Low-overhead, Robustic Slender PUF Authentication for Active Attacks based on Pattern Matching

P. G. Siva Sharma Karthick
Assistant Professor computer science & engineering Nadar Saraswathi College of Engineering and Technology

B. Preethi
Final year, computer science and engineering
Nadar Saraswathi College of Engineering and Technology

M. Gokula Priya
Final year, computer science and engineering
Nadar Saraswathi College of Engineering and Technology

***Abstract*---This paper proposes a low-overhead, Robustic Slender Physical Unclonable Function (PUF) authentication and key exchange protocols for active attacks and it is based on pattern matching. This method is well suited for ultra-low power and embedded devices. The protocols are executed between the prover and verifier. The prover will receive a challenge from verifier for authentication and in return proversend a random subsets of the PUF response strings to the verifier. A key is generated at the prover side when the responses are sent and the verifier will match the substrings by using the pattern matching. Next the key generation is occurred at the verifier side. The authentication is provided only when both the keys are unique. By using this approach, the system performance will not be degraded; moreover active attacks like PUF modeling attack, man-in-the-middle attack and substring replay attacks were prevented.**

Keywords*---*Slender Physical Unclonable Function,Prover and Verifier

I.INTRODUCTION

In the existing system, Physical Unclonable Functions (PUFs) have been used to provide a desired level of security with low implementation overhead. The implementation is done using the hardware device, FPGA (Field Programmable Gate Arrays). The arbiter-based PUF on FPGA were designed to have 64 input challenges. To achieve a higher throughput, multiple parallel PUFs were implemented on same FPGA. When verifier sends the challenges to the prover, prover then sends the responses to verifier by obfusticating the original content by adding the substrings to it. The verifier then matches the substring with the PUF compact model, and generates a key only when substrings are matched. Then the prover will be authenticated.The proposed system does not contain any hardware implementation. Slender PUF protocol is been used for secure and pattern matching. All the characteristics and properties of PUF is been stored the database. The PUF codes will be unique among other systems. The verifier sends the input to the prover as the challenges. The prover in turn sends the responses by hiding the original content by adding the substrings along with response bits. When the response is sent, a key is been generated at the prover side. The verifier obtains the responses and matches the substrings, with the PUF compact model that is stored in the database, by the concept of pattern matching. If the pattern is matched, only then the key is generated for the verifier side. Unlike existing system, only if both the keys are unique at the prover side and verifier side, the authentication is provided for the prover. Our proposed system completely prevents the machine learning attacks. Since there is no additional hardware components used. Active attacks such as man-in-the-middle attack, substring-replay attack and PUF modeling attack is been prevented. The system performance is good, as it doesn't make the system to degrade and do not allow to slow down.

II.LITERATURE SURVEY

PUFs have been subjected to modeling attacks. The basis for PUF modeling attacks is by collecting a set of CRPs and then building them into a numerical or algorithmic model. Previous work on PUF modeling (reverse engineering) and various machine learning technique to attack both implementation and simulations of a number of different PUF families, including linear arbiter PUFs and feed-forward arbiter PUFs such as extracting the secret keys from the integrated circuits. More comprehensive analysis and description of PUF security requirements to protect against modeling attacks on simulated and silicon data and reconfigurable PUFs. In recent years, these have been an ongoing effort to model and protect PUFs against side channel attacks such as power analysis and fault injection. The PUF used the analog difference between the delays of two parallel paths that are equal in design, but the physical device imperfection makes the delays different.

III. EXISTING SYSTEM

Physical Unclonable Functions (PUFs) were used to provide desired level of security. The implementation was

done by the hardware, most commonly known as FPGA (field programmable gate arrays). To achieve a higher throughput, multiple parallel PUFs were implemented on the same FPGA. Verifier at first sends the challenges to the prover. The prover then sends the responses to the verifier along with the substrings added to it. In turn, verifier then matches the substrings in the response bits, with PUF compact model. Since each compact model uses a different substring pattern. If the substring matches the PUF compact model, at the verifier side, only then a key is generated and sent to prover. By this the prover is authenticated.

A. Disadvantages

The existing system used hardware to measure the PUFs compact model. And hence it leads to machine learning attacks or reverse engineering attacks. Hardware failure is possible even though it provides a secure and low-overhead authentication.

