Verilog Implementation of a Secure System for Smart Card Transactions with Advanced Encryption Standard and Secure Hash Algorithm

Iqbalur Rahman Rokon[1], Utshash Das[2], Subhobroto Choudhury[3]
Department of ECE
North South University
Dhaka, Bangladesh

***Abstract*--Smart cards can be used for many different services, ranging from identification and authentication to data storage. Using a single smart card to avail these services and more requires using a multiple service system. This multiple service system opens doorways to user's privacy being breached.**

A design proposed by Hong Mei and Guo Hui in 2009 addressed various security issues related to such a system. This paper presents the Verilog HDL implementation of the proposed design with a few modifications. The design involved two phases where the simulation was done using ModelSim-Altera 10.1c Starter Edition. The design was then synthesized and the RTL schematic diagrams obtained using Xilinx ISE 9.2i.

Keywords—Smart Card; Privacy; Security; Simulation; Synthesis

I. INTRODUCTION

The evolution of Smart Cards from simple phone cards to multifaceted high technology security solutions supporting a large array of applications in the last few decades has been astounding. Security features of previous smartcards were limited to a mechanism preventing the chip on the card to be filled up again whereas today's smart cards are re-usable, hold large quantities of data, speed transaction times, identify the cardholder, and even provide loyalty benefits [1].

By using a multiple-service smart card, a user can access many services from different service providers using a single card instead of having to use separate access devices for each service, hence saving the users from the hassle of carrying many cards. In such a system, each user normally shows the same identity to different service systems. According to [2] this makes the user vulnerable since the user's access behavior to different services can be easily traced, linked and abused by adversaries. Using a single password increases the vulnerability of the system because with a one-password scheme, there is a high risk of passwords being tampered since they are exposed to many different parties. Cryptosystems are hence vital to be incorporated into a smart card to shield users against attackers.

Thus, this design was entirely based on a system that nearly leaves no scope for personal information leaks. A sophisticated system model that addresses the aforementioned problems with the usage of secure cryptosystems, Advanced Encryption Standard (AES) and Secure Hash Algorithm (SHA), is discussed in this paper. The high security smart card will use a single password for authentication for all purposes. The one password scheme allows easy access for the user to his various modes of operations.

II. LITERATURE REVIEW

Many approaches to smart card protection have been proposed which primarily involves identity validation and communication protection.

In 1981, Lamport [3] proposed a scheme for remote password authentication with insecure communication. In his scheme, the user passwords are stored in the hashed format and are also hashed during transmission to prevent password eavesdropping. To realize such a design, the approach uses a sequence of passwords, each of which is formed by repeatedly hashing a given secret value. For each round of authentication, a different password from the sequence is used. The approach displays two common weaknesses: the remote system needs to maintain a password table which causes memory overheads for large numbers of users. Also, it risks attacks if the password table is somehow administered by intruders. Juang [4] in 2004 proposed a multi-server password authentication protocol. In this approach, no password verification table was needed rather user chose their own passwords; users and servers can be mutually authenticated. In October 2009 Smart Card Alliance [5] published a paper describing in details, the security measures which can be taken for a smart card.

In 2013 Rumaisah Munir and Saad Bin Khalid [6] implemented a smart card model both on a hardware and software level in which a user can open an account, withdraw and deposit cash, verify their identities and cancel transactions.

In 2009 Hong and Guo [2] proposed a sophisticated design based on previous contributions that counter most security issues.

III. CRYPTOGRAPHIC ALGORITHMS

A. *Secure Hash Algorithm*

Secure Hash Algorithm [7] is a keyless cryptographic hash function. In this design SHA-256 has been used which takes in an arbitrary length input and after performing a series of manipulations gives out an output of 256 bits. In SHA-256, a message is processed by blocks of 512 = 16 × 32 bits, each block requiring 64 rounds.

B. Advanced Encryption Standard

Advanced Encryption Standard or AES [8] is a symmetric key cipher, meaning it uses only one key (secret) in order to encrypt or decrypt data. It is a block cipher in which the plaintext bits are taken as a whole block to be encrypted on as opposed to a stream cipher which acts on strings of data, one bit at a time. The key size varies for different versions of the encryption standard. In this design AES-128 has been used, where the key size and block size are specifically 128 bits long and hence the name.

IV. METHODOLOGY

Operational Flow

The operational flow for the secure smart card system is shown in Fig. 1.

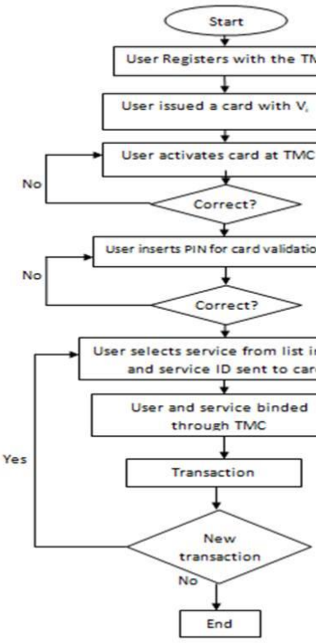
Start
User Registers with the TMC
User issued a card with V$_i$
User activates card at TMC
No
Correct?
User inserts PIN for card validation
No
Correct?
User selects service from list in TMC and service ID sent to card
User and service binded through TMC
Yes
Transaction
New transaction
No
End

Fig. 1. Operational flow for the system

At first the user needs to register with a Trusted Management Center (TMC) by providing some personal information to the TMC namely his national identity card (Ussn) number and date of birth (Udob). Following that, the user is issued a card with a secret parameter (Vi), a personal identification number (PIN) and a symmetric key (kcard) for latter communications stored inside of it.

Before the smart card is used for the first time, it needs to be activated using the stored parameter and a password (PWi) provided by the user. If the smart card is activated, the user can then proceed with service transaction.

To perform service transaction the user is first validated against the PIN stored in the smart card. The user can then select a service provider from the list of service providers registered to the TMC and the service provider's unique identity is sent to the smart card. The user and the service provider then bind through the TMC.

Finally, the user can select the desired transaction and it is carried out. If the user wants to perform another transaction, the steps including and following service selection are repeated. Otherwise, the operation terminates.

V. BLOCK DIAGRAMS OF THE SYSTEM

A. The Top Block of the System

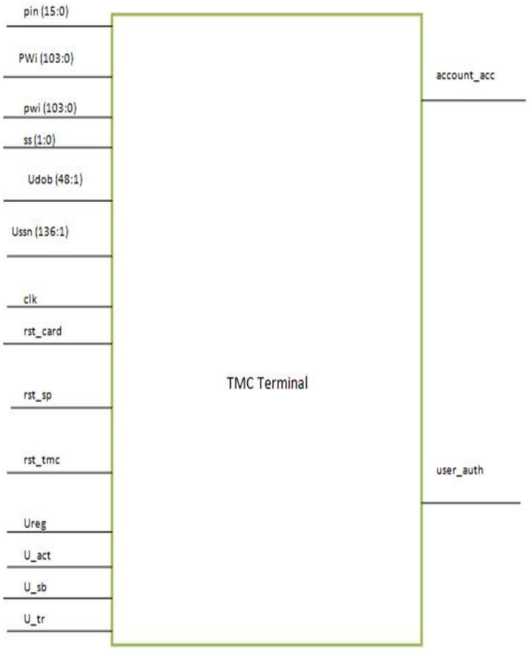

Fig. 2. The top block of the system

The TMC terminal is the top level block consisting of the lower level blocks: the Smart Card, TMC and the Service Provider. This block shows all the primary input and output ports of the system.

B. The Internal Blocks of the TMC terminal

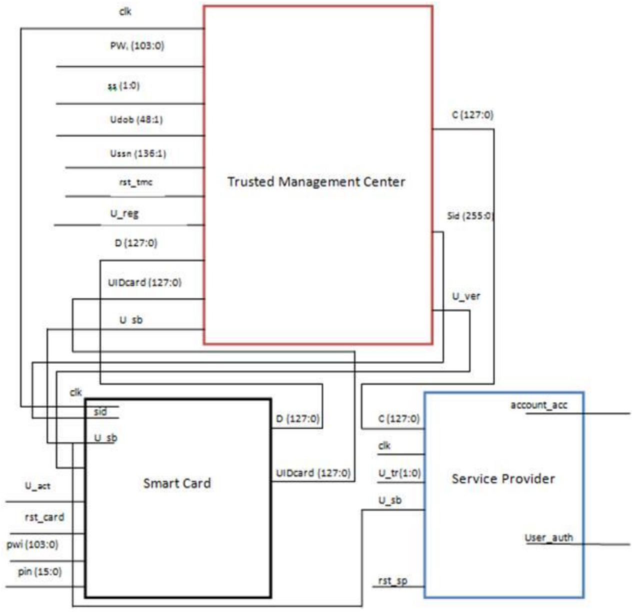

Fig. 3. The internal blocks of the TMC Terminal

These blocks are inside the TMC terminal. Fig. 3 shows the interconnections between the TMC, Smart Card and the Service Providers.

VI. DESIGN DETAILS

Functions used in this section are described as follows
hash ( ) : SHA-256 hash function
E_k ( ): AES-128 encryption algorithm, where k is the cipher key
D_k ( ): AES-128 decryption algorithm, where k is the cipher key

A. User Registration

To register, a user provides the TMC with his/her personal information, namely national ID card number (Ussn) and date of birth (Udob). Both these information are concatenated to form a unique number which is again concatenated with a secret key only known to the TMC, k. The result is hashed using SHA-256 and a selected one hundred twenty eight bits of the resultant 256-bit hashed value becomes the unique user registration identity (UID) for the user. The UID is stored in the TMC.

$$UID = [255:128]\ hash((Ussn||Udob)||k) \qquad (1)$$

None of the user's personal information are stored anywhere in the system. Moreover, the hash function used ensures that, with the knowledge of user registration identity (UID), it is computationally infeasible to retrieve the user's personal information.

Remark 1: Selecting 128-bits of the resultant hash value as the UID in contrast to the whole 256-bits as used in the design proposed by [2] further enhances security.

The user also provides a password, PWi, to the TMC. A bit-wise exclusive OR (XOR) operation is used on the password and the user registration identity (UID) to create a parameter Vi.

$$Vi = UID \oplus PWi \qquad (2)$$

The TMC then issues the user a Smart card loaded with the parameter Vi, a unique personal identification number (PIN) and a symmetric key (kcard) for later communication.
The user's personal information can only be revealed with UID on the TMC. In order to maintain full privacy, the user's password and user registration ID are not stored directly in the smart card. Rather the smart card contains an indirect parameter derived from those information. Hence if the card is lost or stolen, for an adversary to obtain the UID or PWi by exploring information on the card is nearly impossible.

B. Service Registration

A service provider needs its services registered with the TMC as well before it can provide them to the smart card user. When a service provider, registers its services with the TMC, the TMC assigns a unique service identity (SID) and a symmetric key (kj) for latter communications which are stored on both sides of the system.
The service identities (SIDs) are available to any user in the system. However, both the TMC and service providers should keep their communication symmetric-key confidential.

C. Card Activation

Before the Smart Card is used for the first time, it must be activated. To activate a card, the user inserts his/her card into a secure terminal at the TMC, and then inputs his/her password on prompt. The card can now compute the user registration ID (UIDcard) from the given password and the parameter Vi stored in the smart card.

$$UIDcard = Vi \oplus PWi \qquad (3)$$

Once UIDcard has been computed by the card, it is sent to the TMC for verification. The TMC checks whether the UIDcard matches with any of the UIDs stored in it. If there is a match, a user-verification enable signal becomes high, making the card active.

D. User Service Binding and Transaction

This procedure allows a user to receive services from the service providers. A user must bind with a service provider each time it wants to perform some transaction. The user-service binding and transaction phase is described below:

- At a trusted card reader terminal provided by the TMC, the user attaches his/her smart card and provides the PIN. After the card is validated, the user is asked to insert his/her password. With the password provided by the user, the smart card computes the user registration identity (UID) based on Formula 3.

