Software Puzzle: A Counter Measure To Online Security Attacks

Aravarasan P[1], Kevin George A[2]
[1, 2] Dept of Computer Science
[1, 2] KCG College o Technology

Abstract- *Many security primitives are based on hard mathematical problems. Using hard AI problems for security is emerging as an exciting new paradigm, but has been underexplored. In this paper, we present a new security primitive based on hard AI problems, namely, a novel family of graphical password systems built on top of Puzzle technology, which we call Puzzle as graphical passwords (CaPGP). CaPGP is both a Puzzle and a graphical password scheme. CaPGP addresses a number of security problems altogether, such as online guessing attacks, relay attacks, and, if combined with dual-view technologies, shoulder-surfing attacks. Notably, a CaPGP password can be found only probabilistically by automatic online. Guessing attacks even if the password is in the search set. CaPGP also offers a novel approach to address the well-known image hotspot problem in popular graphical password systems, such as Pass Points, that often leads to weak password choices. CaPGP is not a panacea, but it offers reasonable security and usability and appears to fit well with some practical applications for improving online security.*

Keywords- Graphical Password System, Puzzle, Online Guessing Attack, PassPoints, Online Security.

I. INTRODUCTION

Security primitives are based on hard mathematical problems. Using hard AI problems for security is emerging as an exciting new paradigm, but has been underexplored. A fundamental task in security is to create cryptographic primitives based on hard mathematical problems that are computationally intractable. This paradigm has only achieved a limited success when compared to the cryptographic primitives We present a new security primitive based on hard AI problems, namely, a novel family of graphical password systems built on top of Puzzle technology, which we call Puzzle as graphical passwords (CaPGP). CaPGP is both a Puzzle and a graphical password scheme. CaPGP addresses a number of security problems altogether, such as online guessing attacks, relay attacks, and, if combined with dual-view technologies, shoulder-surfing attacks. Notably, a CaPGP password can be found only probabilistically by automatic online guessing attacks even if the password is in the search set. CaPGP also offers a novel approach to address the well-known image hotspot problem in popular graphical password systems, such as Pass Points, that often leads to weak password choices. CaPGP is not a panacea, but it offers reasonable security and usability and appears to fit well with some practical applications for improving online security**.** We present exemplary CaPGPs built on both text Puzzle and image-recognition Puzzle. One of them is a text CaPGP wherein a password is a sequence of characters like a text password, but entered by clicking the right character sequence on CaPGP images. CaPGP offers protection against online dictionary attacks on passwords, which have been for long time a major security threat for various online services. This threat is widespread and considered as a top cyber security risk. Defense against online dictionary attacks is a more subtle problem than it might appear.

II. RELATED WORK

To improve the security of the domain number of new methodologies have been found. One among them is the use of captchas. Captchas are used to identify humans from robots. This is used to stop computer programs developed by hackers from trying to break into a domain. Another method used widely is the use of puzzles. These puzzles are used as an additional layer of security. These puzzles are built on top of the available username and password system. These puzzles are provided during the time of registration and the user chooses the puzzle to be used at the time of login.

Security primitives are based on hard AI problems. Using hard AI problems for security is emerging as a new paradigm, but has been underexplored.

A fundamental task in security is to create cryptographic primitives based on hard mathematical problems that are computationally intractable.

This paradigm has achieved just a limited success as compared with the cryptographic primitives based on the hard mathematics problems and their applications. Using hard artificial intelligence problems for security, initially proposed is an exciting new paradigm. Under this paradigm the most

notable primitive invented is puzzle, which distinguishes human users from computers by providing a challenge.

III. PROPOSED SYSTEM

We present a security primitive based on hard AI problems namely a novel family of graphical passwords system built on top of puzzle technology which we call as Puzzle as Graphical passwords. This is system which involves both puzzle and a graphical password. It addresses a number of security problems altogether, such as online guessing attacks, relay attacks. The password in this system can be found only probabilistically by online surfing attacks even if the password is in the search set. It also provides a novel approach to address the well known image hotspot problem in popular graphical password systems such as pass points, that often leads to weak password choices. This offers reasonable security and usability and appears to fit well with some practical applications for improving online security. We present a system which is built on both text puzzle and image puzzle. One of them is a text wherein a password is a sequence of characters like a text password, but entered by clicking on the right character sequence of characters on images. It offers protection against online dictionary attacks on passwords, which have been a major security threat for various online services. This threat is widespread and is considered as a top cyber security risk. Defense against online guessing attacks is a more subtle problem than it might appear.

