International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 12– No.31, July 2020 – www.ijais.org

Improved Secure Biometric Authentication Protocol

Bakare K. A.
Dept. of Computer Science
Ahamadu Bello University
Zaria, Nigeria

Junaidu S. B.
Dept. of Computer Science
Ahamadu Bello University
Zaria, Nigeria

Ahmed M. Y.
Dept. of Computer Science
Ahamadu Bello University
Zaria, Nigeria

ABSTRACT
This paper presents an improvement to algorithm used in remote biometric authentication protocol [1]. A good remote biometric authentication protocol is expected to ensure secrecy of biometric data against all kinds of threats. This paper improves the security protocol algorithm by ensuring that secrecy of data is maintained when threat model in [1] is tested on the protocol. Security protocols should be flexible enough to guarantee Secrecy of data regardless of the intruder model. Network threat model in [6] is considered the foundation when analyzing security protocols and the remote biometric authentication protocol [1] was implemented on this threat model. The work of [6] assumes that the adversary has taken over the communication channel which allows the adversary to intercept and modify messages in transit. Scyther tool for protocol analysis implements threat model of [2] and was used in verifying and validating the property of Secrecy-of-Data for remote biometric authentication protocol. Therefore, the scope of this paper entails improving only the property of Secrecy-of-Data for remote biometric authentication protocol when faced with a threat model.

General Terms
Remote biometric authentication communication protocol security

Keywords
Security Protocol, Scyther protocol verifier, Dole-Yao threat model, Gavin-Lowe threat model

1. INTRODUCTION
Security protocols are groups of programs that enable secured exchange of messages between two or more communicating systems. They are useful in securing message exchanges during electronic payment, e-banking, and more recently electronic elections. A well designed security protocol is expected to have at the minimum three properties; Secrecy of Data, Liveness and Authentication.

With advancement in Technology and its attendant increase in cybercrime, new technological solutions are gradually being implemented to provide strong authentication. Biometrics authentication is an ideal means of identification and access control.

The processes for User authentication entail all the human-to-computer or computer-computer interactions that enable such user to gain access to certain services on a system.

The term biometrics refers to any human characteristics such as fingerprint, iris, hand geometry or even key stroke.

It is very unlikely that biometrics from two separate individuals will be the same; even if these individuals are twins.

Attacks on stored biometric data, man-in-the-middle, attack on biometric readers or transmission channel, compromise by disgruntle staff etc. are possible threats to any biometric authentication system. Therefore, a secured biometric authentication protocol must guarantee security of biometric data and provide secure communication against all kinds of adversaries.

Network threat model is considered the foundation when analyzing security protocol. In [2] threat model assumes that:

- That the adversary has taken over the communication network.
- That the cryptography technique used cannot be decrypted by adversaries
- Messages being transmitted over communication channels are intangible.

In 1996, [3] presented an improvement of the threat model by [2.]. Lowe's threat model assumes that an adversary has compromised a system (User) in addition to having control over the communication network.

2. REVIEW OF RELATED WORKS
There are several works that have been proposed and implemented in the past by various researchers to implement biometric authentication within a network environment. Some of the most important works are cited below.

I. In [14], "Comparative study on various authentication protocols in wireless Sensor Networks", the paper compared and analysed the security of wireless transmission of key management protocols, lightweight authentication protocols and broadcast authentication protocols. However, protocols studied here do not include LAN authentication protocols.

II. The work of [15], Google Patents; "Biometric authentication of client network connection", the patent proposed architecture for client authentication when requesting for network resource. The architecture requires transmission of fingerprint template to remote server for verification. Limitation: transmitting captured fingerprint over the network poses a serious threat as fingerprint once comprised cannot be used again.

III. The work of [16] proposed an "infrastructure for cloud computing Authentication using biometric-Kerberos scheme on Strong Diffi-Hellman- DSA key exchange". However, this infrastructure relies on third party to host database of captured fingerprint which also poses a threat while transmitting captured fingerprint.

IV. In the paper "Biometric Authentication Technique with Kerberos for Email Login" [17], the system saves biometric details on clients systems and sends copies to authenticating server. This approach is prone to numerous attacks as the only security mechanism in place is an SSL service.