- The user then selects the service it wants to register with. When a service provider is selected, its service identity, SID is transferred to the smart card. A bit-wise exclusive OR (XOR) operation is applied on the UID and SID and the result is hashed. A selected one hundred and twenty eight bits of the resultant 256-bit hash value becomes the user-service binding identity, USID.

$$USID = [255:128]\ hash\ (UID \oplus SID) \qquad (4)$$

Remark 2: Again selecting 128-bits of the resultant hash value as the USID in contrast to the whole 256-bits as used in the design proposed by [2] further enhances security for the User-Service binding phase.

- The USID computed is then sent to the TMC with AES encryption. The symmetric key shared between the smart card and the TMC is used to encrypt the USID.

$$D = E_{kcard}(USID) \qquad (5)$$

- Upon receiving the AES encrypted USID, which is D, the TMC decrypts it using the symmetric key between the Smart Card and the TMC to retrieve the USID. The TMC then encrypts the USID with the symmetric key (kj) that it shares with the selected service provider and sends it to the service provider.

$$USID = D_{kcard}(D) \quad (6)$$

$$C = E_{kj}(USID) \quad (7)$$

- On receiving the AES encrypted USID, C, from the TMC, the service provider decrypts it using the symmetric key shared by itself and the TMC. This signifies the completion of the user-service binding. The USID computed can be used as the symmetric key between the smart card and the service provider for latter communications.

$$USID = D_{kj}(C) \quad (8)$$

- Once user-service binding has been completed, the user selects the transaction that it wants to perform with the service provider. In our design, we provide two such transactions, namely user authentication and account access.

Remark 3: In the design proposed by [2], the User-Service binding identity, USID is stored when the user asks for a service. The USID, in our design, is only present for the duration of the service transaction. Each time a service is to be requested by the user, the USID needs to be dynamically calculated.

The hash function used in the calculation of USID again ensures that with the knowledge of USID, it is computationally infeasible to find the user registration identity (UID). Neither the user's real identities nor their activities on other services are exposed to the Service Provider.

VII. EXPERIMENTAL RESULTS AND DISCUSSION

A. Simualtion Results

The simulations were done in phases. The first phase is the User Registration phase as shown by Fig. 4.

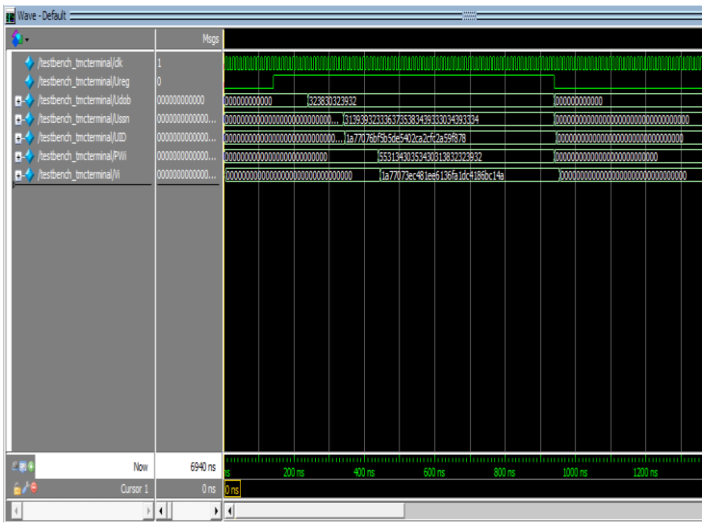
Wave - Default
Msgs
/testbench_tmcterminal/clk
/testbench_tmcterminal/Ureg
/testbench_tmcterminal/Udob
/testbench_tmcterminal/Ussn
/testbench_tmcterminal/UID
/testbench_tmcterminal/PWi
/testbench_tmcterminal/Vi
323830323932
3139393233363735383439333034393334
1a77076bf5b5de5402ca2cfc2a59f878
5531343035343031383232393 2
1a77073ec481ee6136fa1dc4186bc14a
Now
6940 ns
Cursor 1
0 ns
200 ns
400 ns
600 ns
800 ns
1000 ns
1200 ns

Fig. 4. Simulation of the User Registration phase

The second phase is the Card Activation Phase as shown by Fig. 5.