The main advantage of this system is that it offers reasonable security and usability and appears to fit well with some practical applications. Defence against online guessing attack is more a subtle program than it looks. The key difference when compared with other systems is that it uses puzzle technology and image solving. The system has four modules.

A. PUZZLE LOGIN

The security and usability problems in text-based Login And password schemes have resulted in the development of Puzzle password schemes as a possible alternative.

We can visualize the sum 1+2+3+...+n as a triangle of character. Numbers which have such a pattern of character are called Triangle (or triangular) numbers, written T(n), the sum of the integers from 1 to n time Using Factorial base Login Puzzle Solving.

B. RANDOM CAPTCHA SELECTION

A CAPTCHA is a test that is used to separate humans and machines. CAPTCHA stands for "Completely Automated Turing test to tell Computers and Humans Apart." It is normally an image test or a simple mathematics problem which a human can read or solve, but a computer cannot. It is made to stop computer hackers from using a program to automatically set up hundreds of accounts, such as email accounts. It is named after mathematician.

Each individual is chosen randomly and entirely by chance, such that each individual has the same probability of being chosen at any stage during the sampling process, and each subset of n individuals has the same probability of being chosen for the sample as any other subset of n individuals This process and technique is known as simple random sampling, and should not be confused with systematic random sampling. A simple random sample is an unbiased surveying technique.

C. IMAGE PUZZLE SOLVING

we study how to prevent DoS/DDoSattackers from inflating their puzzle-solving capabilities. To this end, we introduce a new client puzzle referred to as software puzzle. Unlike the existing client puzzle schemes, which publish their puzzle algorithms in advance, a puzzle algorithm in the present software puzzle scheme is randomly generated only after a client request is received at the server side and the algorithm is generated such that:

1) an attacker is unable to prepare an implementation to solve the puzzle in advance

2) the attacker needs considerable effort in translating a central processing unit puzzle software to its functionally equivalent GPU version such that the translation cannot be done in real time. Moreover, we show how to implement software puzzle in the generic server-browser model.

D. OTP GENERATION

A one-time password (OTP) is a password that is valid for only one login session or transaction, on a computer system or other digital device. OTPs avoid a number of shortcomings that are associated with traditional (static) password-based authentication; a number of implementations

also incorporate two factor authentication by ensuring that the one-time password requires access to something a person has (such as a small keyring fob device with the OTP calculator built into it, or a smartcard or specific cellphone) as well as something a person knows (such as a PIN).

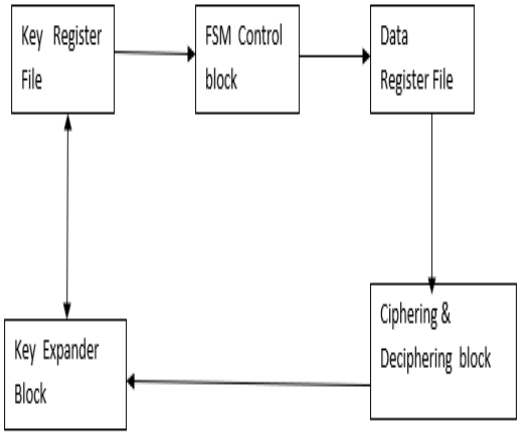
V. ALGORITHM

AES – ADVANCED ENCRYPTION STANDARD

AES is based on a design principle known as a Substitution permutation network. It is fast in both software and hardware. Unlike its predecessor, DES, AES does not use a Feistel network. AES has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits, whereas Rijndael can be specified with block and key sizes in any multiple of 32 bits, with a minimum of 128 bits. The block size has a maximum of 256 bits, but the key size has no theoretical maximum. AES operates on a 4×4 column-major order matrix of bytes, termed the *state* (versions of Rijndael with a larger block size have additional columns in the state). Most AES calculations are done in a special finite field. The AES cipher is specified as a number of repetitions of transformation rounds that convert the input plaintext into the final output of ciphertext. Each round consists of several processing steps, including one that depends on the encryption key. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same encryption key.

