The database, has to have a functional set-up in which, once a new entry is given, or if a photo or lock screen is changed/modified it would need to delete, by means of updating the database, the previous photos/lock-screen values, so that it would or can only take necessary values that are of current relevance.

There would be a singular process, occurring in the background, which would hold the previous database values for a limited period of time, this however, would only be until the new values generated from the phone no, lock screen, and profile photo are used to generate a new, unique code. Once, the new value to secure the device, is compiled or processed by the system (mobile device), then the previous code's history can be deleted from existence. The generation of each new code, would be allocated a minimum value in the memory in order to be able to finish it in a reasonable amount of time, but at the same time, the memory allocated would have to not be large enough, so that it doesn't affect those applications, and system applications that are used by the user on a regular and day-to-day basis.

The below process represents diagrammatically the above explain text and information in brief.

Working schematics

In the case of the images - such as the profile pictures, that would be in the form of an image of a specific format, they would need to undergo a certain degree of image processing in order to convert the image into a set of RGB values that would have a particular size ratio. Since, this methodology would exhibit a large amount of data, all the shade and values that are similar could be clubbed together, and made a one, in order to reduce the vastness of it all, at the same time reducing redundancy.

An external application would need to be developed. This application, would co-ordinate and function with the other applications in order to retrieve the necessary information. For example, the external application should be able to work with the system applications to retrieve permission first-hand to get access to the lock screen details set, as well as from the application that would house the picture that is set by the user as his/her/his-her profile picture and it would need to verify authenticity of the mobile user's phone no.

Some may wonder and question, yet even point to the notion that vulnerabilities of varying nature can be exploited, if an external entity or hacker accesses the external application that houses all this information.
Then the whole point of all this would become null and void.

In order to combat this, once the application is downloaded and installed, it would be designed to have no way to connect to the internet, to Bluetooth, or any other media or way of transfer, including USB access.

If an update for the application is available, the new version has to be downloaded, wherein which it would completely replace the old application, along with the information stored on it.

This processing and code generation would need to follow a specific and particular as determined by the designer and developer of the application.

The application would also need to store a dataset of values, that would map a small portion of the finally generated/iteration of the code, with that of the stored dataset. It would check all the dataset values, and if one out of the multiple values matches - success. If not, the application would need to be re-installed.

When re-installed, the application comes with a new set of dataset values from a database. This would allow it to match.

How matching occurs

The application would consist of a two different algorithms. One algorithm to process the first half of the profile picture, lock screen data and the phone no. This same algorithm would, also be responsible to generate a multiple set of values, but it would randomly select one value generated from the three and also store it in the dataset.

The second algorithm would be responsible for processing the second half of the code that would be part of the phone no, profile picture and the lock screen.
Hence, this mapping process would occur.

The first half of the final code generated, then would be checked in the background whenever the mobile device is powered on.

This wouldn't act as a visible security measure. It would indignantly act as a second-tier, substitution methodology to improve and enhance encryption already present in the mobile device and its various manifolds.

Coding and algorithm

The proposed intent would be to use the RC4 or a similar symmetric/private key algorithm that would aim to take the secret key as the phone number which would be authenticated by the service provider.

Separately, the whatsapp image would be able convert into strings of values that would consist of the RGB and alpha values. This would be subject to RC4.

Intended purpose of mobile security layer - simple enough to be implemented and not obvious enough to be seen, to be recognized and tampered with.

Also subject to the RC4, the RC4 cipher consists of two parts: 1. The Key Scheduling Algorithm (KSA), and 2. The Pseudo Random (Byte) Generation Algorithm (PRGA) – the phone lock/lock screen.

KSA:
for(d=0; d<256; d++) S[d] = d; // initialize array
e = 0
for(d=0; d<256;d++){
e = (e + S[d] + key [d mod keylength]) mod 256;
swap(S[d], S[e]);
}

PRGA:
d = 0; e = 0;
do{
d = (d + 1) mod 256;
e = (e + S[d]) mod 256;
swap(S[d], S[e]);
F = S[(S[d] + S[e]) mod 256];
output F;
} while (required);

Concurrent interim process

While the images are temporarily stored in the database, they would also be subjected to a randomized storage location, meaning this would further secure their cohesiveness within a time space that would isolate vulnerabilities. This would cause the infiltration of the database from a long-term perspective to be of no particular value to the infiltrator/perpetrator. The database on the other hand, would be in-charge of restructuring itself, and only when valid users of a particular dataset try to access their particular information would it open up to release the un-randomized information.