V. The work of [18] proposed a new scheme that utilizes "RFID tag and biometric features for authentication". This scheme caters for database security, since only partial data is saved on database. However, loss of RFID tag, carrying RFID on clients are issues that have not been contented.

3. SYSTEM ARCHITECTURE AND DESIGN

3.1 Remote Biometric authentication protocol

The published protocol [1] is analyzed on a simulated on-line banking service that requires biometric authentication (Figure.1). The user is requested to authenticate to the system using his biometrics, e.g. his fingerprint, before he is allowed to proceed with the transaction. This remote biometric authentication protocol for on-line banking has three intended security properties: Secrecy of Biometric Data, Liveness and Authentication.

3.2 Protocol Sequence

Step 1- A user intending to access a bank service sends his account details and amount to activate the bank service.

Step 2 – The bank responds with a signed request for biometric authentication through the user to the Biometric Authentication Server (BAS)

Step 3 - The BAS initiates biometric authentication services and signs its request to user to send biometric data using the device attached to his computer.

Step 4 – The user responds with his captured biometric data and signs the packet with BAS public key.

Step 5 – The BAS requests for the user to verify the earlier sent biometric data

Step 6 – The user verified the data

Step 7 – BAS machine verifies the submitted data against existing template and issues a match- success or match-failure result to user for onward transmission to Bank server.