HIGH-LEVEL DESCRIPTION OF THE ALGORITHM

1. KeyExpansion—round keys are derived from the cipher key using Rijndael's key schedule
2. Initial Round
    1. AddRoundKey—each byte of the state is combined with the round key using bitwise xor
3. Rounds
    1. SubBytes—a non-linear substitution step where each byte is replaced with another according to lookup.
    2. ShiftRows—a transposition step where each row of the state is shifted cyclically a certain number of steps.
    3. MixColumns—a mixing operation which operates on the columns of the state, combining the four bytes in each column.
    4. AddRoundKey
4. Final Round (no MixColumns)
    1. SubBytes
    2. ShiftRows
    3. AddRoundKey



VI. METHODOLOGY

This project focuses on the security of the system hence the user must have access to the domain. They must have an account in the particular domain. To create an account an user must provide his details. This is done during the registration process. They will provide all the basic details other details too. These additional details are used to create the appropriate captcha and the image puzzle which are used during the login process. The successful completion of the registration process will send a mail to the corresponding users mail id. The mail will contain the user id which will be used to login into the particular domain.

During the login process the user uses the user id and the password which is unique to each individual. This is the first layer of the login process. Successful completion of the first layer will direct you to the captcha page. Here there will be captchas to be solved. These are image captchas. These captchas must be chosen perfectly to reach the next layer. Any difficulties during the solving of captchas will send an alert to the users mail id, and the login process will be terminated.

The next layer is the image puzzle solving layer. In this layer an image puzzle chosen during the registration process must be solve. To solve the puzzle a limited time will be given. Unable to solve the puzzle will terminate the login process and send an alert mail to the user. Completion of all these layers will send an OTP to the users mail. Once the OTP is verified the system will allow into the system.

VII. CONCLUSION

The software puzzle may be built upon a data puzzle,it can be integrated with any existing server-side data puzzle scheme, and easily deployed as the present client

puzzle schemes do. CAPTHCHA is widely research field act as internet rectifier to secure web applications by discern human from bots. CAPTCHA presented which will improve resistance of math calculus CAPTCHA. By use, Boolean operations and expressions instead of trigonometric and differential function which will help in reduce the complexity of CAPTCHA and help to achieve better usability and security as compared to math calculus CAPTCHA. Boolean CAPTCHAcan be easily use by educated user. No need of technical skill, by using intellectual mind to solve this CAPTCHA and help to reduce time complexity.

REFERENCES

[1] Yongdong Wu, Zhigang Zhao, Feng Bao, Robert H. Deng, " Software Puzzle: A Counter Measure to Resource Inflated Denial of Service Attaks ", IEEE Transactions on Inormation Forensics and Security year 2015, vol 10, issue 1

[2] MehmudAbliz, TaiebZnati " A Guided Tour Puzzle for Denial of Service Prevention" Annual computer security applications conference, year 2009.

[3] Sophie McGill Smith, Richard Green " Jigsaw Puzzle Solver to Locate Pieces Positions" International conference on Image and Vision Computing, year 2019

[4] YoppySazaki ;HadipurnawanSatria ; AngginaPrimanita ; ReziApriliansyah, "Application of the Steepest Ascent Hill Climbing Algorithm in the Preparation of the Crossword Puzzle Board", 2018 4th International Conference on Wireless and Telematics (ICWT)

[5] Ning Chen ;Sunghun Kim, "Puzzle-based automatic testing: bringing humans into the loop by solving puzzles", 2012 Proceedings of the 27th IEEE/ACM International Conference on Automated Software Engineering

[6] J. Griesbach ; J. Wiejaczka ; R. Frangopol ; F. Harsono ; D. Etter, "The Puzzle Project: a case study in multimedia signal processing", 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing