Transparent Two Factor Authentication

Seminar Report

submitted in partial fulfillment of the requirement
for award of Degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

of

APJ Abdul Kalam Technological University

Submitted by

JIFINI ANN JOSE

Department of Computer Science and Engineering
Mar Athanasius College of Engneering
Kothamangalam

Mar Athanasius College of Engineering

KOTHAMANGALAM

Transparent Two Factor Authentication

Bonafide record of seminar done by

JIFINI ANN JOSE
Register Number: MAC15CS032

submitted in partial fulfillment of the requirement
for the Degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

of

APJ Abdul Kalam Technological University

....
Prof Joby George
Seminar Coordinator

....
Dr. Surekha Mariam Varghese
Head of the Department

....
Prof Neethu Subhash
Seminar Coordinator

ACKNOWLEDGEMENT

First and foremost, I sincerely thank the 'God Almighty' for his grace for the successful and timely completion of the seminar.

I express my sincere gratitude and thanks to Dr. Solly George, Principal and Dr. Surekha Mariam Varghese, Head Of the Department for providing the necessary facilities and their encouragement and support.

I owe special thanks to the staff-in-charge Prof. Joby George and Prof. Neethu Subhash for their corrections, suggestions and sincere efforts to co-ordinate the seminar under a tight schedule.

I express my sincere thanks to staff members in the Department of Computer Science and Engineering who have taken sincere efforts in helping me to conduct this seminar.

Finally, I would like to acknowledge the heartfelt efforts, comments, criticisms, co-operation and tremendous support given to me by my dear friends during the preparation of the seminar and also during the presentation without whose support this work would have been all the more difficult to accomplish.

ABSTRACT

Traditional username and password-based single factor authentication is easy to deploy. But vulnerable to dictionary attacks, snooping and brute force attacks. Two-Factor Authentication (2FA) has been proposed to improve the security. The smart devices are used as the second authentication factor. However, the interaction between human and the smart device is required, which is inconvenient to users. . In addition, an attacker is able to get the second authentication factor through fraud, thus invalidating current 2FA mechanisms In order to solve these problems, a Transparent Two-Factor Authentication (T2FA) based on Physical Unclonable Function (PUF) and voiceprint is proposed. The second authentication authenticates the user's mobile phone through the PUF. The third one is to determine whether the login terminal and the user's mobile phone are in the same environment with the environment voiceprint. The second and third authentication is completely transparent to users. Therefore, T2FA avoids the tedious interaction and provides the same high user experience satisfaction as the single-factor authentication and exhibits high security simultaneously.

CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATION

T2FA	Transparent Two Factor Authentication
2FA	Two Factor Authentication
PUF	Physically Unclonable Functions
NFC	Near Field Communication
API	Application Program Interface
GPS	Global Positioning System
FPGA	Field Programmable Gate Array
CRPs	Challenge Response Pair
LUTs	Look Up Table

Introduction

With the rapid development of e-commerce, more and more companies conduct business online. However, due to the openness, versatility and multi-service of the Internet, network security issues become more and more serious. According to the statistics from China Internet Network Information Center, in 2015, 42.7% of Internet users encountered network security problems.

User authentication is to establish the trust between users and devices and has become the most forefront defense for cyber-security. The static single-factor authentication such as "username + password" has been used widely because it is easy to deploy without additional devices. However, the security of single-factor authentication depends on the password. Such authentication is effective in the early Internet, where remote access was not used widely and the attack pattern was single. Nowadays, Trojans are able to intercept the user's keyboard record and even decrypt the user's login account and password by collecting the location clicked by the mouse, thus breaking the password protection technology. In 2011, Chinese internet suffers the most serious user data leak in history. China's largest software programmers' web site China Software Developer Network was hacked, and account information for more than 6 million users was leaked and quickly spread via the Internet . In 2015, accounts and passwords disclose became the second serious Internet security incident in computers and mobile phones, accounting for 22.9%. In May 2016, Google announced to completely cancel the password . Therefore, single-factor authentication has become increasingly unsuitable.

In order to enhance the security, Two-Factor Authentication was proposed. 2FA is a method of confirming a user's claimed identity by utilizing a combination of two different factors: 1) something they know, 2) something they have, or 3) something they are, i.e., a combination of passwords and physical entities such as smart cards, mobile phones, tokens, or fingerprints. However, compared with the password-based single-factor authentication, two-factor authentication brings inconvenience to users when a physical entity is used as the second authentication factor, where many additional operation steps are added. For example, the dynamic token method is a one-time password and provides the high security, but it requires carrying different tokens when the user visits different sites.

In addition, current 2FA approaches are being questioned. For example, McAfee and Guardian Analytics released a joint report titled "Dissecting Operation High Roller". It mentioned an international criminal group who used an automated operation to attempt to steal large sums of money through unauthorized and fraudulent transfers. Since criminal group implanted malicious software on the victim's computer, the password information could be easily stolen. The theft of passwords was then integrated into the process for automated attacks. Criminals even could manipulate the user's authentication process so that the two-factor authentication tokens used to verify bank account access authorization became useless.

There are many other reports that the two-factor authentication is threatened, and even fraud can be achieved without any technology. In 2016, China Central Television (CCTV) reported a piece of news about telecom fraud on the two-factor authentication. Even with the two-factor authentication, criminals can still steal all assets of the victim easily through the unpopular business of China Mobile's online 4G card replacement . The root cause of this event is that the current authentication process of the second factor involves the user interactions, and the criminals can obtain the second authentication factor by fraudulent means so as to complete the authentication process.

As discussed above, the traditional single-factor authentication faces a great threat that once the password is leaked, the user will have no security at all. In addition, although current two-factor authentications are able to improve security, the tedious interactions between users and entities are added, which not only reduces the user experience significantly, but also makes it vulnerable to fraud and other threats.

