Integrated Database Security System Architectures based on WCF Services

[1]Ayman Mohamed Mostafa, [2]Mohamed Hashem Abdel Aziz, [1]Ibrahim Mahmoud El-henawy
Faculty of Computers and Informatics,
[1]Zagazig University, Egypt
[2]Ain Shams University, Egypt

Abstract

Database security is a set of mechanisms, rules, and procedures that can be used to ensure confidentiality, integrity, and availability of data to protect database from unintended activities. The main goal of this paper is to provide integrated database security system architectures based on Windows Communication Foundation (WCF) services using a set of effective database security policies. Applying WCF services to the system can enhance the performance between all clients and database server and this is considered as one of the major contributions in this paper. The diversity of security policies in this paper depends on controlling database administrators and users who may abuse their privileges to penetrate confidential data in the database server. The controlling process depends on applying WCF services to a set of novel algorithms to provide efficient, flexible, and harmonious secret sharing connections between database users, database administrators, and super administrator in order to prevent any hostile act.

Keywords: *Database security, Security policies, Windows communication foundation*

1. Introduction

The security mechanism in any organization depends mainly on network security, physical security, and database security. The network security is considered as the outer security layer for protecting the system from external intruders. The main backbone for securing the system network is by using intrusion detection systems (IDS), firewalls and by using secure socket layer (SSL) and transport layer security (TLS). Physical security provides a kind of access control measures to deny access to unauthorized users. It can be used also to distinguish between authorized and unauthorized users, detects intrusions, and triggers appropriate responses with respect to attacks.

All these security measures will be useless if attacks on the system were from inside. The inside intruders can misuse their legitimate privileges to attack confidential information inside database system. So, database security is considered as the last line of defense for securing database from external attackers and internal intruders.

The U.S. security magazine "SECURE CYBERSPACE" survey stated that, 89% of the users install a firewall, 60% of the users install intrusion detection systems, but there are still 90% of the user's system security has been damaged [2].

The main idea of this paper is to implement a database security system based on our multi-layer security policies architecture that has been presented in our paper [1]. This architecture has been implemented based on WCF services to provide declarative infrastructure, interactive and flexible security system that can be used in any recent organization.

The WCF services are modern Microsoft technology that have been developed in the last four years and are applied to many research areas but these services have not been used, explained, and implemented in the field of database security yet. Implementing WCF services in the field of database security and clarifying the use of these services in all levels of the implemented system are considered as one of the novel contributions on this paper.

The contribution of this can be summarized as follows:

- Providing security architecture for securing relational databases based on multi-layer security policies.
- Securing relational database that lies on the server side and monitor authorized users who may misuse their privileges on the client side.
- Implementing WCF services to the database security system to present harmonious connections between database users, database administrators (DBAs), and the super administrator who can be a trusted third party (TTP) in the system.
- Providing novel algorithms for having an efficient, flexible, and extensible service-oriented database security system based on WCF services.

The rest of the paper is organized as follows. Section 2 presents recent related works in the field of database security. Section 3 presents overview about WCF services, their components and recent papers which use WCF in their implementations. Section 4 presents a database security architecture based on multi-layer security policies. Section 5 presents WCF deployment to the security architecture. Section 6 explains WCF services infrastructure that has been

used in the security operation. Finally, we conclude the paper with some future works in section 7.

2. Related work

Database security applications depend on three main dimensions to provide a strong security system. These dimensions are known as: confidentiality, integrity, and availability. Data confidentiality refers to the protection of database from unauthorized disclosure and is enhanced by using different database cryptographic techniques. Data integrity refers to the prevention of unauthorized users from modifying data and is enhanced by using access control techniques. Data availability refers to the capability of data to be available on the web even if there is a system crash and is enhanced by machine learning techniques [3].

The integrity of database can be maintained by using access control mechanisms. One of the recent researches in access control is to build hierarchical access control on database [9]. It proposes automated design method to reduce the number of operations for user data spaces creation. Leon Pan [10] presents a novel criterion-based access control approach to deal with multilevel database security. In this approach, authorization rules are transformed to security criteria, security criterion expressions, and security criterion subsets. Shyni et al [11] provides a model for privacy protection in object relational database. The key feature of this model is to allow multiple purposes to be associated with each data element and also supports explicit prohibitions. Access control mechanisms can be conducted also to web databases in a model called RBAC+. The central idea of RBAC+ includes the concepts of application, application profile and sub-application session when controlling the access to web databases [4].

The confidentiality of database is enhanced by using different cryptographic techniques that are applied to data when it is stored on secondary storage or transmitted on a network. Recently, the use of encryption techniques has gained a lot of interest in the context of outsourced data management. One of the recent achievements in encrypting database is to provide mixed cryptography operation over database [5]. It proposes a framework in order to encrypt databases over un-trusted networks in a mixed form using many keys owned by different parties. The proposed framework is very useful in strengthening the protection of sensitive data even if the database server is attacked at multiple points from the inside or outside.

There are also different researches that have been targeted on implementing cryptography algorithms for securing database as presented in [6, 7, and 8]. There are different policies have been presented for securing database. The authors in [12] implement different policies based on different types of access control mechanisms. The authors in [13] provide a study on different types of access control such as discretionary, mandatory, and role-based access control. This paper provided advantages and disadvantages on these access control types on different applications.

All these researches focused only on securing database from external and internal users without providing a way for protecting confidentiality and integrity of database from database administrators. This paper presents a comprehensive integration between access control mechanisms, cryptography techniques and intrusion detection systems for providing multi-layer policies for protecting database. These multi-layer policies monitor any intrusive behavior inside database and make the intruder (user or database administrator) under

pressure at all levels of security policies. This pressure is made by each policy that presents a security point to protect database from a particular dimension but each single policy can be penetrated by any malicious database administrator or system user. This problem can be solved by protecting each single policy with the existence of another policy that provides a second layer of security and so on. The interconnected policies are implemented together in a harmonious system using WCF services to provide an effective database security system based on multi-layer policies.

3. WCF services

WCF provides a declarative infrastructure and extensibility for all forms of communication to and from the Windows platform. It provides tools and libraries for building web services (WS) that are secured via the mechanisms of WS-Security and related specifications. Using WCF, communications happen at designated service endpoints, and an endpoint can implement different protocols. An endpoint is simply a resource on the network to which messages can be sent. The extensible characteristics are what make WCF a strong distributed application platform [14, 17]. Each WCF service has three main parameters: address, binding, and contract.

Address defines where on the network messages should be sent so that the endpoint receives them. This is the location to which messages must be sent by the client. The binding defines the channel used to communicate with an endpoint. Channels are the conduit through which all messages pass within a WCF application. A channel is composed of a series of binding elements. The lowest level binding element is the transport, which delivers messages over the network. Contract defines the operations that an endpoint exposes and the message formats

that the operations require [18]. The built-in transports modes are presented in Table 1.

Table 1. WCF Built-in Transport Modes

Communication Mode	Usage
TCP	This mode is used for services that require fast communication on a local network.
HTTP	This mode is used for services that require web services standards.
Named Pipes	This mode is used for fast on-machine communication.
MSMQ	Queued communication on-machine and across the network. This communication mode requires MSMQ to be installed.
Peer to peer	Create a node in a peer to peer network in which each computer in the network can act as a client or server for the other computers in the network

One of the most recent researches based on WCF is to develop agent-based controls in complex service oriented architecture (SOA) [15]. Another research direction is to use WCF to realize vector data services [16] because there is a variety of vector data formats and vector data contains a large number of coordinates that require a network service to handle them. Applying WCF services to the field of database security is very rare [14]. So, one of our major contributions of this paper is to apply WCF services through a client-server application to provide efficient and flexible secret sharing operation between database administrators in a database security system.

4. DB security architecture

Unlike database security frameworks that exist today, which mostly detect imminent problems, generate an alert, and produce a report, this architecture provides different database security policies that protects database from internal intruders while keeping the database in a consistent state. As presented in our paper [1], the database security architecture depends on different security policies. Each policy presents a specific security point to protect database from a particular target. But each single policy can be penetrated by any malicious database administrator or system user.

This problem can be solved by protecting each single policy with the existence of another policy that provides a second layer of security and so on.