IV. PROPOSED SYSTEM

Our proposed system does not consist of any additional hardware features to implement. Instead of PUF, Slender PUF is been used for secure and low-overhead authentication. The protocol also uses the pattern matching concepts. The details of the PUF compact model is been stored in the database at the verifier side. The PUF codes will be unique among other systems. The verifier sends the inputs to the prover as the challenges. The prover sends the responses to the verifier along with the substrings added to the response bits. A key will be generated at the prover side, when the response is sent. The verifier then uses the pattern matching concepts, and matches the substrings in the response bits, with the details of PUF compact model in the database. Only if the pattern is matched, the key is generated at the verifier side. When both keys generated at the prover and verifier side are unique, only then prover will be authenticated.

A. Advantages

The proposed system completely prevents the machine learning attacks. Since there is no additional hardware used. It also secures the system from active attacks such as man-in-the-middle attack, substring-replay attack and PUF modeling attack. The system performance will not degrade and provides a security to a greater extent.

V. ARCHITECTURE

The architecture of the proposed system is shown in the below given figure. It shows that verifier is sending the challenges to the prover. Prover sends the responses to the verifier along with substrings. A key is generated by the key generator ‘p’. The substring matching is done by the concepts of pattern matching, by using the details of the compact model stored in the database. Another key is generated at verifier side by key generator ‘v’. The comparison of generated keys (v+p) must be unique, and only then the authentication is provided to prover. The analysis of attacks has been performed. And the system performance is measured abruptly and a high security is been provided to the overall process model. The further explanation is been given in the modules.

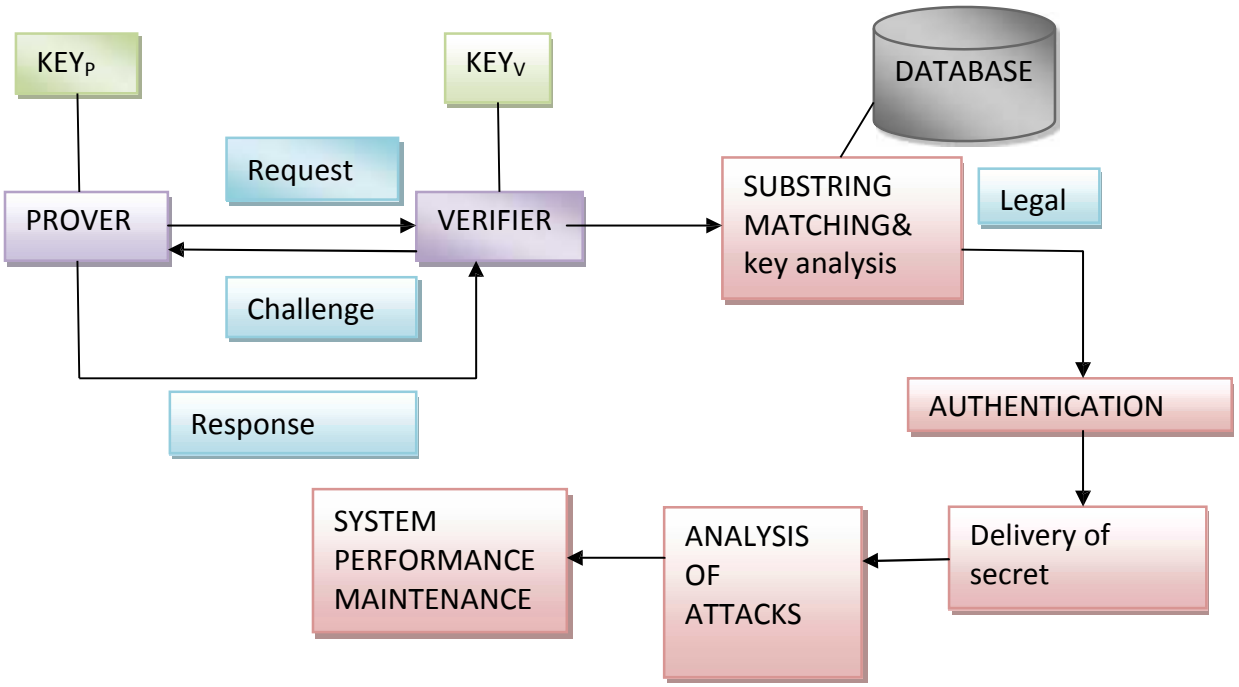


Fig 1: Block diagram for prover authentication and message transfer

VI. ALGORITHMS

A. Identity-Based Encryption

The identity based encryption is a public key encryption method. It will have the unique information of prover. Identity-Based Encryption will use the arbitrary string as a public key and it protect the data. The ID based encryption is as follows,

SETUP- initializes the key scheme.