In order to mitigate the above issues, we propose the concept of Transparent Two-Factor Authentication (T2FA), and propose the first T2FA technique based on physical unclonable functions and voiceprint. The second factor in the T2FA consists of two authentication.

1. Legal verification of smart devices by using PUF.
2. Match of the same environment between the computer and the user's mobile phone when the user logs in.

Without changing the user experience, we propose the PUF-based device authentication and the similarity comparison of the environment background sound between the mobile phone and the computer. In T2FA,the second factor is completely transparent to the user, which avoids the tedious interaction between the user and the device. As there is no need for users to participate in the authentication of the second factor, the human-machine interaction and the threat of user-oriented fraud can be eliminated technically. Therefore, T2FA is able to improve user experience and enhance the authentication security significantly.

Related Works

2.0.1 Hardware Token

Time-based hardware tokens follow the standard of 2FA. The SecurID is stored as the second factor in the dongle.However, this mechanism requires the interaction between the user and the hardware token. Besides, the server needs to provide each client with a separate hardware token, which makes the deployment of token expensive. It also requires each site having its own authentication devices and needs users to carry them. Therefore, such 2FA mechanism is only used in e-government, online banking and other similar fields.

2.0.2 Software Token

Google 2-Step authentication is a typical software token mechanism based on user's phone and authentication code. The software authentication code is sent by short message service or an application running on the mobile phone. Such mechanism needs the user to copy the authentication code on the mobile phone to the login interface of the browser. Sound-Proof uses the sound of the environment as an authentication factor, but it does not consider the legality authentication of the mobile phone and suffers security issues in a silent environment.

2.0.3 Short Range Wireless Communication

Short-range wireless communication implements the second factor authentication with Bluetooth, NFC or WiFi, which is able to reduce the interaction between the user and mobile phone. Bluetooth [13] is the most widely used technique. The browser and the phone generate a challenge-response pair via Bluetooth to compute the distance between them. However, the Bluetooth API is no longer supported by current mainstream browsers. NFC is a new short-range wireless communication method for smart devices. However, the mainstream browsers do not provide APIs to support NFC devices, and most of personal computers are not integrated with NFC. Moreover, NFC requires the user to hold his mobile phone to complete the two-factor authentication. Therefore, the interaction between the user and the mobile phone is complicated. WiFi communication requires that the computer located in the same network as the mobile phone. The computer needs to use additional software to generate an access node so that the user's mobile phone can be connected to the network where the computer is located. Such method requires that the mobile application continuously monitors the login request sent by the browser, which would degrade the performance of smart devices.

2.0.4 Short Range Ultrasonic

The browser and mobile phone can communicate with each other by the short-range ultrasound which can be recognizable to the computer and not audible to the human ear. Due to the limited performance of the phone, this communication method can only produce highly directional near-field ultra-high frequency sound waves, and the signal attenuation is quite fast. When this technique is deployed for authentication, mobile phones cannot use external devices such as earphones, and such high-frequency sound wave will have a health impact on children and animals.

2.0.5 Location Information

With the Global Position System (GPS) information, the server can detect whether the computer and phone are located in the same environment. GPS sensors have been deployed in mainstream smartphones but not in most of computers. Many APIs provided by the browser can obtain geolocation information to complete the login. However, the geolocation information is not accurate. For example, when the device is located in a VPN network or an enterprise's large management network, the geolocation is not the real location of the device and can be obtained by attackers, which would result in an authentication failure.

The Proposed Method

The traditional single-factor authentication is based on username and password, which is easy to deploy but vulnerable to dictionary attacks, snooping and brute force attacks. Although using different passwords on different websites can improve the security, it brings a lot of trouble to users in terms of memory. In the urgent need to design a more secure authentication mechanism, two-factor authentication (2FA) comes out, combining the password with the entities such as credit card, mobile phone, token or fingerprint. However, as visiting different sites often require different tokens, when you need to visit many sites at the same time, carrying a long list of tokens will be very troublesome. Therefore, compared with the single-factor authentication, two-factor authentication will bring more inconvenience to users when using different physical entities as authentication factors. Besides, the interactive between human and entities may allow an attacker to obtain the second factor through fraud.

PUF and voiceprint-based T2FA aims to add the authentication of mobile phones and the feature comparison of environmental background sounds without changing the user experience. The authentication is completely transparent to the user, which means that the user only needs to enter the user name and password. This not only enhances the security of the single-factor authentication, but also addresses the security issue that the traditional two-factor authentication requires the user interaction. Therefore, T2FA owns the anti-fraud ability that previous 2FA techniques do not have.

This paper propose the first transparent two-factor authentication based on PUF and voiceprint. The function of the browser side is implemented for the login and registration, environment sound recording, audio file encryption and data transmission. The server side is used for the authentication of mobile phones, database access, and data transmission. At the same time, the server stores the PUF's challenge-response pairs or PUF's delay parameters which are obtained by machine learning such as deep neural network for authentication. The mobile phone is used for recording environment sound, audio decryption, data transmission and audio similarity comparison. The whole authentication flow is shown in Figure(authentication flow diagram).

3.1 Design Of T2FA

T2FA is composed of login/registration module, browser recording module, mobile phone recording module, PUF module, audio encryption and decryption module, data transmission module, database access module and audio comparison module. The functional modules in the T2FA system are shown in Figure 3.1.

1. Login/Registration Module: This module is responsible for the authentication and transmission of input information when logging in or registering.

