Human-centric Computing and Information Sciences
a SpringerOpen Journal

RESEARCH Open Access

# Smart card based time efficient authentication scheme for global grid computing

Jaspher Willsie Kathrine Gnanaraj[1*], Kirubakaran Ezra[2] and Elijah Blessing Rajsingh[3]

* Correspondence:
meet.katee@gmail.com
[1]Department of Information Technology, Karunya University, Coimbatore, Tamilnadu, India
Full list of author information is available at the end of the article

## Abstract

Decentralization in every walk of life has resulted in the development of Global Grid networking. Data sharing and access depends on their availability, capability, cost and user requirements. One of the needs for a secure Grid Environment is a strong authentication for users. Since Authentication is the entry point into every network, a novel smart card based authentication scheme has been proposed. The proposed authentication scheme utilizes the biometric data embedded in a smart card along with the ID and password of the user. The Time efficient performance of the proposed scheme in comparison with the existing Secure Socket Layer based authentication scheme is discussed. The attacks which the proposed scheme is able to withstand are also discussed.

**Keywords:** Grid computing; Authentication; Authorization; Biometric

## Introduction

Grid computing involves sharing heterogeneous resources which are located in geographically distributed places belonging to different administrative domains [1]. Grid data sharing is not file exchange but rather access to computers, software, data and other resources. Grid involves the creation of a dynamic Virtual Organization (VO). Each virtual organization comprises of users and their resources and any other services (S) joined by a common goal [2]. Each of the user or resource is available from different administrative domains (DO). Each user/resource have their own trust policy which requires a local to global and global to local mapping of the access policies as discussed in [3].

The basic security for the Globus Toolkit (GT 4) is the Grid Security Infrastructure (GSI) [4,5]. It depends on the Public Key Infrastructure (PKI), X.509 Proxy certificates and Transport Layer Security (TLS) for authentication. GSI involves third party verification for authorization. The GT framework is based on the Open Grid Services Architecture (OGSA) which uses the Secure Socket Layer (SSL) based on TLS. The GSI security is secure enough but has scalability problems [5].

The existing authentication schemes are based on the user name and the password and certificates which are generated by a secure Certificate Authority (CA) [5]. The existing authentication schemes belong to two factor authentication scheme which involves user name/password and some cards like those used in Banks. The Security for the Grid Environment is deployed in the middleware which is used to access the grid network. Examples of Grid middleware are UNICORE (Uniform Interface to Computing Resources) [6], Globus [6], Legion [7] and Gridbus [8].

In [9] a Four-Factor based Biometric Authentication has been proposed. But the addition of location does not guarantee the avoidance of insider attack. The proposed authentication scheme optimizes the security of a grid environment by adding more features like biometric data in a smart card for optimal authentication.

User authentication has been in discussion for a long time to enhance the security of any system at the entry level itself. Many methods such as password based systems, ID based systems, and etc. have been used. A hash-chain based remote user authentication in which all the passwords are encoded is given in [10]. In all the initial remote based authentication systems, a verifier table is to be placed in the server side which becomes a problem if the server is compromised.

In this paper the remote based authentication system which is very much suited for the Grid Environment is considered. Based on the existing remote authentication systems, an enhanced system is designed. In order to avoid maintaining a verifier table Hwang et al., proposed a non-interactive smart card based scheme without verifier tables [11]. A finger print based remote user authentication scheme was proposed in [12]. This scheme was found to be vulnerable to masquerade attacks and many other attacks [13,14]. In [15-17], the biometric data itself is taken as a key for encryption/decryption. The secret data is extracted by using the biometric template as the key. The biometric data is to be stored in the server side and used for comparison. But for effective Biometric authentication, the process is to be done in the client side [18] to avoid any problem due to the server being compromised [19]. In [20], the method has been optimized with the matching being done in the server side. But the server does not store any biometric data in its database thereby protecting the privacy of the user.

The method in [20] provides a three factor authentication which is password – something the user knows; smart card – something the user has; biometrics – something the user is. A further enhancement to this type of authentication is to add a fourth factor thereby providing a four factor authentication [21]. The fourth factor can be the addition of location of the user – someplace the user is. The military data sharing requirements take into consideration the place in which the user is positioned so as to find the location of any valid/invalid user. So, the sensitive areas of application require security with some amount of privacy preservation. Section three gives an overview of the existing authentication systems in grid computing. Section four discusses the proposed security framework with reduced stages for authentication of a grid user.

## Existing security framework for grid

A Grid Environment is created by means of using general-purpose grid software libraries known as middle ware. The Grid environment is based on a layered architecture as shown in Figure 1.