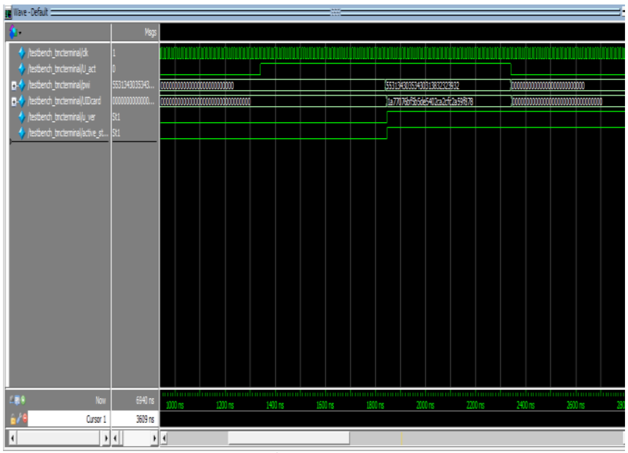
Wave - Default
Msgs
/testbench_tmcterminal/clk
/testbench_tmcterminal/U_act
/testbench_tmcterminal/pwi
/testbench_tmcterminal/UIDcard
/testbench_tmcterminal/ve
/testbench_tmcterminal/active_st...
Now
6940 ns
Cursor 1
1000 ns
1200 ns
1400 ns
1600 ns
1800 ns
2000 ns
2200 ns
2400 ns
2600 ns

Fig. 5. Simulation of the Card Activation phase

The third phase involved the User-Service Binding with User Authentication Transaction and Account Access Transaction.

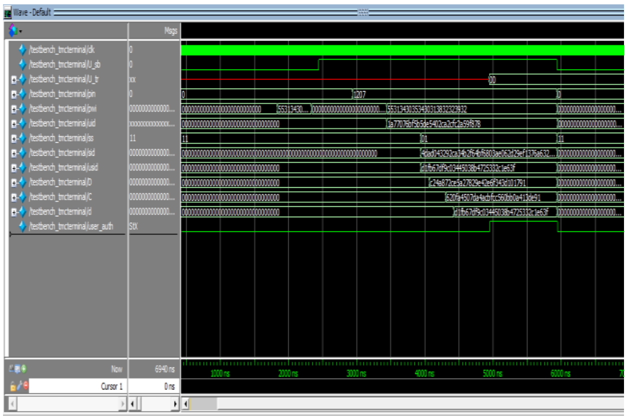

Fig. 6. Simulation of the User-Service binding and User Authentication Transaction