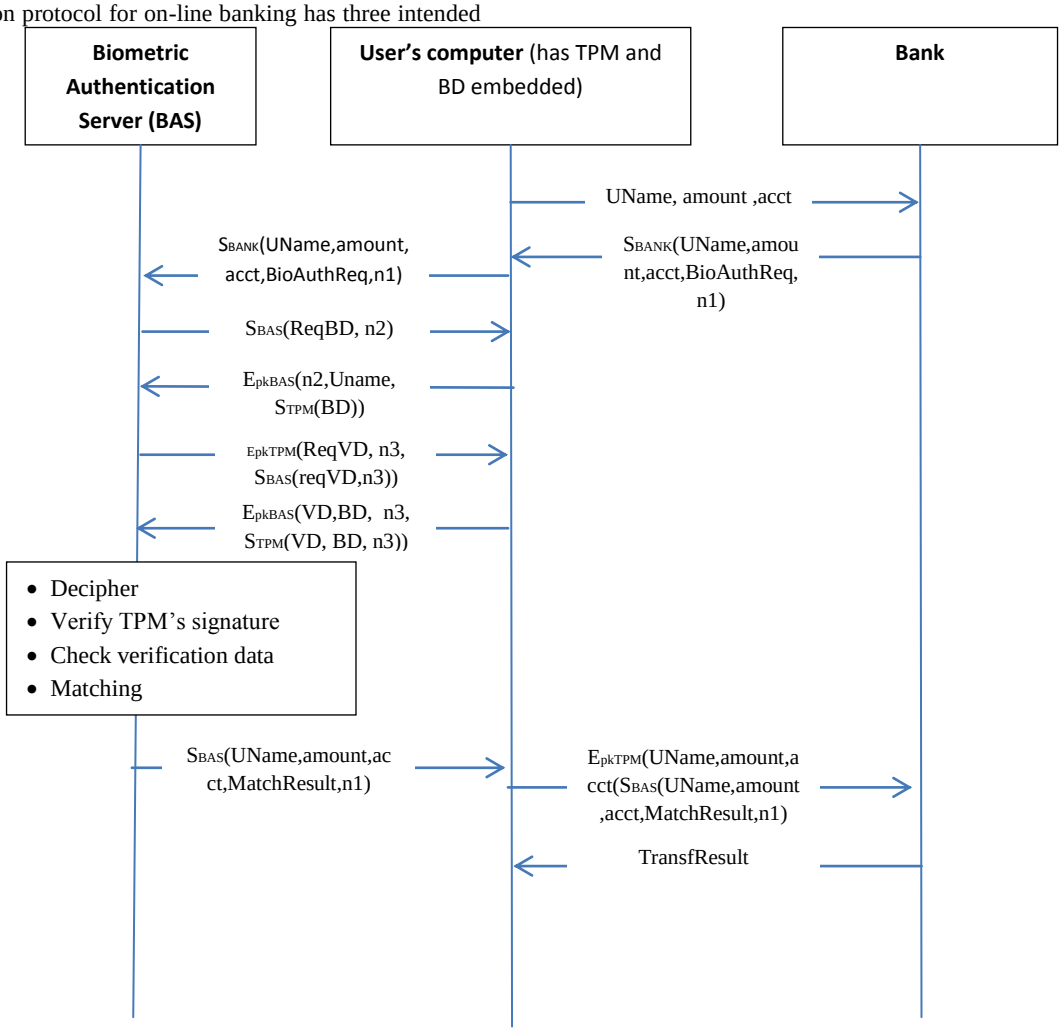

Figure 1: Message Sequence Chart for published protocol depicting the implementation of the protocol in an online banking service request

3.3 Analyzing protocol on Gavin-Lowe model

Secrecy property of protocol requires that certain information is not revealed to an adversary, even though this data is communicated over an untrusted network. The remote biometric authentication protocol (Figure 1) was analyzed using Scyther protocol verifier tool; Table 1 shows the result of the analysis. The result shows that at protocol initiation, messages; Username (uN) and generated nonce (ni) are vulnerable on a compromised system, this vulnerability will enable an adversary to replay those messages to authenticating system and therefore gain access to other services.

Table 1: Scyther Protocol verifier showing vulnerability in the published protocol

PROTOCOL ns1 (USER, BANK, SERVER)				
Entity	**Message execution number**	**Secrecy Test**	**Result**	**Attack type**
USER	U1	Secret uN	**Fail**	Falsified- At least 1 attack possible
	U3	Secret ni	**OK**	No attacks
	U8	Secret BD	**OK**	No attacks
	U12	Secret BD	**OK**	No attacks
	U16	Secret MatchResult	**OK**	No attacks
	U21	Secret MatchResult	**OK**	No attacks
	U22	Secret Result	**OK**	No attacks
SERVER	S2	Secret BA	**OK**	No attacks
	S5	Secret n2	**OK**	No attacks
	S8	Secret BD	**OK**	No attacks
	S11	Secret BD	**OK**	No attacks
	S15	Secret MatchResult	**OK**	No attacks
BANK	B1	Secret BuN	**Fail**	Falsified - At least 1 attack
	B5	Secret ni	**Fail**	Falsified - At least 1 attack
	B6	Secret Bacct	**Fail**	Falsified - At least 1 attack
	B8	Secret MatchResult	**Fail**	Falsified - At least 1 attack
	B9	Secret nii	**Fail**	Falsified - At least 1 attack

4. IMPROVED BIOMETRIC AUTHENTICATION PROTOCOL

To ensure Secrecy of Data at protocol initiation a Kerberos Server was introduced into the network environment. Kerberos is network authentication protocol used for identifying entities on a compromised network; only approved users may access the available services

The architecture comprises of a Client device that is authenticated by Kerberos server when the device boots up.

The Kerberos Server consist of an Authentication Server (AuthServer) which is used to authenticate client machines, a Ticket Granting System (TGSystem) which is used to generate tickets for clients' verification. The AuthServer and TGSystem constitute a Key Distribution Centre (KDC).

Furthermore, within the environment there also exists a Biometric Authentication Server (BioAuthServer) and also an Application Server (AppServer) that serves its applications to verified users only.