From the Figure 1, the security features are seen in the middleware portion of the grid layer. The existing security solution uses Open Grid Services Architecture (OGSA) architecture [22]. This security feature used in GT is also used in Virtual Organization Membership Services (VOMS) [23] for the purpose of authorization also. The OGSA architecture uses GSI which in turn depends on the certificate based SSL for authentication and WS-Secure Conversation message transport and confidentiality. The existing

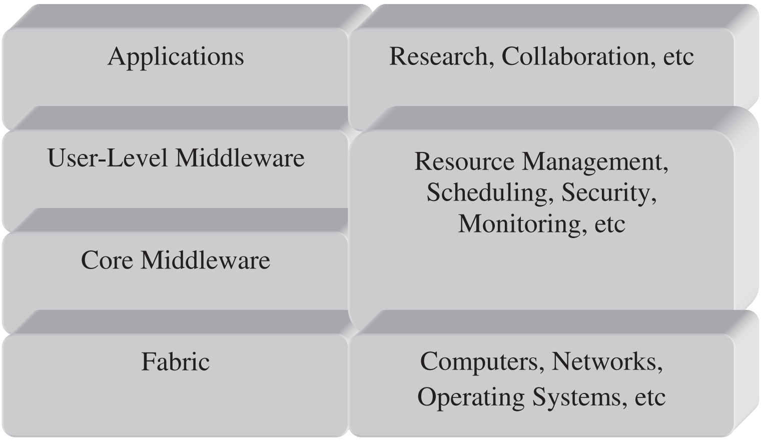
Applications
Research, Collaboration, etc
User-Level Middleware
Resource Management, Scheduling, Security, Monitoring, etc
Core Middleware
Fabric
Computers, Networks, Operating Systems, etc

**Figure 1** Layered architecture of grid.

system based on OGSA and GSI have some basic security solutions for solving the authentication and authorization criteria. The scalability, heterogeneity and increase in attacks have led to the need of a new security framework which is based on the existing architecture with additional features to tackle the day to day attacks. The next section discusses about the proposed authentication scheme.

## Proposed authentication system

The proposed authentication scheme has three phases such as the Registration phase, the Login phase and the Mutual Authentication phase. An additional password change phase is added to ensure that the user can change his/her password when required. In each phase distinct operations are defined for the user and the server. The proposed authentication methodology is shown in Figure 2.

In the Registration phase, based on the details provided by the user along with the inputs given by the server, the smart card data is stored and given to the user through a secure medium. Only during the Login and the mutual authentication phase is the user and the server authenticated to each other. Once the mutual authentication is a

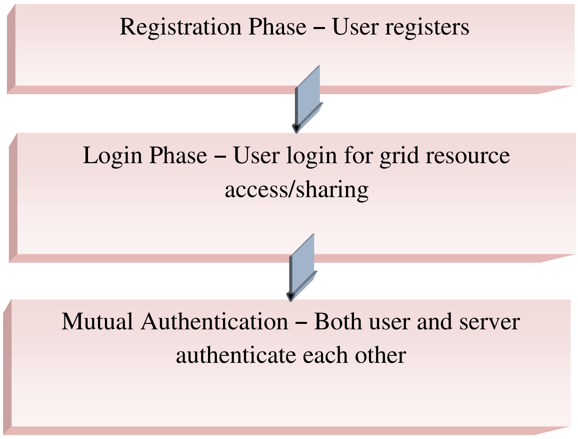
Registration Phase – User registers
Login Phase – User login for grid resource access/sharing
Mutual Authentication – Both user and server authenticate each other

**Figure 2** Proposed authentication methodology.

success, then the user can go on to the next operation involved in the data transaction. The triple DES along with any other light-weight encryption algorithm can be used. The process flow of the proposed authentication scheme is shown in Figure 3.

During the initialization phase, the server stores both the asymmetric and symmetric key in its database. Once a user requests for registration, the server accepts the user's hashed password in a secure way. This way assures that the server does not know the actual data and neither is data stored openly in any database within the server. The validity of the user is checked based on the comparison of the hashed data rather than the original data. This method of storage makes sure that the user's data is not lost under any circumstance.

All the hashed data are stored in the Registration/Authentication server's database and the encrypted data required for the further use of the user is stored in the smart

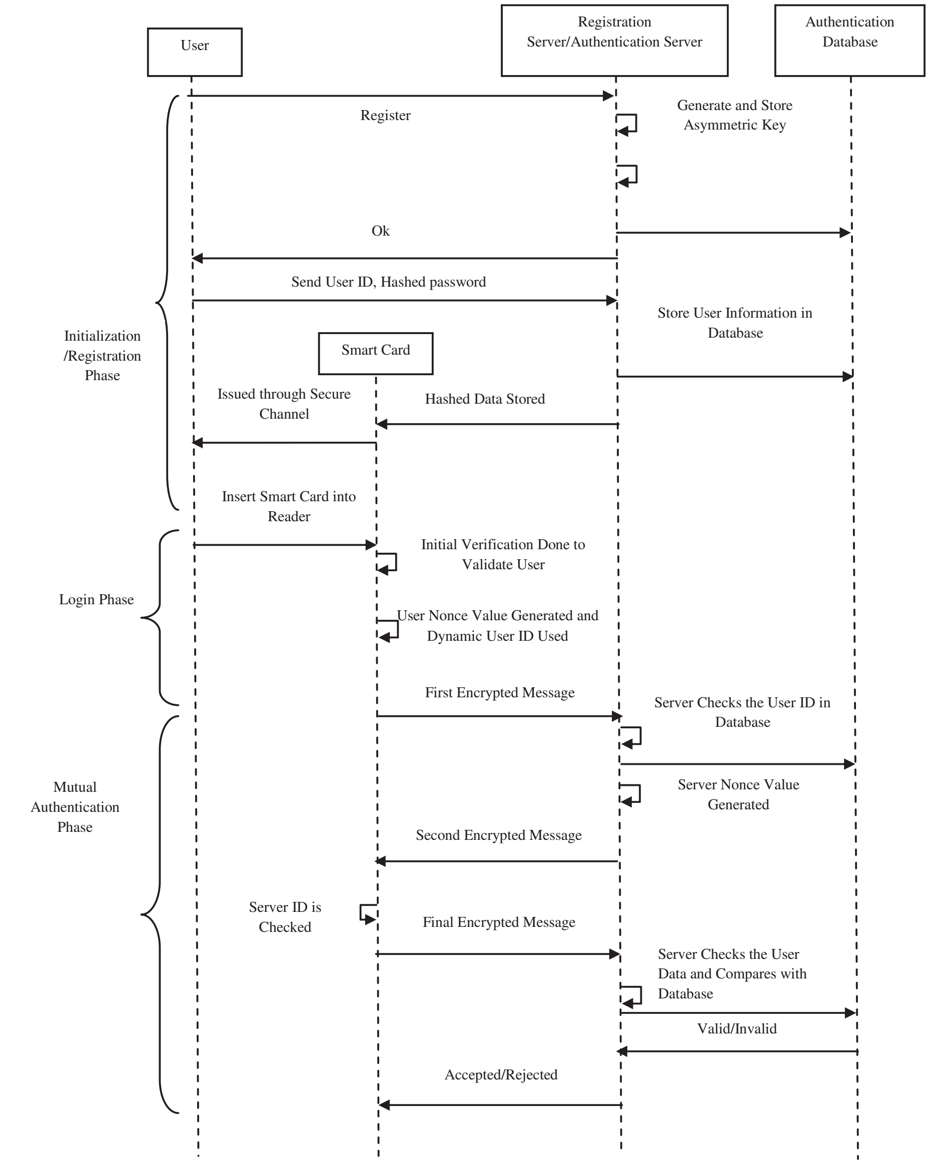
User
Registration Server/Authentication Server
Authentication Database
Register
Generate and Store Asymmetric Key
Ok
Send User ID, Hashed password
Store User Information in Database
Initialization /Registration Phase
Smart Card
Issued through Secure Channel
Hashed Data Stored
Insert Smart Card into Reader
Initial Verification Done to Validate User
Login Phase
User Nonce Value Generated and Dynamic User ID Used
First Encrypted Message
Server Checks the User ID in Database
Mutual Authentication Phase
Server Nonce Value Generated
Second Encrypted Message
Server ID is Checked
Final Encrypted Message
Server Checks the User Data and Compares with Database
Valid/Invalid
Accepted/Rejected

**Figure 3** **Process of the proposal authentication scheme.**

card and sent to the user. The user then uses the smart card for further access to the Grid environment. The smart card does the initial validation of the user and then forwards the user data to the server, where further authentication is the done. In the proposed scheme, both the user and the server validate each other and hence it is complete mutual authentication. Only when the user and the server both satisfy the validation criteria then the data transfer occurs. If the user validation does not succeed it is rejected or the user is requested to start the authentication from the beginning of the login phase. The next section gives the detailed explanation of each phase of the proposed authentication system.

## Details of the authentication scheme

This authentication scheme involves three factors, using a smart card which holds the data of what the user is i.e., the biometric data (B), what the user knows i.e., the password (PW), Identifier (ID) and the data that the user has i.e., smart card. This scheme has three phases such as Registration Phase, Login phase and a mutual authentication phase. The added features are the dynamic User ID (CID) and the dynamic server ID (SID). The Registration/Authentication server is configured to support symmetric and asymmetric encryption and decryption. In grid a client at one scenario can be the server at the next instant since it has to satisfy a need. Hence this system should be in such a way that each user is able to identify the requestor of the resource by his/her credentials. The notations used in this paper are given in Table 1.

### Registration phase

During the registration phase, the user tries to register for a grid membership within a Virtual Organization (VO). During the membership registration, the user is given a particular Identifier ($ID_i$). The user registers his/her biometric data ($B_i$) which maybe a fingerprint or an iris template. The user also selects a random number r and a password ($PW_i$).

The following are the series of steps done in the server:

a. Server generates public-private key pair ($p_k$, $s_k$) for asymmetric encryption/decryption.
b. Server generates a secret key x for symmetric encryption/decryption.
c. Both (x, $s_k$) are kept secure in the server.

**Table 1 Notations used in this paper**

| S | Server |
|---|---|
| $U_i$ | User |
| $S_{ID}$ | Identity of server $S_i$ |
| $ID_i$ | Identity of user $U_i$ |
| $B_i$ | Biometric data of $U_i$ |
| ($PW_i$) | Password of $U_i$ |
| h(.) | One-way hash function |
| x | The master secret key |
| ($p_k$, $s_k$) | Public-private key pair |
| $\oplus$ | The exclusive-OR operation |
| ‖ | Message concatenation |
| R | Random number generated by $U_i$ |
| $n_u$ | Nonce value generated by $U_i$ |
| $n_s$ | Nonce value generated by the server $S_i$ |

The operations done at the user side are:

a. The user records his/her user Identifier ($ID_i$)
b. The user records the biometric template ($B_i$)
c. The user selects a random string r and password ($PW_i$)

The user computes $SB_i = \delta\ (B_i) = h(r \oplus h(B_i))$.The value of $SB_i$ is sent to the server securely along with the one-way hash function h(.) of the Password and the identifier ($ID_i$) of the user. The server receives ($ID_i$ , $h(PW_i)$,$SB_i$) through a secure channel. By using the values sent by the user, the server computes, $y_i$ such that,

$$y_i = E_x(ID_i \| h(PW_i) \| SB_i) \quad (1)$$

where $E_x$ (**.**) represents the symmetric encryption using the secret key x The server stores the user's password ($PW_i$) and the related identifier ($ID_i$) of the user and the calculated $y_i$. The operations continued in the server side are:

a. Server computes $K = h(ID_i \ \| \ x)$
b. Server stores (K, h(.), $p_k$) in the smart card.
c. Server sends smart card to the user securely.

Once the user receives the smart card, a few entries are to be stored in it along with the data already available in the smart card i.e., $y_i$.

The following operations are done to confirm the registration:

a. The user enters the biometric data which can be an iris data /fingerprint $B_i$
b. The user encrypts the random number r with $PW_i$ such that $E_{pW_i}(r)$ is obtained.
c. $E_{pW_i}(r)$ is stored in the smart card.
d. $SB_i = \delta\ (B_i) = h(r \oplus h(B_i))$ is stored in the smart card.

### Login phase

A user $U_i$ is allowed to enter the grid environment using his/her smart card. The user enters his/her Password (PW)' and does a biometric scan denoted by $B_i^*$. The user's smart card retrieves the random value "r" from $E_{PW'}(r)$ by using the password (PW)' entered by the user $U_{i.}$ The smart card computes $SB_i^* = \delta\left(SB_i^{'}\right) = h\left(r \oplus h\left(B_i^{'}\right)\right)$. This value is compared with the already stored value of $SB_i = \delta\ (B_i) = h(r \oplus h(B_i))$ to confirm if the user is the same. Then the smart card generates a nonce value "$n_u$" and computes $M = (K \oplus n_u)$. Then $CID_i$ is calculated such that, $CID_i = h\ (ID_i \| n_u)$.

Then value of $C_0$ is computed such that,

$$C_0 = E_{p_k}(M \| CID_i \| y_i \| u) \quad (2)$$

Where $E_{p_k}(.)$ denotes the encryption function using the server's public key. "u" is the random value selected by the user during login time. To ensure the liveliness of the user, a nonce value is added in the value of $C_0$ along with the already existing random values to add more security. $C_0$ is sent to the server.

### Mutual authentication phase

Once $C_0$ is received by the server, the server does the following operations,

a. Server decrypts $C_0$ using its private key $s_k$
b. Server computes "$n_u^*$" such that $n_u^* = M \oplus K$ where $K = h(ID_i \parallel x)$. The server uses the $ID_i$ obtained from $y_i$.
c. The validity of the user is checked by using the Identifier $ID_i$ to the one received by the server. By using the value of $n_u^*$ the value of $CID_i^*$ is calculated.
d. Then the value of $CID_i^*$ is compared with the value of $CID_i$ to check if $CID_i = CID_i^*$.
e. Also the value of $ID_i$ can be verified with the ID stored in the ID table for the users at the server end. A comparison of ID's is done to make sure that verification is done correctly even when the Server ID table is corrupted.
f. The remaining terms of $C_0$ i.e., $(h\ (PW_i) \parallel SB_i)$ is retained for future reference.

Server computes a values of $C_1$ such that

$$C_1 = E_u(N \| SED \| S_{ID} \| v) \tag{3}$$

Where $S_{ID}$ = Server's identity and v is the random number chosen by the server and u is the random number selected by the user and sent in $C_0$. The server generates a nonce value "$n_s$" and computes $N = (K \oplus n_s)$. From the value of $n_s$, the value of the symmetric key u is generated. Server ID $SED = h(S_{ID} \| n_s)$. The dynamic ID and $n_s$ is used to make sure that the data was not tampered during transmission. Server sends $C_1$ to the user $U_i$.

In the User Side, the following operations are done,

a. The smart card decrypts $C_1$ using the random value of u.
b. The value of $S_{ID}$ is checked for valid server ID. The smart card computes $SED^* = h(S_{ID} || n_s^*)$ using its nonce value $n_s^*$. Smart Card computes "$n_s^*$" such that $n_s^* = N \oplus K$ where $K = h(ID_i \parallel x)$.
c. Then $SED^*$ is calculated by using the value of the generated $n_s^*$ and $ID_i$., i.e., $SED^* = h(S_{ID} || n_s^*)$. If $SED^* = SED$, then the server is valid and the data has not been tampered with.

The smart card calculates the following value

$$C_2 = E_v\ (h\ (PW))' \parallel SB_i{}' \tag{4}$$

The server decrypts $C_2$ using v and calculates the value of $y_i^*$ from the values sent in $C_2$. If $y_i^* = y_i$, the server matches the values of the password and the biometric template to confirm the authenticity of the user.

If an attacker is to attack, he/she has to deduce the random and the nonce values which makes the attack much difficult. The value of $SB_i^*$ in $C_2$ is compared with $SB_i$ of $y_i$. If the value match is within a threshold range then the user is confirmed valid. The three phases are considered for computing the cost since they will be used repeatedly. Once all the steps have been completed successfully, it is clear that mutual authentication of both the user and the server is done for login of the user. The server secret number v can be used as a session key material and h (v) can be used as a session key which is shared with the server.

### Password change phase

The user $U_i$ is authenticated by using the Password (PW') used initially for login process. Once authenticated, the user is prompted to enter the new password. Once the new password (PW") is entered, the $y_i = E_x (ID_i \parallel h(PW_i) \parallel SB_i)$ value of $h(PW_i)$ is replaced with the value of $h(PW_i")$. An intimation of the password change is given to the server and it replaces the old password for the user identifier with the new password. Thereby the user is allowed to further login by using the new password.

## Implementation of the proposed authentication scheme

In this section, the performance and functionality of the proposed authentication scheme is analysed and comparison has been made with the existing SSL based Authentication used in the OGSA framework of Globus Toolkit.

The biometric matching is not done mostly in the smart card in proposed scheme but rather in the remote server without losing the privacy of the biometric data. Any light-weight public-key cryptosystem can be used for the encryption and decryption process. The total time taken for the execution of the proposed algorithm is purely based on the crypto-algorithm selected for the process of encryption and decryption.

In our proposed Scheme, Advanced Encryption Standard (AES) based on block cipher is used. Also Rivest Cipher 4(RC4) algorithm which uses stream cipher can be used. RSA of 1024 bits [24] is used for the Asymmetric Encryption. The AES algorithm used here has a key length of 128 bits and RC4 algorithm of 128 bit key length can be used. The time taken for execution of the SSL based authentication in milliseconds (ms) is shown in Table 2. The algorithm is executed for an input of 10 users each of a 26 kB biometric finger print image. MD5 scheme has been used for hashing.

### Performance analysis of initial/registration phase

A simple Grid environment was created and the security algorithm was implemented for 10 users. A simple hosting environment has been created as presented by [25]. In a Microsoft .NET platform and J2EE application server as an administrative server, the hosting environment has been implemented for 10 connected users. The time taken for

**Table 2 Total time taken for registration and authentication**

| User | Time taken for each access-proposed scheme (ms) | Time taken for registration and authentication-SSL based scheme (ms) |
|---|---|---|
| **1** | 710.32 | 553 |
| **2** | 710.76 | 612 |
| **3** | 565.58 | 550 |
| **4** | 555.95 | 518 |
| **5** | 604.90 | 549 |
| **6** | 719.52 | 594 |
| **7** | 710.11 | 617 |
| **8** | 570.58 | 553 |
| **9** | 591.16 | 525 |
| **10** | 606.35 | 550 |

execution of the Initial/Registration phase and the login and mutual phases were calculated. The resistance of the security algorithm to attacks has been analysed in the next section through the equations. Based on the time factor criteria, the implementation of the proposed algorithm is based on the following system configuration of Processor Speed – 2.13 GHz, RAM size 3.00 GB, System Type – 32-bit OS. The implementation has been done in Java. The time taken for the Initial process and authentication of each user is shown in Table 2.

The Table 2 gives the time taken for each user for the initial registration and access in to the grid environment. The Figure 4 is the corresponding chart for Table 2. From the graph it is clear that the time taken for initial registration of a user using smart card is marginally more than the existing SSL based scheme. The Table 3 shows the time for each user login and authentication. It is the time taken for a single access into the grid network. The corresponding graph is shown in Figure 5. From the graph it is clear that the time for each access is very less when compared to the time taken for the SSL based authentication. Table 4 gives the total time taken for the users as they increase in entering into the grid environment. The Figure 6 gives the corresponding graph for the Table 4. From Table 4 it is clear that the time for the combined registration and access is more in the initial phase due to the collection of biometric data and the smart card distribution. The Figure 6 is the graph for Table 4.

In Table 5, the time taken for each access of user login is given. The Figure 7 is the corresponding graph for Table 5. The Figure 7 shows that the total time for each user login is very less when compared to that of the SSL based authentication scheme. It is clear from the collected data, that though the time for initial operation is more for from the Figures 5 and 7, it is clear that even though the registration phase of each user is more, the time taken for each access is much lesser than the time taken for execution of the SSL scheme. This increase in time during initial stages is very much compensated during each user access. It is clear from the data collected that the selection of the encryption algorithm used for encryption influences the time taken for completion of the execution of the process. Lightweight algorithms like Camellia [26] in place of AES algorithm and Elliptic Curve Cryptography in place of RSA algorithm can also be considered for usage.

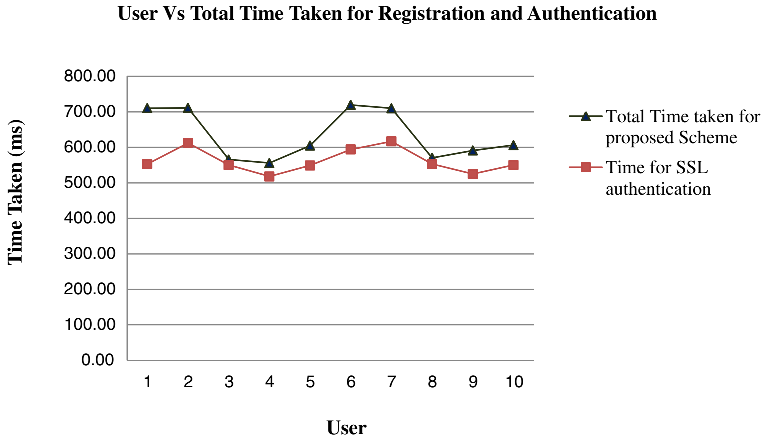

**Figure 4** User vs total time taken for registration and authentication.

**Table 3 Time taken for each grid access**

| User | Time taken for each access-proposed scheme (ms) | Time taken for each access-SSL based scheme (ms) |
|---|---|---|
| **1** | 70.77 | 553 |
| **2** | 71.33 | 612 |
| **3** | 72.28 | 550 |
| **4** | 68.64 | 518 |
| **5** | 70.75 | 549 |
| **6** | 72.92 | 594 |
| **7** | 68.82 | 617 |
| **8** | 82.65 | 553 |
| **9** | 72.56 | 525 |
| **10** | 70.38 | 550 |

The next section gives a brief discussion on the security analysis of the proposed authentication scheme.

## Security analysis of the proposed authentication scheme

In this section, the security and performance analysis of the proposed authentication scheme are presented. The attacks which are withstood by the proposed scheme of authentication are explained.

### ID-theft attack

As in equation $C_0 = E_{p_k}(M\|CID_i\|y_i\|u)$, a dynamic user ID named as $CID_i$ is created by the smart card based on the nonce value $n_u$ instead of using the user's own ID. This helps to withstand the ID-theft attack and also preserves the privacy of the user.

### Clock synchronization and replay attack problem

In [27], the problem in timestamp based authentication is given as replay attack due to the transmission delays in an unpredictable network. Even though the networks are fast

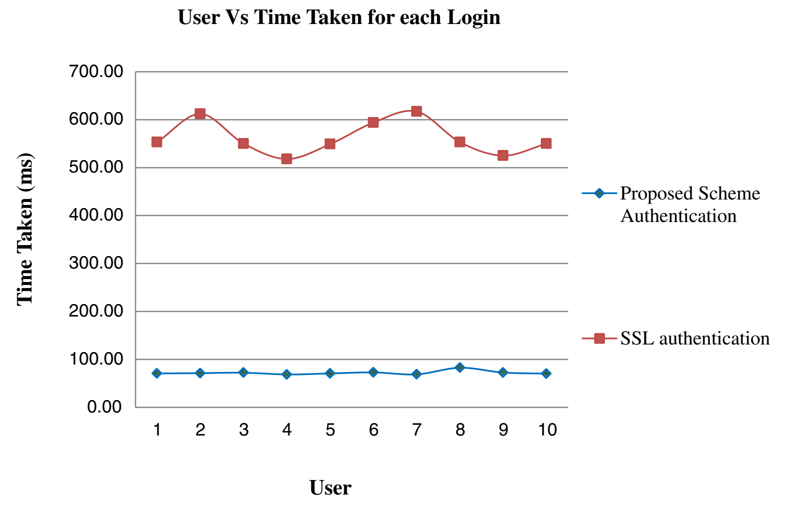

**Figure 5** User vs time for each login.

**Table 4 Initial time taken for SSL based and proposed scheme**

| No. of users | Total time for no. of users - SSL based (sec) | Total time for no. of users - proposed scheme (sec) |
|---|---|---|
| **1** | 0.55 | 0.64 |
| **2** | 1.17 | 1.56 |
| **3** | 1.72 | 2.20 |
| **4** | 2.23 | 2.82 |
| **5** | 2.78 | 3.49 |
| **6** | 3.38 | 4.29 |
| **7** | 3.99 | 5.08 |
| **8** | 4.55 | 5.71 |
| **9** | 5.07 | 6.40 |
| **10** | 5.62 | 7.08 |

the speed may vary based on the geographical and political distribution. To avoid using of timestamps, a nonce value $n_u$ is used each time the user sends his/her data and a nonce value $n_s$ is also used by the sever to proclaim the server's validity. Since a nonce value such as $n_u$ and $n_s$ in equations $C_0 = E_{p_k}(M \| CID_i \| y_i \| u)$ where $M = (K \oplus n_u)$ and $C_1 = E_u\ (N \| SED \| S_{ID} \| v \| pos_s)$ where $N = (K \oplus n_s)$ can be used only once, and not repeated, the user/server can be safeguard themselves from replay attacks.

### Modification attack

Each authentication message in from equation (1), (2), (3) and (4) include a one-way hash function along with an encryption algorithm. The hash value in each equation requires a nonce value or a random value. Even if the attacker gets hold of each of these equations the decryption part and breaking the hash function is not possible. If the attacker has the value of $h(PW_i)$, to find the password, the attacker needs find an equivalent of the hash function by trying each password. This attack is difficult because the attacker has to first break into the encrypted data $C_0 = E_{p_k}(M \| CID_i \| y_i \| u)$. The