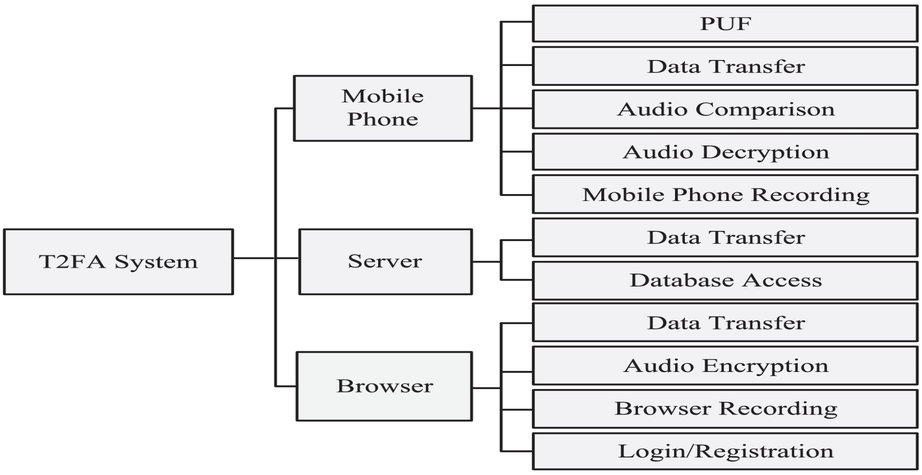


Figure 3.1: Functional Modules in Proposed T2FA System

2. Browser Recording Module: This module is responsible for collecting the environment sound around the user's computer after the server passes the user name and password authentication.

3. Mobile Phone Recording Module: This module is responsible for collecting the environment sound around the user's mobile phone after the server passes the user name and password authentication.

4. PUF Module: This module is responsible for the authentication of the user's mobile phone after the server passes the user name and password authentication.

5. Audio Encryption and Decryption Module: This module is responsible for encrypting the audio files collected by the browser and transmitting the encrypted file to the server.

6. Data Transfer Module: This module is responsible for transferring data and instructions between browser, server and mobile phone.

7. Database Access Module: This module is responsible for querying or inserting data in the database during user login or registration.

8. Audio Comparison Module: This module is responsible for extracting the mark information in the two audio time domains and comparing the similarity of the two pieces.

The whole authentication flow is shown in Figure 3.2.

1. The browser sends the username and password to the server.

2. The server verifies whether the user name and password are legal. If legal, the server will send a PUF challenge to the mobile phone for authentication. If the PUF response equals with the response stored in the server, the record command will be sent to the browser and mobile phone. If illegal, the server will give a warning message to the browser.

3. The browser and phone start recording 3 seconds environment sound, respectively.

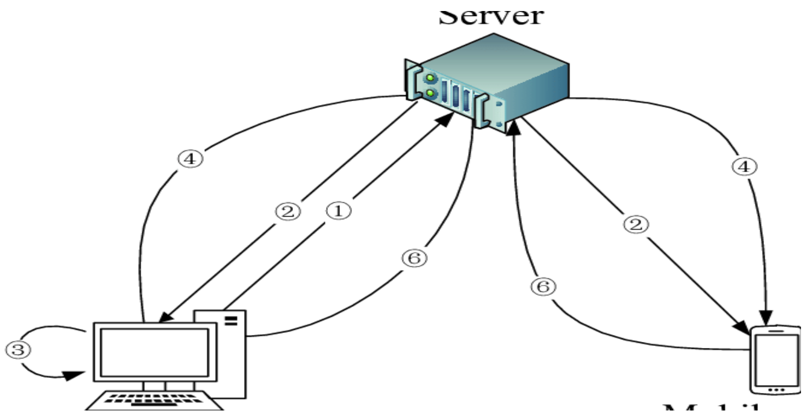


Figure 3.2: Authentication flow of T2FA

4. The browser encrypts and sends the recorded environment sound file to the server, and then the server transmits the encrypted environment sound file to the mobile phone.

5. The mobile phone decrypts the received environment sound file and compares the similarity between the two audio files.

6. The mobile phone sends the result of audio similarity comparison to the server. Then the server transmits the result to the browser.

7. The browser determines whether the login is admitted.

3.2 Voiceprint

The similarity computation for two sound segments is similar to the audio fingerprint and automatic media retrieval. For media retrieval, it may be detrimental to the match of the similar audio in the database when using a noisy audio to compare. Therefore, we can only extract the specific attributes related to the noisy audio and use them to compare with the sample that needs to be matched. When the background sound is noisy or thin, the extracted attributes must be robust. In the automatic media retrieval, high-frequency spectral coefficients, microwaves, and peak frequencies can be used as robust attributes. Because of the time misplacement problem, these attributes should focus on the frequency domain in which the audio samples are located. We compare two aligned audio samples by extracting relevant information from their time domain, and calculate the similarity between two audio samples using cross correlation and error energy.

3.2.1 Cross-Correlation

Cross-correlation is a common method used to calculate the similarity of audios sampled at the same time. We use x and y to represent n discrete points of the same two consecutive signals. The cross-correlation $C_{x,y}(l)$ of the two audio segments is calculated by the function log l [0, n 1]:

$$C_{x,y}(l) = \sum_{n=1}^{n-1} x(i).y(i-1)(i < 0|i > n1, y(i) = 0)$$

In order to accommodate the signal with different amplitudes, the expression of cross-correlation can be normalized to the Equation 2:

$$C'_{x,y}(l) = \frac{}{C'_{x,y}(l)}\sqrt{C_{x,x}(0)C_{y,y}(0)}$$

where

$$C_{x,y}(l)$$

represents the auto correlation coefficient. The range of normalization unit

$$C'_{x,y}(l)$$

is [1,1].

$$C'_{x,y}(l)$$

= 1 means that even if the amplitudes of the two signals are different, the two audio segments have the same graphic structure;

$$C'_{x,y}(l)$$

= 1 means that the two audio segments have the same graphic structure but they are completely opposite signals;

$$C'_{x,y}(l)$$

= 0 means that the two signals are irrelevant. If the values of the two signals are unknown, we can use the absolute of the cross-correlation maximum value as the similarity. In addition, the complexity of calculating Cx,y(‘) can be reduced to calculating Cx,y(‘) = F 1 (F(x) ·F(y)), where F() represents a Fourier transform, and * represents a complex pairing operation.