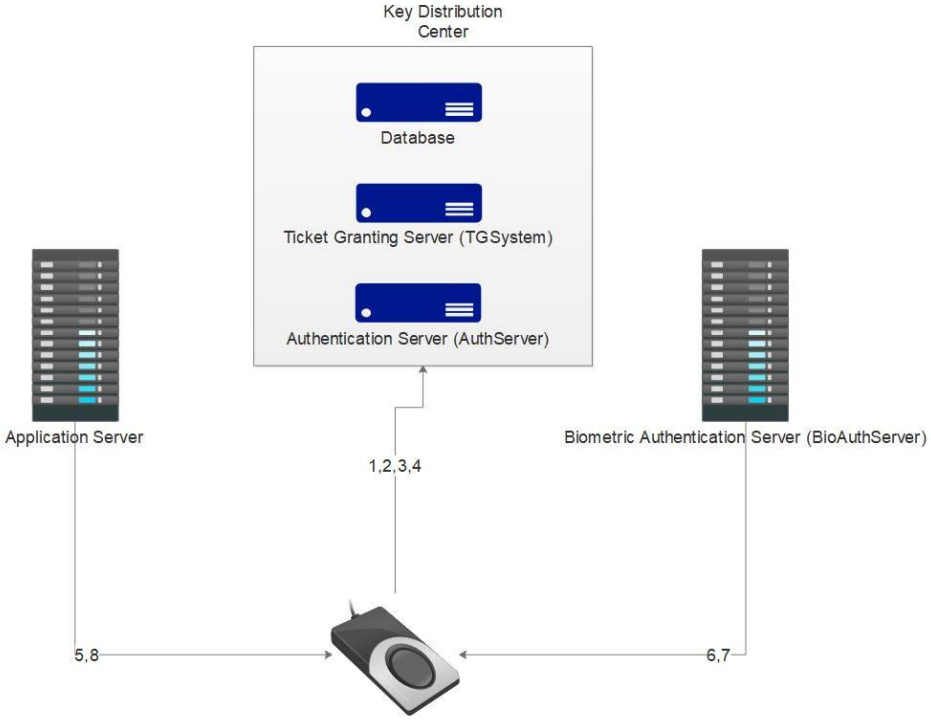

Figure 2: System Architecture for Improved Biometric Authentication Protocol

4.1 Improved protocol's Sequence

Step 0- At registration of devices, a unique ID, password is created for both Client and Application Server and stored by the KDC

Step 1 – At process initiation. The biometric capturing device identifies itself by requesting and obtaining a ticket from TGSystem using its registered ID. At this point, communication is between devices only.

Step 2: The AuthServer verifies if ID supplied by Biometric device exits in its database. If the ID exists, the AuthServer then checks if the TGSystem is available, if both queries return true, a secret key for TGSystem is generated, also another secret key is generated for the client using hash of user password.

AuthServer generates a session key (SK-I) and encrypts it using Client key stored in its Database. The encrypted key is shared between client and TGSystem.

AuthServer generates a Ticket. This ticket (T-1) is generated by encrypting a string combination of Client ID and network address, lifetime, timestamp and session key. This string combination is encrypted with TGSystem's secret key and can only be decrypted by the TGSystem.

Step 3: The KDC responds to the client's ticket request by sending an encrypted message. This message is a combination of the generated session key (SK-I) and Ticket (T-1). The message can only be decrypted by the client whose secret key was used for encryption.

Step 4: On receiving the response from KDC, the client uses its secret key which was created at registration point, to decrypt the message. The clients then extracts the Session Key (SK-I) and assigned Ticket (T-1). The client further generates message combination of Client ID, Network address and client machine time stamp. This message is then encrypted with the extracted Session Key) SK-I). This message serves as an authenticator (Auth-I) of the client to the TGSystem. Thereafter the client sends the generated authenticator and extracted Ticket (T-1) to TGSystem's request for ticket to allow access the Application Server.

Step 5: On receiving the request for Application Server Ticket from the client, the TGSystem extracts Session Key (SK-I) which was used to encrypt the client request. Using this key, the TGSystem decrypts the Authenticator and verifies Client ID and network address against the same value which was used in generating the Authenticator, furthermore the validity of timestamp is also checked.

IF the Client ID, network address, timestamp are all valid, the TGSystem generates a second Session Key (SK-II). SK-IIserves as a secret key between Client and Application Server.

The TGSystem then retrieves the Application Server Secret Key (created at registration phase) from its database. The TGSystem then generates a Service Ticket (ST-I) contain a combination of Client ID, network address, timestamp and Session Key (SK-II). The ST-I is encrypted using Application Server Secret Key, ensuring that only the Application Server can decrypt the Service Ticket (ST-I).

The TGSystem then encrypts the Service Ticket (ST-I) using Session Key (SK-I) and sends to the client.