ENCRYPT- encrypt the user's message.

KEY GENERATION- generates the private key for the given user.

DECRYPT- decrypts the message.

A.a Advantages

The advantage is that the identity based encryption algorithm generates a key based on the prover's personal identity such as name, e-mail id, mobile number, etc.

B. Attribute-Based Encryption

The Attribute-Based Encryption is a type of public key encryption, where the secret keys and the cipher text are dependent upon the attributes such as system properties, system code, etc. This is the decryption is possible only of the set of attributes of the prover key match the attributes of cipher text. The attribute based encryption is as follows,

SETUP- it takes no input other than the implicit security parameters.

KEY GENERATION- the key generation algorithm takes the input master key and a set of attributes and describes the key. It output will be a private key.

ENCRYPT- the encryption algorithm takes the input as public parameters, a message and an access structure upon the attributes. The algorithm will encrypt the message and produce a cipher text such that only the verifier will have the access to decrypt the message.

DECRYPT- the decryption algorithm will takes the input the public parameters and cipher text.

C. Pattern matching

Our proposed algorithm is multiple skip multiple pattern matching algorithm which is based on Boyer - Moore ideas. It scans the input file to find all occurrences of a pattern within this file, based on skip techniques.
The proposed algorithm MSMPMA assumes that there is input text file that has size and there is a pattern with size so the algorithm proceeds as follows:

1. Input text along with size and pattern along with size.
2. Output starting index of all substring occurrences of the text that is equal to pattern and output if no such substring exists.
3. Initialization is done for starting procedure.
4. Check index, if index less than or equal to the difference between the size of the text and size of the pattern, then proceed the further steps, otherwise end the procedure.
5. Set index as j of pattern to 1.
6. Check j. If j I is less than or equal to the size of the pattern go to next step, otherwise return.
7. Compare the text and pattern, if they are equal skip the step, otherwise go back to the previous step.
8. Increment number of occurrences
9. Return number of occurrences.

The comparison table of MSMPMA algorithm with other classic algorithms is shown in the below table:

Table 1: Comparison with other algorithms

Algorithm	Number of occurrences	Number of comparisons	Comparisons per character
MSMPMA	11	1298	1.268
Brute-Force	11	1318	1.287
Trie-matching	11	1321	1.290
Native String Search Algorithm	11	1310	1.279

VII. MODULES

In this section, we perform the implementation on the basis of software by setting the PUF parameters, key exchange protocols and CRPs, prover authentication, pattern matching, analysis of attacks, system performance and protection and security.

A. SETTING PUF PARAMETERS

In existing, PUF parameters were measured through hardware FPGA. Unlike existing system, our proposed system uses only the software implementation. The PUF parameters are set by retrieving the information from the PUF compact model and storing it in the database at the verifier side.

B. KEY-EXCHANGE PROTOCOLS AND CRPs

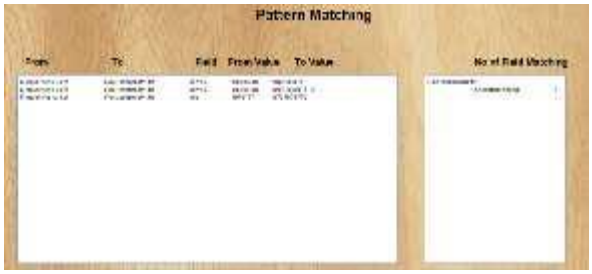
The PUF must be able to produce exponentially many challenge response pairs. According to existing system, the CRPs were generated by two methods. One, by entering the oscillators to have configurable delay paths similar to arbiter PUF. Second, in programmable logic such as FPGAs, a challenge can determine the oscillator configuration such as the number of invertors and which look-up tables and wires to be used. In proposed system, all the challenge-response pairs are generated through software. The verifier sends the challenges to the prover. And prover sends the response along with sub-strings. A will be generated at the prover side, after sending the response bit.