CONCLUSIONS

The decisive conclusion is that, for added security or encryption on your mobile devices, which in the age of the internet, which is essential, we would require better ideas in order to protect one's private data and information. The above tries to address these concerns, by creating a more flexible and automatic way to decrease vulnerabilities.

Though, the RC4 isn't particular flawless, and in natures' course has found to have inconsistence and loop holes. The real fact still remains, with respect to its simplicity and adaptability in the landscape of different environments that already are in use. Provided with the added physical layers that have been proposed and examined, the real nature of the assets may very well shine, as they aim to assist and facilitate the use and working of the RC4 algorithm to a better extent. Causing fewer lapses, and better security in a 256-bit environment.

In terms of short-term benefits, it's very adaptive and can easily be programmed to make change, very well making maintenance uncomplicated and efficient. This added to the code present set the bar for the well

Further it pertains to increase the number of levels or layers with which a hacker or unauthorized personnel have to pass through before they reach a destination of scrambled or meaningless data.

Adding to this, one might question the time taken or the amount of processing require for the mobile to have the need to do all these background tasks as a pre-requisite, but though it requires adequate processing, in the age of faster processor, and faster mobile devices, it would harder hurt or hinder the overall cause, when it comes to the overall price of protecting and safeguarding our information from the wrong hands.

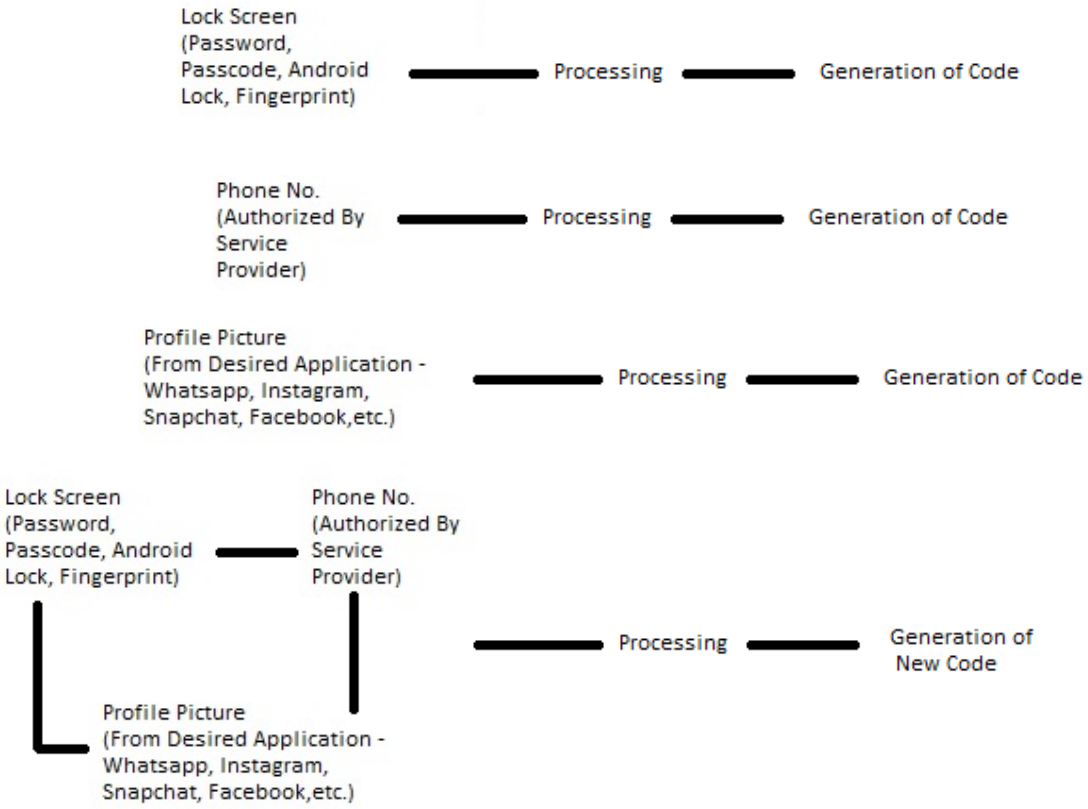

Figure-1. Basic process/methodology.