3.2.2 Error Energy

The error energy can be used to measure the similarity of waveforms, which is similar to those used to determine the orthogonality of functions in advanced mathematics. Assuming that the two signals are x(t) and y(t), we can choose a constant a to make a · y(t) approach x(t). The error energy can be expressed using the integral of (x(t) a · y(t)) in the time domain. The constant a must be selected to ensure that the energy error is minimized. By taking the derivative and extreme value of the functions, we can know that it can meet the condition when a is the integral ratio of x(t) · y(t) to y(t) · y(t) in the time domain. The correlation coefficient between x(t) and y(t) is defined as Pxy. And the difference between (Pxy) 2 and 1 is the relative error energy, i.e., the ratio of error energy to the integral ratio of x(t) · x(t) in the time domain. The numerator of the equation for Pxy is the integral of x(t)· y(t) in the time domain. The denominator is the square root of the integral of the x(t) 2 and y(t) 2 in the time domain. It can be mathematically proved that the modulus of the numerator is smaller than the denominator, that is, the modulus of the correlation coefficient Pxy will not be greater than 1. Since the energy is fixed for a signal, the magnitude of the correlation coefficient Pxy is only determined by the integral of x(t) · y(t) in the time domain. If the amplitude and time of two waveforms are independent, we can get x(t) · y(t) = 0, and the integral result is also 0. Therefore, when the correlation coefficient is 0, the similarity is the worst. When the correlation coefficient is 1, the error energy is 0, which indicates that the similarity between the two signals is good and linearly related.

3.3 Physically Unclonable Function For Authentication

3.3.1 Physically Unclonable Function

Physical unclonable functions (PUFs) are a promising innovative primitive that are used for authentication and secret key storage without the requirement of secure EEPROMs and other expensive hardware.This is possible, because instead of storing secrets in digital memory, PUFs derive a secret from the physical characteristics of the integrated circuit (IC).This approach is advantageous over standard secure digital storage for several reasons:

- PUF hardware uses simple digital circuits that are easy to fabricate and consume less power and area than EEPROM/RAM solutions with antitamper circuitry. In addition, simple PUF applications do not require expensive cryptographic hardware such as the secure hash algorithm (SHA) or a public/private key encryption algorithm.
- Since the "secret" is derived from physical characteristics of the IC, the chip must be powered on for the secret to reside in digital memory. Any physical attack attempting to extract digital information from the chip, therefore, must do so while the chip is powered on.
- Invasive attacks are more difficult to execute without modifying the physical characteristics from which the secret is derived. Therefore, continually powered active antitamper mechanisms are not required to secure the PUF.
- Nonvolatile memory is more expensive to manufacture. EEPROMs require additional mask layers, and battery-backed RAMs require an external always-on power source.

3.3.2 Types Of PUFs

The two primary applications of PUFs are for low-cost authentication and secure key generation. These two applications have resulted from the fact that PUFs designed during the past decade have mostly fallen into two broad categories described as "strong PUFs" and "weak

PUFs." Strong PUFs are typically used for authentication, while weak PUFs are used for key storage.

Each PUF can be modeled as a black-box challenge– response system. In other words, a PUF is passed an input challenge c, and returns a response r=f(c), where f(.) describes the input/output relations of the PUF. The blackbox model is appropriate here, because the internal parameters of f(.) are hidden from the user since they represent the internal manufacturing variability that the PUF uses to generate a unique challenge–response set. Such parameters would include the variability of a circuit's internal gate delay as described in the Introduction. PUF security relies on the difficulty of measurement or estimation of these parameters as well as the difficulty of manufacturing two chips with the same set of parameters. The fundamental difference between weak and strong PUFs is the domain of f, or informally, the number of unique challenges c that the PUF can process. A weak PUF can only support a small number of challenges (in some cases only a single challenge). A strong PUF can support a large enough number of challenges such that complete determination/measurement of all challenge–response pairs (CRPs) within a limited timeframe is not feasible.

Weak PUF Model

The first class of PUFs leveraging manufacturing variability are weak PUFs [also known as physically obfuscated keys (POKs)]. These PUFs can be thought of as PUFs that directly digitize some "fingerprint" of the circuit. This direct measurement results in a digital signature that can be used for cryptographic purposes. Because the fingerprint signature remains largely invariant, this means that the PUF can only be interrogated by one or a small number of challenges. In the above black-box description, this corresponds to f having a domain of one or only a small number of inputs. Correspondingly, f will also have a very small range, as a given challenge should always result in the same response (ignoring noise, which is considered later). One can clearly use several instances of the above black box to support more CRPs or response bits. However, this is still considered a weak PUF, because the number of responses is linearly related to the number of components subject to manufacturing variation. Explicitly stated, weak PUFs have the following properties:

- a small number of CRPs (linearly related to the number of components whose behavior depends on manufacturing variation);
- response is stable and robust to environmental conditions and multiple readings so that a challenge always yields the same response;
- responses are unpredictable and depend strongly on the innate manufacturing variability of the device;
- it is impractical to manufacture two devices with the same physical fingerprint

An example weak PUF is the power-on state of an SRAM. Although a SRAM cell is symmetric, manufacturing variability will give each cell a tendency toward a logical "1" or "0" at power-on. This variability is random across the entire SRAM, giving it a unique fingerprint on power-on that can be identified. In this case, if the "response" consists of the entire SRAM state at power-on, the notion of a “challenge” is not useful, as there is only one possible “challenge”: powering on the SRAM. The output signature is always the same (ignoring noise). One can allow for more output bits by increasing the size of the SRAM, but the response space is still linearly related to the number of components subject to manufacturing variation (each SRAM cell). The SRAM is an extreme example of a weak PUF in the sense that it only has one “CRP.” Note that since weak PUFs in general have only a small number of CRPs, these pairs must be kept secret. If a weak PUF only has one CRP, and it is revealed, then any device can emulate the PUF. For this reason, weak PUFs are well suited for use in key derivation processes. The PUF provides the randomness and secure storage, and the secret key (derived from the PUF’s response bits) is never revealed during operation. Once the key is recovered by the PUF (this typically requires error correction), any cryptographic process may follow. For example, the weak PUF output may be used as the key in a keyed-hash message authentication code challenge–response sequence. In addition, the output may be used as a secret key to encrypt/decrypt data on the device