C. PROVER AUTHENTICATION

The verifier receives the sub-strings padded response bits and performs pattern matching, by using the details of the Compact model in the database, at the verifier side. If the substrings are match able only then another key is generated at the verifier end. The response strings will be in the encrypted format.

D. PATTERN MATCHING

Both the keys generated at the end of the prover and verifier side must be unique, only then the prover will be authenticated. The pattern matching is done after decrypting the encrypted message received from the prover. Verifier then matches the substring using the PUF compact model in the database and then generates a key. The key at both verifier and prover side must be unique. Only then the prover will be authenticated.



E. ANALYSIS OF ATTACKS

In this model, we quantify the resistance of the proposed protocols against different attacks by a malicious party (prover and verifier). Due to same system properties of prover attacks may threat their authentication. Our algorithm in the proposed will never lead to any attack and we have also overcome few as mentioned below:

E.a. Puf-modeling attacks

In the existing process, a trusted IP owner with physical access to the device (e.g. the original manufacturer) can build a compact model by measuring the PUF direct responses. Such compact model by measuring the PUF direct responses. Such compact models can be treated as a secret which can be used by a trusted verifier to authenticate the prover's PUF. Unfortunately, third party observers may also be able to model the PUF based on finite number of CRPs exchange on the communication channel. The hardware that used was capable of leaking challenge-response pairs of the PUF compact model. But in the proposed, we declare the parameters and corresponding CRPs through software only. And hence no PUF modeling attack will be happened.

E.b. Substring-replay attacks

A dishonest prover may record the response substrings from the honest prover, which is sent to honest verifier. This recording may be performed by eavesdropping on the communication channel between the legitimate prover and verifier. The recorded response substrings are used by dishonest prover, by repeatedly contacting the legitimate verifier for authentication. Our proposed protocols do not allow any access for the third-party observers in the communication channels, since no excessive hardware components are used unlike existing system.

E.c. Man-in-the-middle attacks

Asymmetric cryptographic algorithms, such as RSA and Diffie-Hellman, are the algorithms that are traditionally used for secret key exchange. These algorithms are susceptible to man-in-the-middle attacks. Therefore, needed a certificate authority for secure implementation. However, our proposed key exchange algorithm is not susceptible to man-in-the-middle attack and no certificate authority is required for implementation.

F.SYSTEM PERFORMANCE

The system performance is categorized by the amount of useful work accomplished by a computer system compared to time and resources used. The Slender PUF used in proposed, takes less time consumption. The processing speed will not be degraded. Higher throughput that is the rate of production at processing period.

G. PROTECTION AND SECURITY

The protection is a feature that regularly creates and saves information and protects against some threats and vulnerabilities. Our work has been protected the system from active attacks such as PUF modeling attack, substring-replay attack and man-in-the-middle attack. Security is given to prevent and monitor unauthorized access, misuse, modification or denial of a computer network and network accessible resources. Here the authentication is provided in a more secure manner, since as two keys are generated for the authentication purpose both at the prover side, as well as at verifier side.

VIII. CONCLUSION

We have presented a secure, low-overhead, robustic Slender PUF authentication for active attacks based on pattern matching. In the authentication process, the prover reveals only the responses to verifier and a key is generated. The verifier who has the details of the PUF compact model in the database has the full access to retrieve and match the substrings in the response bits. The pattern matching concepts leads to generate another key at the verifier end. Both keys will be unique. The authentication is successful if both keys are unique at verifier and prover side. Further, In future this work will include the pattern matching concept using image or steganography in the prover side unlike text strings.

IX. ACKNOWLEDGEMENT

We express our sincere thanks to our honorable secretary Mr. B.Ramachandran, respected principal Dr.K.P.Satheyamoorthy, and Head of the Department of CSE Mr.UdhayaKumar who guided and encouraged us finish our project.