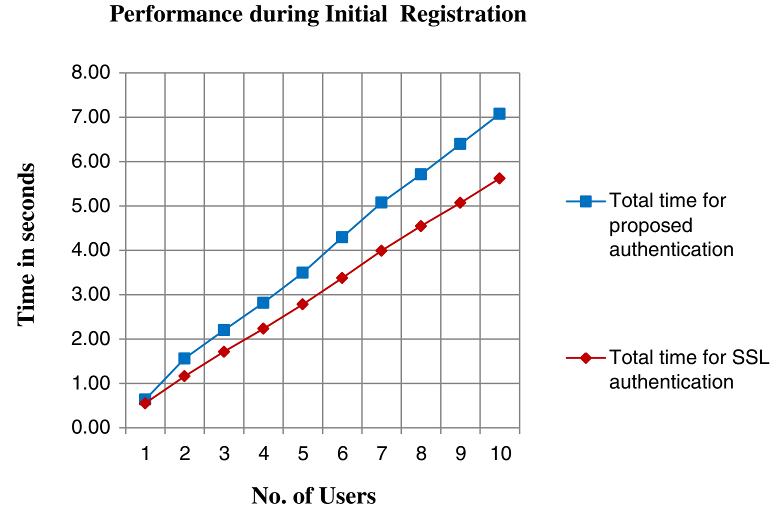

**Figure 6** Performance analysis in terms of total time taken initially.

**Table 5 Time for access of all the 10 users**

| Users | Total time for each access-proposed scheme (sec) | Total time for each access-SSL based (sec) |
|---|---|---|
| **1** | 0.07 | 0.55 |
| **2** | 0.14 | 1.17 |
| **3** | 0.21 | 1.72 |
| **4** | 0.28 | 2.23 |
| **5** | 0.35 | 2.78 |
| **6** | 0.43 | 3.38 |
| **7** | 0.50 | 3.99 |
| **8** | 0.58 | 4.55 |
| **9** | 0.65 | 5.07 |
| **10** | 0.72 | 5.62 |

attacker then needs to send the correct dynamic ID using the nonce. For an attacker to get all the values correct is impossible which makes modification attack difficult. Without knowing the actual data of these two values, the original data cannot be modified. Modification of the equations will be noted by the legitimate user and server and since all the messages are linked, it makes modification attack harder.

### Mutual authentication

At the end of the mutual authentication phase, both the server and the client authenticate each other thereby establishing mutual authentication. During each phase, of the equations $C_0$, $C_1$ and $C_3$, the user and server check the validity of each other using the values of CID, SED, M, N. If the server has any doubt in the validity of the user, the message $C_2$ can be asked to be resent.

### Man-in-the-middle attack

An attacker A who tries to do a man-in-the middle attack needs to know the decryption keys u, v and r in each message signal else its message will be discarded by the server or the client.

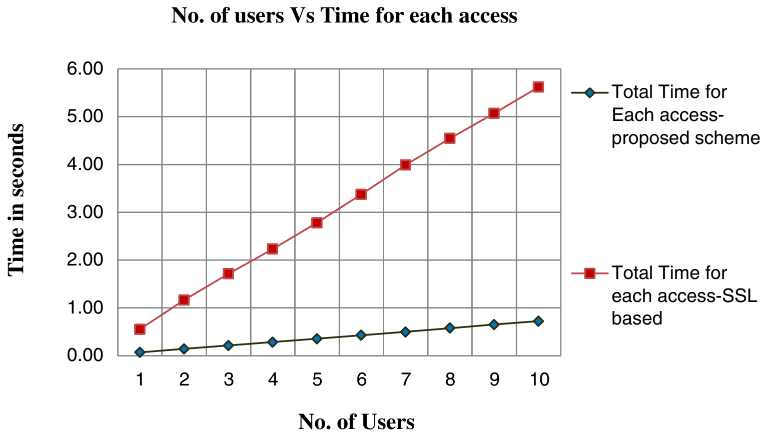

**Figure 7** Total time for user access.

### Security of the stored data on the smart card

The smart card holds the value of ($ID_i$, $y_i$, h(.), $p_k$) where, $y_i = E_x (ID_i \parallel h(PW_i) \parallel SB_i)$. If the smart card is compromised, the data it provides is not easily accessible to the attacker. Without knowing the matching password and the ID of the user, the attacker cannot move further along the authentication phase. Knowing the public key of the server complicates matters since the attacker has to find the encryption algorithm and a matching value of $C_0$ to send to the server. Furthermore, the hash function has to be broken in order to get the secret data. The biometric data is stored in the open for anyone to copy it. It is stored in the form of a template combined with a random string which needs to be found to get the data. Thus the data stored in the smart card is secure.

## Conclusion and future work

The proposed authentication scheme has provided an enhanced security with an optimal overall time taken for the operation. The authentication scheme can be made more secure by using a triple DES algorithm but it increases the security criteria and also increases the overall time taken for authentication. By increasing the security during the authentication phase itself we can try to minimize any other malicious insider attacks and also reduce external attacks. The increase in time during registration is one-time value and hence it is not considered as a disadvantage. The biometric data used for authentication can also be used in the consecutive authorization process thereby lessening the database space utilized by reusing the data used in authentication. The AES, RC4 algorithm can be replaced by any other light weight encryption algorithm like camellia. Further study has to be done by using different combination of algorithms. The data's used for authentication can also be used for authorizing the user for a resource access.

#### Competing interests

The authors declare that they have no competing interests.

#### Author's contributions

GJWK, analyzed the requirement of the security for Grid Computing, designed the framework, conducted the experiment and drafted the manuscript. EK gave full support in conducting the experiment and assisted in draft work and revised the manuscript. EBR contributed by reviewing the work done and in revising the content of the manuscript. All authors read and approved the final manuscript.