Strong PUF Model

Strong PUFs differ from weak PUFs in that a strong PUF can support a large number of CRPs. As a result, a strong PUF can be authenticated directly without using any cryptographic hardware. The requirements for a strong PUF are:

- Large enough challenge–response space such than an adversary cannot enumerate all CRPs within a certain fixed time (ideally, exponential in the number of challenge bits);
- Responses stable to environment, multiple readings;
- an adversary given a polynomial-sized sample on adaptively chosen CRPs cannot predict the response to a new, randomly chosen challenge;
- not feasible to manufacture two PUFs with the same responses;
- The readout only reveals the response r =f(c) and no other data about the internal functionality of the PUF.

It should be noted that a weak PUF can provide authentication capabilities if the weak PUF is paired with crypto hardware supporting HMAC or similar authentication processes (note that HMAC and others support exponentially sized challenge–response spaces but their use requires 100 percentage response stability and, therefore, error correction logic). It should also be noted that the security models for weak and strong PUFs differ. The output of a weak PUF must be kept private, while a strong PUF’s responses do not have the same restriction.

The strong PUF has the additional requirement of readout access restriction [only r =f(c) is revealed] due to this difference in security models. In addition, to prevent total enumeration of the strong PUF, one must also consider the readout time of the PUF in conjunction with the number of CRPs. A faster PUF response allows for faster enumeration of all PUF CRPs. Since a weak PUF provides a secret key, the surrounding digital cryptographic hardware is responsible for limiting access to the weak PUF output. However, the strong PUF does not require the use of additional crypto hardware to provide authentication services, and therefore must itself prevent unauthorized access into its own internal structure.

Error correction

Both weak and strong PUFs rely on analog physical properties of the fabricated circuit to derive secret information. Naturally, these analog properties have noise and variability associated with them. Consider the example introduced in Section I that uses gate delay. This delay depends on temperature, supply voltage, and other environmental parameters. As these parameters vary, so does the “digital fingerprint” measured by the PUF. If the parameters vary too much, the digital key (for the weak PUF) or response (for the strong PUF) will change, and the crypto operation will fail. The first mechanism to mitigate such effects is to use differential design techniques to cancel out first-order environmental dependencies. Using the gate delay example, typical PUFs using this effect will not measure a single gate’s delay, but rather the difference between two identically designed, but distinct gates on a die. In this way, any environmental factor should affect each gate equally. Although differential design methodologies do improve reliability, noise is still a factor in PUF design. Even in optimal environmental conditions, noise will result in one or several of the output bits of the PUF being incorrect for any given challenge. Therefore, modern PUF designs employ multiple error-correction techniques to correct these bits, improving reliability. However, many of these error correcting techniques have been shown to leak bits of the secret key, since they require the computation and public storage of syndrome bits. As such, an excess number of PUF bits are generated and then downmixed to produce a full entropy key.

3.3.3 EXAMPLE STRONG PUF ARCHITECTURES

One of the first implementations of a strong PUF was constructed by Pappu et al. in 2001 [1]. The paper terms the device a “physical one-way function,” but the functionality is identical to that of a strong PUF. Pappu et al. describe a device with three primary components: 1) a laser directed along the Z-axis that can be moved in the XYplane and whose polarization can be modified; 2) a stationary scattering medium that sits along the path of the laser beam; and 3) an imaging device that records the output “speckle” pattern of laser light exiting the scattering medium. In this device, the input challenge is a laser XY location and polarization, and the response is the associated speckle pattern. The speckle pattern is strongly dependent on the input location/polarization because multiple scattering events occur inside the scattering medium. In the implementation by Pappu et al., the scattering medium consisted of a large number of randomly positioned 100-m silica spheres suspended in a hardened epoxy. Each sphere acts as a small lens, refracting individual rays of light as they move through the scattering block. The overall size of the scattering block was on the order of 1-mm thickness. Therefore, even a relatively simple optical path must encounter 10 spheres as it travels through the scattering block.All of these paths then are focused into an image on the detector. It is intuitively true that each of these paths will be very sensitive to input coordinates. Studies on speckle patterns produced by reflection/transmission by rough surfaces have found this to be true both experimentally and mathematically [2]. In addition, the speckle pattern is also sensitive to the internal structure of the scattering block. Therefore, it is difficult to fabricate two blocks with identical speckle patterns. Finally, due to the complex nature of the physical interactions, it is difficult to model the internal dynamics of the scattering medium. It is also difficult to use the output speckle to determine properties of the scattering block (such as the locations of the silica spheres).

Although the capabilities of the above optical PUF are significant, and they represented a significant step forward in the understanding and construction of PUFs, the practical applications are limited due to the macroscopic optical nature. This limitation stemmed from two properties. First, the actual unclonable object (the scattering block) was separate from the measurement apparatus (the imaging device). As a result, the trust gained from authenticating an optical PUF is more limited. In a practical use case, the objective of authenticating the PUF is typically to authenticate the associated processor to which it is connected. However, since the optical PUF is separated from the digital measurement circuitry, an optical PUF as described by Pappu et al. designed to authenticate processor A can easily be detached from processor A and connected to processor B. Processor B could then authenticate itself as processor A. It is more desirable for the digital measurement apparatus to be integrated in with the PUF such that the PUF is not separable from the device it is used to authenticate.