Step 6: On receiving the encrypted Service Ticket (ST-I), the client decrypts the message using its SK-I and extracts the SK-II. The clients then generates a new Authenticator (Auth-II) which is contains a combination of Client ID, network address and Timestamp and encrypts the Authenticator (Auth-II) with extracted Sk-II. The encrypted Authenticator (AUth-II) is then sent to the Application Server.

Step 7: Using its Secret Key, the Application server decrypts the Service Ticket (ST-I) and extracts SK-II, Client ID, network address and Timestamp.

The Server validates the extracted values, if all values matches; a control bit of 1 is added to the extracted timestamp. The Timestamp is then encrypted using SK-II and sent to client. This exchange between Client and Application Server serves to verify and validate both systems to each other, thereby creating a "trust" between them.

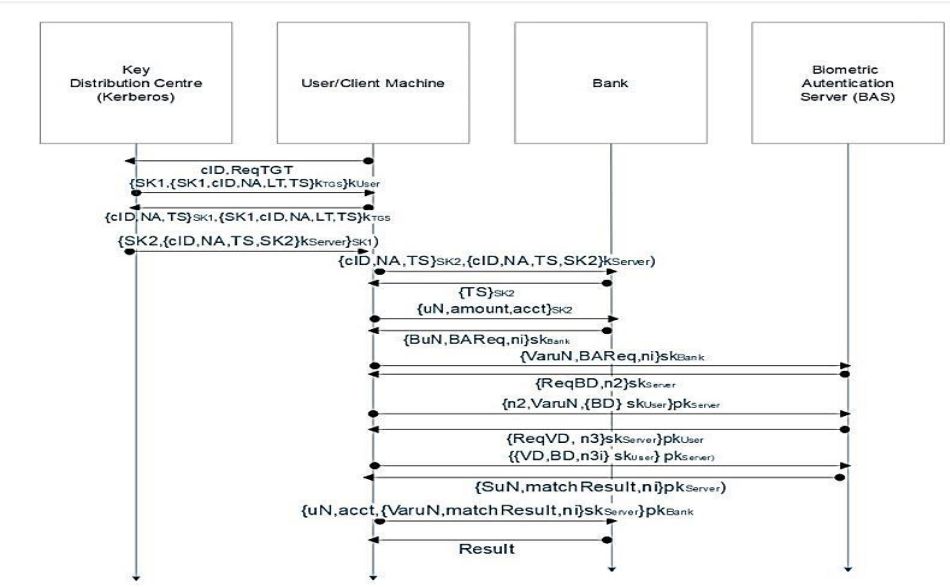

Figure 3: Message Sequence Chart for improved protocol depicting the implementation of the protocol in an online banking service request

4.2 Result and Discussion

A User Bob sends a transaction request message to his banker Alice (Figure 4). The message contains his ID (Uname), amount for transaction and his account number. An Adversary having gain access into the untrusted network will be able to eavesdrop on the initial message (uName, amount, acct) sent to bank to initiate a transaction, mainly because the message was neither signed nor encrypted. Although at this point, the data is of no value to the adversary because a transaction cannot be completed without biometric authentication, but these hijacked message can further be used in guessing or parsing messages within the untrusted network. Therefore the Secrecy of data at protocol initiation does not hold for the old protocol.

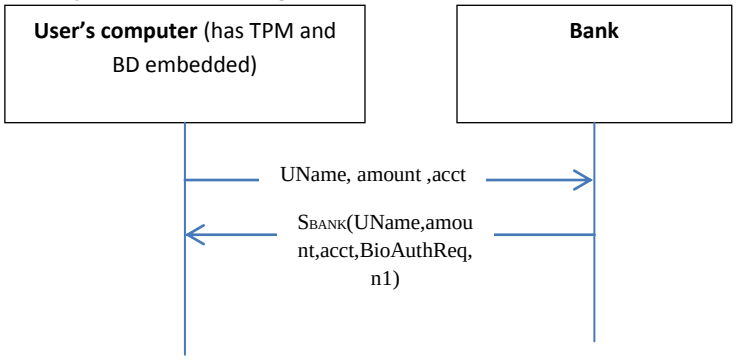

Figure 4: Old Protocol initiation showing initial communication between User and Bank

Table 2: Secrecy verification result for old Protocol

PROTOCOL ns1 (USER, BANK, SERVER)				
Entity	**Message execution number**	**Secrecy Test**	**Result**	**Attack type**
USER	U1	Secret uN	**Fail**	Falsified- At least 1 attack possible
	U3	Secret ni	**OK**	No attacks

The improvement to the old protocol is by introducing a Kerberos server role into the protocol, rather than a client requesting to the banking server directly, the client must first authenticate his workstation via the Kerberos certificates, thereafter, the system is issued a ticket by the Kerberos server, which the banking server can always verify from the Kerberos server whenever the client requests for banking services.

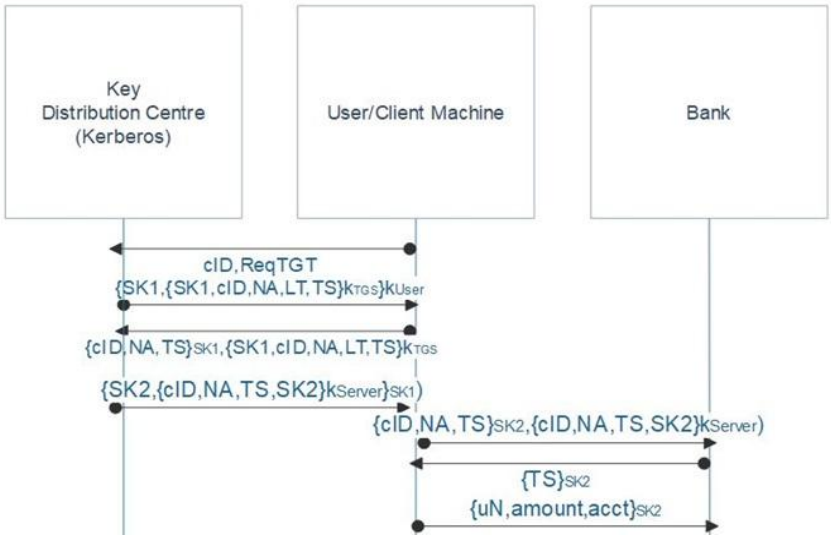

Figure 5: Improved Protocol initiation showing initial communication between User, Kerberos server and Bank

4.2.1 Protocol Description

The protocol is named **ns1**. There are three roles (actors) defined in this protocol; they are User, Bank Server, Key distribution Centre (Authenticating Server, Ticket Granting server)

These roles can either be a message initiator or a message responder. The messages are sometimes encrypted and are stated as (Message) k.

For each run of the protocol, the roles (User, Bank and Server) are expected to makes available memory allocation for some predefined variables and also to carry out independent computations. Some of the variables (FRESH) retain their values through the execution process whiles others (VAR) change values as the protocol runs.

The user system initiates a request to Authenticating server(AS) depicted as **send_1a(User,AS,cID,ReqTGT);**

the message request is comprised of client Id (CID) and Request for Ticket Granting Ticket to provide the user with an authentication ticket(RegTGT).

The Authentication Server (AS) recieved the message request from User system depicted as

recv_1a(User,AS,cID,ReqTGT); and responds with its own message to the user as **send_2a(AS,User,{SK1,{SK1,cID,NA,LT,TS}k(TGS)}k(User));.**

The message consists of two parts, a secret key (SK1) and an encrypted message for the TGS only**,{SK1,cID,NA,LT,TS}k(TGS)** only the TGS can read this part of the messages being that it is encrypted with TGS's secret key **k(TGS)** the user decrypts the message meant for him and obtain the SK1 and forwards the remaining encrypted message to the TGS.

SK1 is then tested to confirm that it cannot be hijacked. (claim_U2a(User,Secret,SK1));the result shows that is not vulnerable to any form of attack.

Thereafter the user appends additional information to identify itself to the TGS and forward this new message to the TGS. ((send_3a (User,TGS,{cID,NA,TS}SK1,{SK1,cID,NA,LT,TS}k(TGS));))

The message consists of two parts, the first part contains the user ID, network address and Timestamp, while the second part consists of the encrypted message from the TGT which was earlier encrypted with the TGS key. Also at this point, testing was done to see if the network address supplied by the user is vulnerable and it is not.