#### Acknowledgement

The authors wish to thank Karunya University for the support by providing infrastructure for carrying out the simulation. The authors thank the senior professors and the industrial experts for providing valuable suggestions to improve the quality of the research paper.

#### Author details

[1]Department of Information Technology, Karunya University, Coimbatore, Tamilnadu, India. [2]SSTP Systems, Bharat Heavy Electricals Limited, Trichy, Tamilnadu, India. [3]School of Computer Science and Technology, Karunya University, Coimbatore, Tamilnadu, India.

Received: 25 April 2012 Accepted: 13 September 2013
Published: 22 September 2013

#### References

1. Foster I (2002) A three point checklist. GridToday 1(6):1–4, July publication
2. Foster I, Kesselman C, Tuecke S (2001) The anatomy of the grid: enabling scalable virtual organizations. Int J High Perform Comput Appl 15(3):200–222
3. Zhou Q, Yang G, Shen J, Rong C (2005) A scalable security architecture for grid, Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies., pp 89–93
4. Bendahmane, Essaaidi M, El Moussaoui A, Younes A (2009) Grid computing security mechanisms: state-of-the-art, International Conference on Multimedia Computing and systems ICMS '09., pp 535–540
5. Von W (2005) Globus toolkit version 4 grid security infrastructure: a standards perspective., Available at: http://www.globus.org/toolkit/docs/4.0/security/GT4-GSI-Overview.pdf, Accessed: January 2011

6. Almond J, Snelling D (1999) UNICORE: uniform access to supercomputing as an element of electronic commerce. Future Generat Comput Syst 613:1–10
7. Andrew S, Grimshaw W, Wulf A (1997) The legion vision of a worldwide virtual computer. Commun ACM 40 (1):39–45
8. Buyya R, Venugopal S (2004) The gridbus toolkit for service oriented grid and utility computing: an overview and status report, Proceedings of the first IEEE International Workshop on Grid Economics and Business Models., pp 19–66, ISBN 0-7803-8525-X
9. Jaspher Willsie Kathrine G, Kirubakaran E (2011) Four-factor based privacy preserving biometric authentication and authorization scheme for enhancing grid security. Int J Comput Appl 30(5):13–20
10. Lamport L (1981) Password authentication with insecure communication. Comm ACM 24(11):770–772
11. Hwang T, Chen Y, Laih CS (1990) Non-interactive password authentication without password tables. IEEE Conference on Computer and Communication Systems 1:429–431
12. Lee JK, Ryu SR, Yoo KY (2002) Fingerprint-based remote user authentication scheme using smart cards. Electron Lett 38(12):554–555
13. Chang CC, Lin IC (2004) Remarks on fingerprint-based remote user authentication scheme using smart cards. ACM SIGOPS Operating System Rev 38(4):91–96
14. Lin CH, Lai YY (2004) A flexible biometrics remote user authentication scheme. Comput Stand Interfac 27(1):19–23
15. Uludag U, Pankanti S, Prabhakar S, Jain AK (2004) Biometric cryptosystems: issues and challenges. Proc IEEE Special Issue on Multimedia Security for Digital Rights Management 92(6):948–960
16. Dodis Y, Ostrovsky R, Reyzin L, Smith A (2004) Fuzzy extractors: how to generate strong keys from biometrics and other noisy data. Advances in cryptology-eurocrypt 2004. Lect Notes Comput Sci 3027:523–540
17. Juels A, Wattenberg M (1999) A fuzzy commitment scheme. In: Proceedings of the 6th ACM Conference on Computer and Communications Security., pp 28–36
18. Sutcu Y, Li Q, Memon N (2007) Protecting biometric templates with sketch: theory and practice. IEEE Transactions on Information Forensics and Security 2(3):503–512
19. Chen CM, Ku WC (2002) Stolen-verifier attack on two new strong-password authentication protocol. IEICE Transactions on Communications E85-B(11):2519–2521
20. Fan C-I, Lin Y-H (2009) Provably secure remote truly three-factor authentication scheme with privacy protection on biometrics. IEEE Transactions on Information Forensic and Security 4(4):933–945
21. Trammell DD (2008) Four-factor authentication., Available at: http://blog.dustintrammell.com/2008/11/21/four-factor-authentication/#more-160, Accessed: January 2011
22. Foster I et al The open grid services architecture, version 1.5., Available at: http://www.ogf.org/documents/GFD.80.pdf, 2006, Accessed: January 2011
23. Alfieria R et al (2005) From gridmap-file to VOMS: managing authorization in a grid environment. Futur Gener Comput Syst 21:549–558
24. Coffey N (2012) Comparison of ciphers., Available at: http://www.javamex.com/tutorials/cryptography/ciphers.shtml, Accessed: January 2012
25. Foster I, Kesselman C, Nick JM, Tuecke S (2002) Grid services for distributed system integration. Journal Computer 35(6):37–46
26. Moriai S, Kato A, Kanda M (2005) Addition of camellia cipher suites to transport layer security., Available at: http://tools.ietf.org/pdf/rfc4132.pdf, Accessed: January 2012
27. Gong L (1991) Security risk of depending on synchronized clocks. ACM Operating System Review 26(1):49–53

doi:10.1186/2192-1962-3-16
**Cite this article as:** Kathrine *et al.*: Smart card based time efficient authentication scheme for global grid computing. *Human-centric Computing and Information Sciences* 2013 **3**:16.