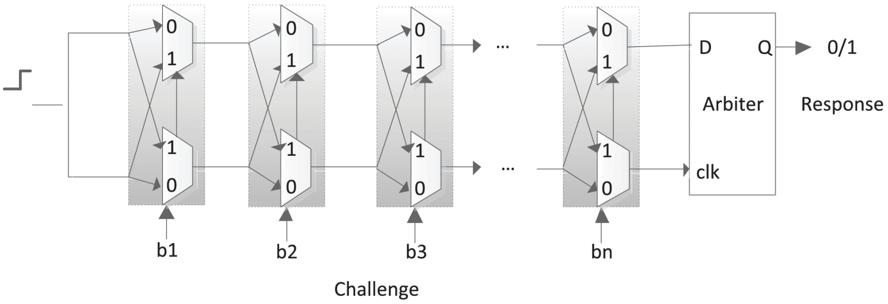


Figure 3.3: Structure of Arbiter PUF

Silicon implementations of strong PUFs were described in the paper by Gassend et al. beginning in 2002 using manufacturing variability in gate delay as the source of unclonable randomness [6]. In one implementation, a race condition is established in a symmetric circuit. An input edge is split to two multiplexers (muxes).Although the layout is identical (propagation time should be the same for each edge no matter what challenge bits are chosen), manufacturing variability in the gate delay of each mux will result in one edge arriving at the latch first, and the latch acts as the "arbiter." The output will, therefore, depend on the challenge bits. In Fig. 1, there are 128 challenge bits and one response bit. Of course, one typically operates multiple identical circuits in parallel to achieve 128 response bits. In this way, the arbiter PUF can be scaled to an almost arbitrary number of CRPs. The security of the arbiter PUF, like the optical PUF before it, is based on assumptions regarding manufacturing capabilities and ultimately metrology of the individual gate delays. Because the design is symmetric, the design does not contain any "secret" information. An adversarial manufacturer that has the PUF design cannot manufacture a duplicate PUF, because the behavior of the PUF is defined by the inherent variability in the manufacturing process. Even the original manufacturer of the PUF could not produce two identical PUFs, since this would require a significant improvement in manufacturing control.

The second security assumption is that the individual gate delays are difficult to measure directly. It assumes that an invasive attacker would have difficulty in extracting the individual delays even with physical access. This assumption is based on the hypothesis that an invasive attacker would destroy the gate delay properties using his/her measurement techniques. The last security assumption is that given a set of CRPs from an arbiter PUF, an adversary could not calculate the internal delays of the gates. For the architecture described above, this is actually not the case. Each delay is independent from all other delays, and the delays add linearly. As a result, one can use standard linear system analysis to intelligently gather data about the gate delays from the response bits. In fact, it can be shown that this system breaks after only a small number of challenges [17]. This problem can be resolved by several approaches proposed by Gassend et al. and described in Section IV-D. Finally, in both optical and arbiter PUF architectures, it should be noted that environmental factors play a significant role. For the optical PUF, calibration of the input location is a concern. In the case of the arbiter PUF, one can easily recognize that environmental variations such as temperature, supply voltage, aging, and even random noise will affect the delay of each edge through the arbiter PUF. In addition, if the delays are close enough, the latch's setup time will be violated, potentially resulting in an unpredictable output. As a result, the response bits may not be stable. In this case, error-correcting techniques are used to increase the stability of the PUF while maintaining its security. Although key generation has zero error tolerance, PUF authentication usually incorporates an allowable error threshold, thereby decreasing the stability requirement, and often obviating the need for error correction.

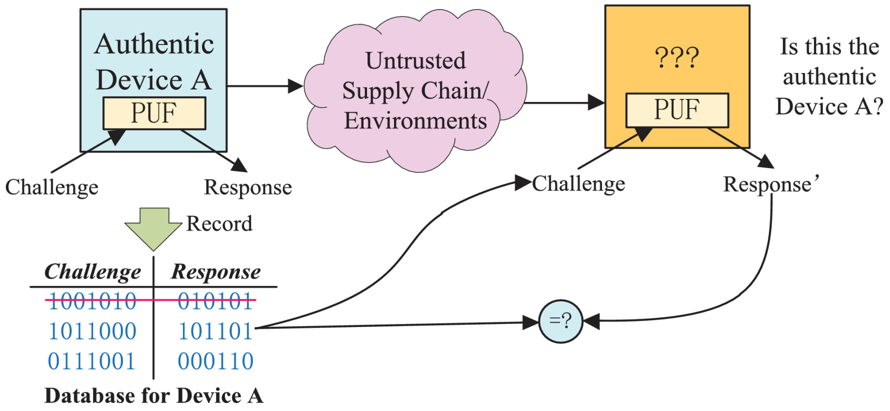


Figure 3.4: Traditional PUF based Authentication

As shown in Figure 3.4,a PUF is embedded into the device A, and CRPs are collected and stored in a secure database. As the PUF response is unique and unpredictable for each device, we can simply compare a regenerated PUF response with prestored response in database when using with same challenge for authentication. In order to protect against man-in-the-middle attacks, the used CRPs will be deleted from database. However, each PUF has a large number of CRPs, which will have a very high demand on server storage to store the CRPs for all devices. Therefore, we do not recommend using this common authentication method. In what follows, we will introduce a low storage overhead PUF-based authentication method.

First, we model the PUF with a machine leaning algorithm and then store the parameters instead of CRPs on the server, which incurs negligible storage overhead. If the delay of all blocks in a PUF path is known, it is easy to calculate the response after a given challenge. However, in practice, it is difficult to measure the delay of each block. Therefore, we use machine learning algorithms to simulate the delay of each block in the path of the PUF, i.e., the machine learning technique is used to build a software model for the PUF to simulate it's challenge-response behavior so as to predict the random response of the PUF. In this paper, Arbiter PUF is taken as an example and logistic regression is used to model the PUF.

The structure of Arbiter PUF is shown in Figure 3. Challenge C is generated by external control bits, and the output 0 or 1 is usually used as the response R. The function of an Arbiter PUF can be expressed with the linear delay model [2]. The total delay of the signal is the accumulated delay of each stage.