The TGS decrypts the message sent and verifies the Network address and Timestamp which was supplied by both the TGT and the User. If all checks well, the TGS then responds to the user's message ((send_4a (TGS,User,{SK2,{cID,NA,TS,SK2}k(Server)}SK1)));

The message contains two part also, the first part is consist of another secret key (SK2) and the second part is for the bank server and its encrypted with bank server secret

The user receives the message for the bank server

recv_4a (((TGS,User,{SK2,{cID,NA,TS,SK2}k(Server)}SK1);)) and decrypts the SK2 and then forwards the remaing part to the bank server. the user then send to the bank a message

((send_5a (User,Bank,{cID,NA,TS}SK2,{cID,NA,TS,SK2}k(Server));))

The bank extracts the SK2 to verify the user thereby creating a trust relationship.

The user can now finally send its request to bank for banking services

send_1(User,Bank, {uN,amount,acct}SK2);

Also testing for vulnerability of user data show negative vulnerability at this juncture.

This shows that with the new protocol a user is able to verify itself on the network without exposing any sensitive data to adversaries.

4.2.2 Protocol implementation on Scyther tool

protocol ns1(User,Bank,Server,AS,TGS){

role User {fresh VD,amount,acct: Nonce;

var ReqVD,n3i,n2,BAReq,ReqBD,VaruN,ni,SK1,SK2,NA,LT,TS,matchResult,Result;

fresh uN,BD,cID,ReqTGT,

send_1a(User,AS,cID,ReqTGT);

recv_2a(AS,User,{SK1,{SK1,cID,NA,LT,TS}k(TGS)}k(User));

claim_U2a(User,Secret,SK1);

send_3a (User,TGS,{cID,NA,TS}SK1,{SK1,cID,NA,LT,TS}k(TGS));

claim_U3a(User,Secret,NA);

recv_4a (TGS,User,{SK2,{cID,NA,TS,SK2}k(Server)}SK1);

send_5a (User,Bank,{cID,NA,TS}SK2,{cID,NA,TS,SK2}k(Server));

recv_6a (Bank,User,{TS}SK2);

send_1(User,Bank, {uN,amount,acct}SK2);

claim_U1(User,Secret,uN);

}

role AS {var cID,ReqTGT;

fresh SK1: Nonce;

fresh NA,LT,TS;

recv_1a(User,AS,cID,ReqTGT);

send_2a(AS,User,{SK1,{SK1,cID,NA,LT,TS}k(TGS)}k(User));

claim_AS2a(AS,Secret,SK1);

claim_AS2b(AS,Secret,TS);

}

role TGS {fresh SK2:Nonce;

var NA,LT,TS,SK1,cID;

recv_3a (User,TGS,{cID,NA,TS}SK1,{SK1,cID,NA,LT,TS}k(TGS));

send_4a (TGS,User,{SK2,{cID,NA,TS,SK2}k(Server)}SK1);

}

Table 3 below shows the result of new protocol analysis conducted using Scyther tool. The result shows that at protocol initiation, user credentials are safe even if an adversary has complete control over the system, unlike the initial result of analysis (Table 1) which shows that at protocol initialization user credentials (Username and generated nonce) are vulnerable on a compromised system.

Table 3: Scyther Protocol verifier showing no vulnerability in the improved protocol

PROTOCOL ns1 (USER, BANK, SERVER)				
Entity	**Message line number**	**Secrecy Test**	**Result**	**Attacke type**
USER	ns1.U1	Secret uN	**OK**	No attacks
	ns1,U3	Secret ni	**OK**	No attacks
	ns1,U8	Secret BD	**OK**	No attacks
	ns1, U12	Secret BD	**OK**	No attacks
	ns1, U16	Secret MatchResult	**OK**	No attacks
	ns1,U21	Secret MatchResult	**OK**	No attacks
	ns1, U22	Secret Result	**OK**	No attacks
SERVER	ns1, S2	Secret BA	**OK**	No attacks
	ns1,S5	Secret n2	**OK**	No attacks
	ns1, S8	Secret BD	**OK**	No attacks
	ns1, S11	Secret BD	**OK**	No attacks
	ns1, S15	Secret MatchResult	**OK**	No attacks
BANK	ns1, B1	Secret BuN	**OK**	No attacks
	ns1, B5	Secret ni	**OK**	No attacks
	ns1, B6	Secret Bacct	**OK**	No attacks
	ns1, B8	Secret MathcResult	**OK**	No attacks
	ns1, B9	Secret nii	**OK**	No attacks