11. REFERENCES

[1] M. Majzoobi, M. Rostami, F. Koushanfar, D. S. Wallach, and S. Devadas,"Robust and Reverse-Engineering ResilientPUF Authentication and Key-Exchangeby Substring Matching" , IEEE transactions in emerging topics on computing.

[2] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, ``Silicon physical random functions," in Proc. Comput. Commun. Security Conf., 2002, pp. 148 160.

[3] U. Ruhrmair, S. Devadas, and F. Koushanfar, Security Based on PhysicalUnclonability and Disorder. New York, NY, USA: Springer-Verlag, 2011.

[4] M. Rostami, J. B. Wendt, M. Potkonjak, and F. Koushanfar. (2014, Mar.). ``Quo vadis, PUF." in Design, Automation & Test in Europe, to be published.

[5] Z. Paral and S. Devadas, ``Reliable and efcient PUF-based key generation using pattern matching," in Proc. Int. Symp. Hardware-Oriented SecurityTrust, 2011, pp. 128 133.

[6] M. Majzoobi, M. Rostami, F. Koushanfar, D. S. Wallach, and S. Devadas, ``Slender PUF protocol: A lightweight, robust, and secure authentication by substring matching," in Proc. IEEE Symp. Security Privacy Workshops, May 2012, pp. 33 44.

[7] F. Koushanfar, Hardware Metering: A Survey. New York, NY, USA: Springer-Verlag, 2011.

[8] B. Gassend, D. Clarke, M. Van Dijk, and S. Devadas, ``Delay-based circuit authentication and applications," in Proc. ACM Symp. Appl. Comput., 2003, pp. 294 301.

[9] M. Majzoobi, F. Koushanfar, and M. Potkonjak, ``Testing techniques for hardware security," inProc. Int. Test Conf., 2008, pp. 1 10.

[10] Suh and S. Devadas, ``Physical unclonable functions for device authen-tication and secret key generation," in Proc. 44th ACM/IEEE Des. Autom.Conf., Jun. 2007, pp. 9 14.

[11] M. Majzoobi, F. Koushanfar, and M. Potkonjak, ``Lightweight secure PUF," in Proc. Int. Conf. Comput. Aided Des., 2008, pp. 670 673.

[12] M. Majzoobi, F. Koushanfar, and M. Potkonjak, ``Techniques for design and implementation of secure recon gurable PUFs," ACM TRETS, vol. 2, no. 1, pp. 1 33, 2009.

[13] A.Mahmoud, U. Ruhrmair, M. Majzoobi, and F. Koushanfar. (2013). Combined Modeling and Side Channel Attacks on Strong PUFs [Online].Available: https://eprint.iacr.org/2013/632

[14] M.-D. M. Yu and S. Devadas, ``Secure and robust error correction for physical unclonable functions," IEEE Des. Test Comput., vol. 27, no. 1, 48 65, Jan./Feb. 2010.

[15] M. Majzoobi, G. Ghiaasi, F. Koushanfar, and S. Nassif, ``Ultra-low power current-based PUF," in Proc. IEEE Int. Symp. Circuits Syst., 2011, 2071 2074.

[16] M. Majzoobi, F. Koushanfar, and S. Devadas, ``FPGA-based true ran-dom number generation using circuit metastability with adaptive feedback control," in Proc. Int. Cryptograph. Hardware Embedded Syst., 2011, pp. 17 32.

[17] M. Majzoobi, F. Koushanfar, and S. Devadas, ``FPGA PUF using pro-grammable delay lines," in Proc. IEEE Int. Workshop Inf. Forensics Secu-rity, Dec. 2010, pp. 1 6.

[18] V. Boyko, P. MacKenzie, and S. Patel, ``Provably secure password authenticated key exchange using Dif_e_Hellman," in *Advances in Cryp- tology*. New York, NY, USA: Springer-Verlag, 2000, pp. 156_171.

[19] M. Bellare, D. Pointcheval, and P. Rogaway, `Authenticated key exchange secure against dictionary attacks," in *Proc. 19th Int. Conf. Theory Appl. Cryptograph. Tech.*, 2000, pp. 139_155.

[20] N. Beckmann and M. Potkonjak, ``Hardware-based public-key cryptography with public physically unclonable functions," in *Information Hiding*. New York, NY, USA: Springer-Verlag, 2009, pp. 206_220.