As discussed above, Arbiter PUF response can be expressed as a linear function of the challenge. Therefore, we can model the Arbiter PUF with the machine learning algorithms. However, attackers can also model the PUF with collected CRPs [6]. In order to resist the modeling attack, many obfuscation techniques [6] have been proposed to obfuscate the map relationship between challenges and responses and hence increase the difficulty of modeling attacks.

3.3.4 Authentication Protocol

As described previously, the strong PUF receives a challenge and generates a response. However, the requirements of a strong PUF state that an adversary provided with polynomial CRPs should not be able to predict the response to a new challenge. Although this is a desirable property, it also presents a usage problem. Since the PUF acts as a "black box," even the authentication server only has access to previously observed CRPs and, therefore, also cannot predict the response to a new challenge. Therefore, the protocol for using PUFs is significantly different than most public/private key cryptographic systems. Consider a server authenticating a client.

1. PUF is manufactured.
2. Server obtains access to PUF and generates a table of CRPs. These pairs are stored in internal secret storage
3. PUF is given to the client.
4. The client submits a request to the server to authenticate.
5. Server picks a known CRP and submits the challenge to the client.
6. The client runs the challenge on the PUF, returns the response to the server.
7. Server checks to see that the response is correct and marks the CRP as used.

3.4 Security Attacks

3.4.1 Guessing Attacks

The security of T2FA stems from the inability of attackers to guess the sound in the victim's environment at the time of the attack. It is difficult for attackers to guess the sound recorded by the victims phone since the recorded sound is dependent on the environment and recorded time. The experimental results prove that T2FA can discriminate the legitimate and fraudulent logins in noisy indoors, noisy outdoors and even in quiet indoor/outdoor areas. Therefore, T2FA is immune to guessing attacks.

3.4.2 Impersonation Attacks in Quiet Environments

In order to prevent the impersonation attack in quiet environments, we require the login user humming when the victim's environment is quiet (e.g., sleeping at night), which can improve the similarity significantly in the quiet environment. In Section IV, the experimental results show that the login success rate in quiet environment is 0%, but after humming in this environment, login success rate becomes 100%. Therefore, T2FA is immune to impersonation attacks in quiet environments.

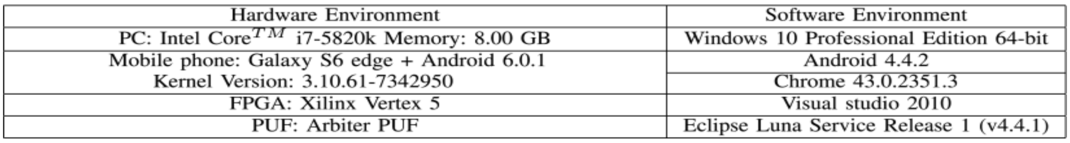
3.4.3 Modelling Attacks

Machine-learning based modeling attacks are one of well-known attacks for PUFs. They are exclusively applicable to strong PUFs such the Arbiter PUF. Strong PUFs has a publicly accessible CRP interface, which allows the simple collection of the large numbers of CRP that are required in this attack type during their learning phase . Since the Arbiter PUF response can be expressed as a linear function of the challenge, it is possible for attackers to collect a huge number of CRPs to model the PUF behavior. However, current obfuscation techniques can be used to obfuscate the map relationship between challenges and responses to resist modeling attacks.

Implementation

4.1 Test Environment

Implemented our proposed PUF and voiceprint-based transparent two-factor authentication system, where A 64 64 Arbiter PUF is implemented in a Xilinx Vertex 5 FPGA and consumes 184 LUTs. The hardware and software environment are shown in Figure 4.1.



Hardware Environment	Software Environment
PC: Intel Core^{TM} i7-5820k Memory: 8.00 GB	Windows 10 Professional Edition 64-bit
Mobile phone: Galaxy S6 edge + Android 6.0.1 Kernel Version: 3.10.61-7342950	Android 4.4.2
	Chrome 43.0.2351.3
FPGA: Xilinx Vertex 5	Visual studio 2010
PUF: Arbiter PUF	Eclipse Luna Service Release 1 (v4.4.1)

Figure 4.1: Test Environment

4.2 Audio Similarity Algorithm Test

4.2.1 Audio Similarity

We use Node.js to build a simple server that supports the test of audio similarity algorithm. During the test, every time the user logs in, the mobile phone and the computer will record 3s audio, and the audio files collected in different environments are processed by the algorithm to obtain the similarity values of the audio files.

We try to login in different scenarios to collect data such as noisy indoors (e.g., people playing music, video or chatting in the dorm room), noisy outdoors (e.g., eating in the canteen) and quiet indoor/outdoor (e.g., sleeping at night).

In the noisy indoor environment, we tested six different cases separately. We got the similarity values between the mobile phone and the computer in the six cases of close contact, 1m, 2m, 3m, more than 4m and the mobile phone in the pocket. The test results are given in Figure 5, which shows that close contact can obtain the best similarity, and similarity decreases as the distance increases.

In the noisy outdoor environment, we tested three different cases (close contact, more than 1m and the mobile phone in the pocket) separately to compute the similarity values. We can see from the test results in Figure 6 that in pocket and close contact have good similarity.

Similarly, in a quiet environment, the test results show that humming can improve the similarity significantly in the quiet environment.

4.2.2 Threshold for Similarity Algorithm

The ideal threshold should meet following requirements:

- In the indoor environment, when the distance between the mobile phone and the computer is less than 2m, the success rate of login is 100%; and the success rate of login is 0% when the distance is more than 4m.
- In the outdoor environment, when the distance between the mobile phone and the computer is less than 1m, the success rate of login is 100%; and the success rate of login is 0% when the distance is more than 1m.
- Regardless of indoor and outdoor, the quiet environment login success rate is 0%. After humming in this environment, login success rate becomes 100%.