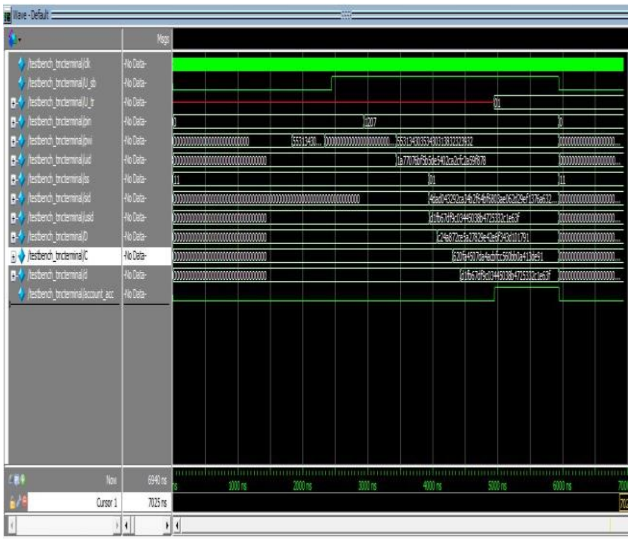

Fig. 7. Simulation of the User-Service binding and Account Access Transaction

B. *Synthesis Results*

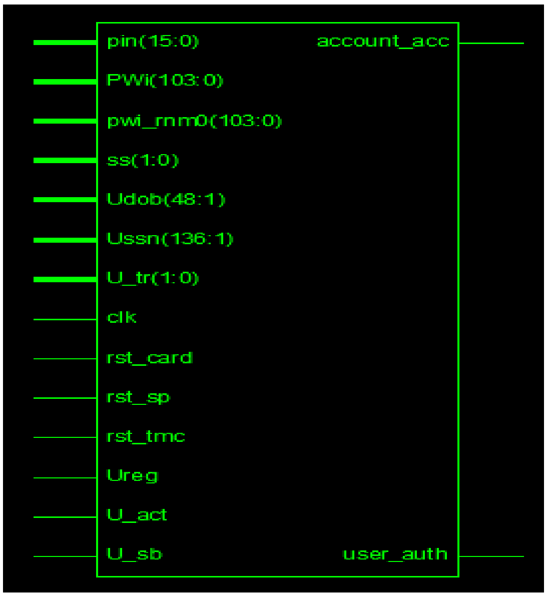

Fig. 8. Register Transfer Level Schematic of the Top block from Xilinx

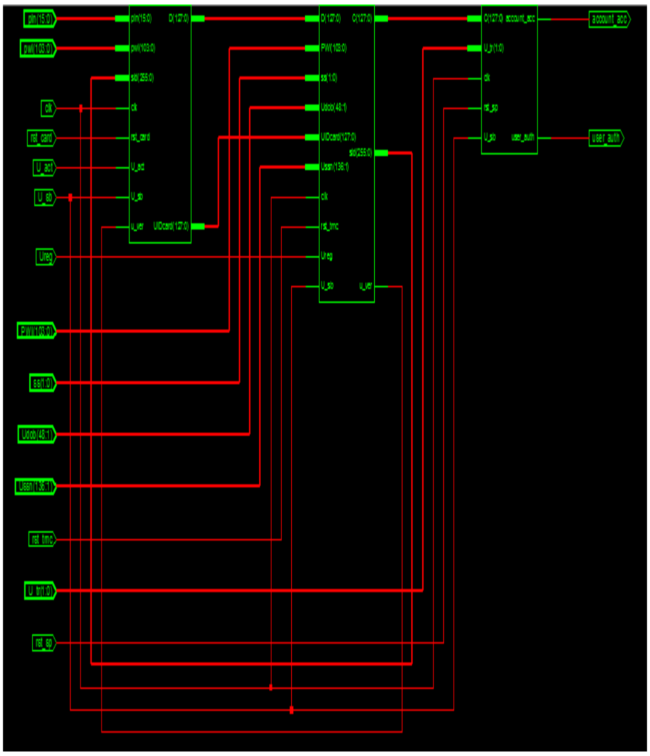

Fig. 9. Register Transfer Level Schematic of the Internal Blocks: Trusted Management Center, Smart Card and Service Provider

VIII. Conclusion

Our work depicts the applicability of the system proposed by Hong Mei and Guo Hui. We also included certain modifications to enhance the security of the system. In this paper we have included all the simulation waveforms which show that our Verilog codes perform without any error. Then we synthesized all the blocks under one top module and the resultant register transfer level schematic diagrams obtained have also been included.

ACKNOWLEDGMENT

Authors would like to express their gratitude towards Almighty for giving them the opportunity to do this work. They would also like to thank Dr Arshad M. Chowdhury, Chairman of the ECE Department at North South University.

REFERENCES

1. K. Hoon and C. Ronnie D., 'A Review of Smartcard Security Issues', Journal of Security Engineering, vol. 8, no. 3, 2011.
2. H. Mei and G. Hui, 'Design of Multi-Service Smart Card Systems for High Security and Performance', International Journal of Security and its Applications, 2009.
3. L. Lamport, 'Password authentication with insecure communication', Commun. ACM, vol. 24, no. 11, pp. 770-772, 1981.
4. Wen-Shenq Juang, 'Efficient multi-server password authenticated key agreement using smart cards', IEEE Transactions on Consumer Electronics, vol. 50, no. 1, pp. 251-255, 2004.
5. Smart Card Alliance, 'What Makes a Smart Card Secure?', 2008
6. R. Munir and S. Bin Khalid, 'Secure Debit Card Device Model'.
7. National Institute of Standards and Technology, 'Secure Hash Standard (SHS)', Federal Information Processing Standards Publications, 2012.
8. National Institute o Standards and Technology, 'Announcing the ADVANCED ENCRYPTION STANDARD (AES)', Processing Standards Publication, 2001.