5. SUMMARY AND CONCLUSION

The result of this paper shows that the improved protocol satisfies the property of Secrecy-of-Data during packet transmission when subjected to both [6] and [2] threat model. Introducing a Kerberos server role into the old Remote Biometric Authentication protocol guarantees that the initial request satisfied the property of Secrecy. The establishment of secret key (SK2) between user and Bank by the Kerberos server also ensures that the user does not need to re-authenticate each time he wishes to perform a transfer transaction within a session.

6. FUTURE RESEARCH

The major properties of any security protocols that need to hold true are Secrecy and authentication. The improved remote biometric authentication protocols satisfied the property of Secrecy, future work will entail the modeling, verification and analysis of Authentication Protocol property.

7. ACKNOWLEDGMENTS

Profound gratitude goes to all those who have contributed in one way or the other throughout the period of researching and writing this thesis. This accomplishment would not have been possible without them.

8. REFERENCES

[1] SALAIWARAKUL, "Secured Biometric Authentication protocols" (2014)**113-128**.

[2] DOLEV, A.C. YAO, On the security of public key protocols. IEEE Trans. Inf. Theory 29(2), 198–207 (1983) Biometric Cryptosystems: Authentication, Encryption and Signature for Biometric Identities.

[3] LOWE, G.: An attack on the Needham-Schroeder public-key authentication protocol. Information Processing Letters 56 (1995) **131-133.**

[4] CHEN, L., PEARSON, S., VAMVAKAS, A.: Trusted Biometric System.

[5] SULOCHANA.V, PARIMELAZHAGAN.R (2016). Highly Efficient Kerberos Style Authentication and Authorization for Cloud Computing. International Journal of Computer Science and Network Security, **VOL.16 No.11,** November 2016.

[6] ZHOU, XUEBING **(2012).** Privacy and Security Assessment of Biometric Template Protection. (Phd Thesis). Technische Universität, Darmstadt.

[7] DOLEV, A.C. YAO, On the security of public key protocols. IEEE Trans. Inf. Theory 29(2), 198–207 (1983) Biometric Cryptosystems: Authentication, Encryption and Signature for Biometric Identities.

[8] Precise Biometrics. UNDERSTANDING BIOMETRIC PERFORMANCE EVALUATION.

[9] UMUT ULUDAG (2006). Secure Biometric Systems. (Doctor of Philosophy Computer science and Engineering). Michigan State University

[10] PATRICK WIEDERKEHR (2009). Approaches for simplified hotspot logins with Wi-Fi devices.(Master of Science in Computer Science). Swiss Federal Institute of Technology Zürich

[11] TAEKYOUNG KWON AND JAE-IL LEE. Practical Digital Signature Generation using Biometrics. Sejong University, Seol, Korea.

[12] J. KELSEY, B. SCHNEIER, D. WAGNER, Protocol interactions and the chosen protocol attack, in 5th International Workshop on Security Protocols, ed. by B. Christianson, B. Crispo, T.M.A. Lomas, M. Roe, Paris, France. Lecture Notes in Computer Science, vol. 1361 (Springer, Berlin, 1997), pp. 91–104

[13] Operational_Semantics_and Verification of Security Protocols

[14] S.Raja Rajeswari et al. Comparative Study on Various Authentication Protocols in Wireless Sensor Networks. (2016)

[15] Eric E. Miller, Biometric authentication of client network connection. (2004)

[16] Hamid Roomi Talkhaby. "infrastructure for cloud computing Authentication using biometric- Kerberos scheme on Strong Diffi-Hellman- DSA key exchange. (2016).

[17] Rashmi Hegde 2015, in the paper "Biometric Authentiation Technique with Kerberos for Email Login. (2015).

[18] Sonali Patil et al,. Design and implementation of secure biometric based authentication system using RFID and secret sharing. (2017).