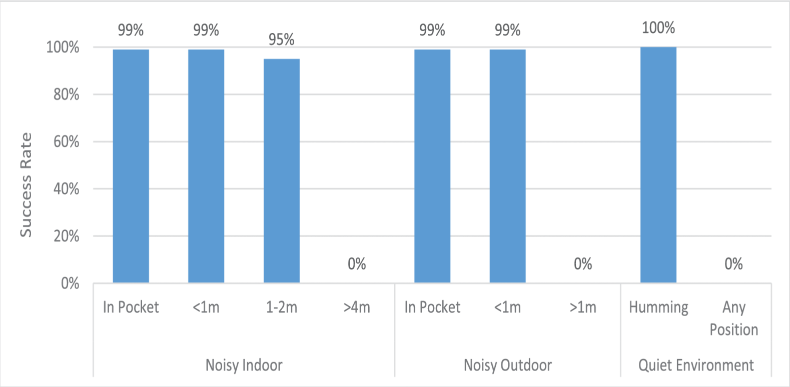


Figure 4.2: Success Rate in Noisy indoor,Noisy Outdoor and Quiet Outdoor/indoor areas

The test results satisfies the following conditions:

- In the noisy indoor environment, when the distance between the phone and the computer is less than 1m or the mobile phone is in the pocket, the success rate is 99%; when the distance is greater than 1m and less than 2m, the success rate is 95%; when the distance is greater than 4m, the success rate is 0%.
- In the noisy outdoor environment, when the distance between the phone and the computer is less than 1m or the mobile phone is in the pocket, the success rate of login is 99%; when the distance is greater than 1m, the success rate is 0%.
- Regardless of indoor or outdoor, the success rate for quiet environment is 0%, and after humming in this environment, the success rate becomes 100%.

4.2.3 Modelling Arbiter PUF

As shown in figure, for a 64 64 arbiter PUF, we can achieve 95% prediction accuracy with only 650 CRPs and the training time is less than 1s, and achieve 99% prediction accuracy

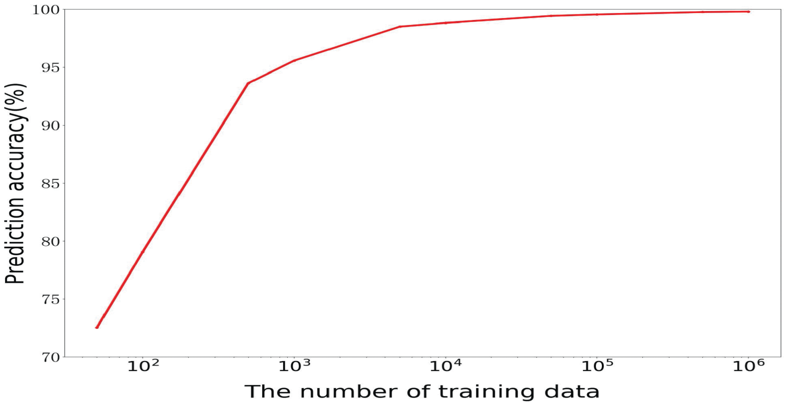


Figure 4.3: LR on a 64 * 64 Arbiter PUF

with less than 3000 CRPs in 1s, and 99.9% prediction accuracy with about 20000 CRPs in about 2s. Therefore, in our proposed T2FA, storing the model parameters of PUFs on the server for device authentication not only reduces the storage overhead but also improves the authentication efficiency.

Conclusion

This paper proposes the concept of transparent two-factor authentication and then proposes the first PUF and voiceprint-based transparent two-factor mechanism. The first authentication factor is still the traditional username and password.The second authentication factor is the mobile phone, which includes two novel authentication ways: 1) PUFs are used to authenticate the mobile phone; 2) the same environment verification between the computer and the mobile phone. When the user logs in, the browser sends a login request to activate the mobile phone. Both of them use their respective microphones to record the environment noise, and adjust the time-stamp to synchronize the time. The mobile phone compares the similarity of the two audios to determine whether the browser is in the same environment as the mobile phone and then determines whether the login is valid. The second factor authentication is completely transparent to the user, avoiding the tedious interaction between the user and the device. Therefore, our proposed T2FA exhibits the anti- fraud ability with good application prospect.The effectiveness of our proposed T2FA is further proved with detailed experiments.

REFERENCES

[1] J. Zhang, G. Qu, Y. Lyu, and Q. Zhou, “A survey on silicon PUFs and recent advances in ring oscillator PUFs,” J. Comput. Sci. Technol., vol. 29, no. 4, pp. 664–678, Jul. 2014.

[2] C. Herder, M.-D. Yu, F. Koushanfar, and S. Devadas, “Physical unclonable functions and applications: A tutorial,” Proc. IEEE, vol. 102, no. 8, pp. 1126–1141, Aug. 2014.

[3] J. Zhang, “A practical logic obfuscation technique for hardware security,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 24, no. 3, pp. 1193–1197, Mar. 2016.

[4] C. M. Bishop, “Pattern recognition and machine learning,” J. Electron. Imag., vol. 16, no. 4, p. 49901, Jan. 2007.

[5] M. Majzoobi, F. Koushanfar, and M. Potkonjak, “Lightweight secure PUFs,” in Proc. IEEE/ACM Int. Conf. Comput.-Aided Design, Nov. 2008, pp. 670–673.

[6] G. E. Suh and S. Devadas, “Physical unclonable functions for device authentication and secret key generation,” in Proc. 44th ACM/IEEE Design Automat. Conf., Jun. 2007, pp. 9–14.

[7] P. Qiu et al., “Physical unclonable functions-based linear encryption against code reuse attacks,” in Proc. 53rd Annu. Design Automat. Conf., Jun. 2016, pp. 1–6.

[8] Verayo Technology. (2018). [Online]. Available: http://verayo.com/ tech.php