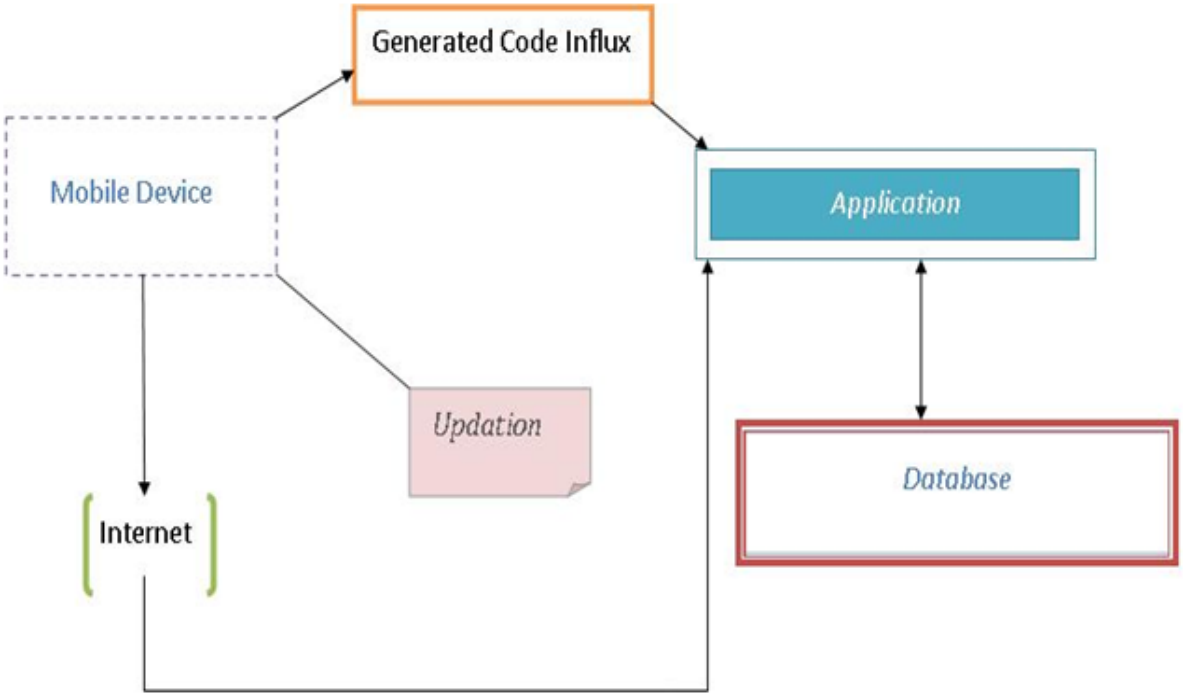

Figure-2. Architecture of system and its components.

REFERENCES

T. D. B. Weerasinghe. An effective RC4 stream cipher. 10.1109/ICIInfS.2013.6731957, Industrial and Information Systems (ICIIS), 2013 8th IEEE International Conference.

Feng Qiu Nat. Eng. Lab. for Next Generation Internet Interconnection Devices, Bejing Jiaotong Univ, Miao Xue; Yajuan Qin; Hongke Zhan. A distributed mapping system to support mobility in identifier/locator separation architecture, 10.1109/ICNIDC.2009.5360982, Network Infrastructure and Digital Content, 2009. IC-NIDC 2009. IEEE International Conference.

P. Singh Comput. Sci. and Inf. Security, NITK Surathkal, Mangalore, India, B. R. Chandavarkar; S. Arora; N. Agrawal. Performance Comparison of Executing Fast Transactions in Bitcoin Network Using Verifiable Code Execution. 10.1109/ADCONS.20.Advanced Computing, Networking and Security (ADCONS), 2013 2nd International Conference.

Jianjun Chen, Dept. of Intell. Syst. Design Eng., Toyama Prefectural Univ., Imizu, Japan, N. Takagi. A Pattern Recognition Method for Automating Tactile Graphics Translation from Hand-Drawn Maps. 10.1109/SMC.2013.711.Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference.

I. Alqassem Electr. Eng. and Comput. Sci., Masdar Inst. of Sci. and Technol., Abu Dhabi, United Arab Emirates, D. Svetinovic. Towards Reference Architecture for Cryptocurrencies: Bitcoin Architectural Analysis. 10.1109/iThings.2014.78. Internet of Things (iThings), 2014 IEEE International Conference on, and Green Computing and Communications (GreenCom), IEEE and Cyber.