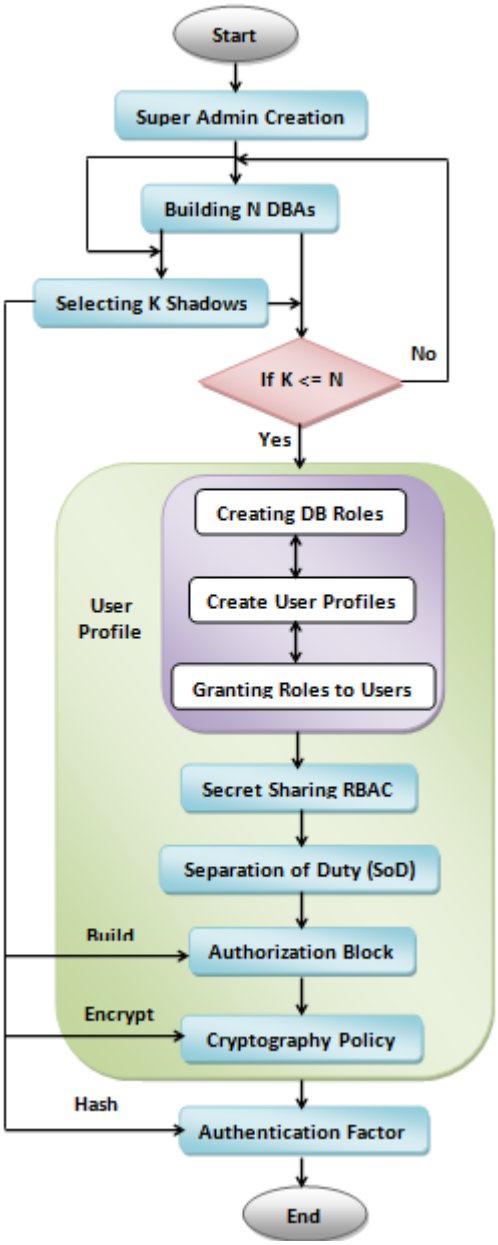

Figure 1. Security policies architecture

As presented in Figure.1, the super administrator (SA) is responsible for building N database administrators who can connect to the database. The super administrator also determines a number of K shadows which are the minimum number of database administrators (DBAs) who can manage and control any transactions through the database server.

A checking procedure is executed to determine if the number of K DBAs is more than the number of N DBAs, then a system error will rise to put the parameters again such that the number of K shadows must be less than or equal to the number of N DBAs.

After building N DBAs, Each database administrator can connect separately to the system and will have the ability to create database roles, create user profiles, determine user activity whether

an (active $<U_A>$ -intermediate $<U_{IM}>$ -inactive $<U_I>$ users on database, and grant roles to each created user.

Based on the classification of users, a secret sharing operation (secret sharing RBAC) that requires a number of users to grant the operation together is created. The users who belong to the same role must share together to grant or deny the request from the requester user. This policy will be very time consuming such that the database system may have large number of legitimate users. In addition, there are some operations that require fast actions and any delay will make the operation useless.

So, a supporting policy called SoD (Separation of Duty) is built based on the classification of database attributes. Database attributes can be classified as normal attributes (**An**), sensitive attributes (**As**), and most sensitive attributes (**Ams**). Normal attributes are part of database columns in which there will be no harm when there is a disclosure of data. Sensitive and most sensitive attributes are part of database columns in which there will be a great harm when there is a disclosure of data.

The operation of (secret sharing RBAC) will be executed based on sensitive and most sensitive attributes only in order to save large time in the secret sharing process.

Another policy for protecting sensitive and most sensitive attributes from any hostile act is by building authorization block which can be used to determine the maximum number of (insert – update – delete – select) operations granted for each user. Any user who exceeds the maximum number of operations stored in the authorization block is considered as malicious user. The intrusion response action on this intrusive behavior will be based on the classification of the user profile. Any active user (U_A) or intermediate user (U_{IM}) who is detected as intrusive user means that an aggressive action must be taken to automatically disconnect his connection from the application server. He will need to reconnect again but under the control of the super admin (**SA**). Any inactive user (U_I) who is detected as intrusive user will block or suspend his profile from database server until the super admin (SA) or a number of K DBAs deactivate the suspension again.

In order to protect the authorization block from any malicious action, it can be encrypted in order not to give the user the ability to change the policy itself by increasing his/her DML operations.

Another policy of authentication to provide a second layer of defense is created by executing the hash function (H) on the entire database schema using secret key sharing of K DBAs or the secret key of the super admin (SA). Any intrusion attempt from unauthorized user or single administrator will generate another schema that differs completely from the original one. The database security policies are explained in detail in our paper [1].

5. WCF service deployment

WCF is a unified programming technique for building service-oriented architecture (SOA) [15]. In order to deploy WCF services to the database security policies which have been presented in Figure.1, three-tier security architecture is built as presented in Figure.2. The architecture is presented vertically according to its components (Client-Application Server – Database Server) and is presented horizontally according to WCF deployment to (Trusted Third Party – Database Administrator – Legitimate User). The security architecture is presented as follows:-

5.1 Security architecture components

The multi-layer security policies explained in Figure.1 have been implemented in a database security system based on Three-Tier architecture. The vertical components in Figure.2 are: Client, Application Server, and DB Server. Each tier is presented as follows:-

5.1.1 Client tier. The client tier presents the graphical user interface of the database security system. The client side that connects to the database system can be the super administrator (SA) which is the trusted third party (TTP) inside database, database administrator (DBA), or legitimate user. Each one of them can connect to the database system from his private session.

5.1.2 Application server tier. The application server is considered as the intermediate tier between client and DB server. It has three main layers:

- WCF Service Layer.
- Business Layer.
- Data Access Layer (DAL).

Each layer will be discussed in detail in section 5.2

5.1.3 DB server tier. The DB server is considered as the repository for storing:

- Super administrator (SA) accounts.
- Database Administrators (DBAs) accounts.
- Legitimate user accounts.
- User profiles (active – intermediate – inactive).
- Database roles to be used by different users.
- User authorizations to be granted to each user.
- Secret key certificate for each (DBA).

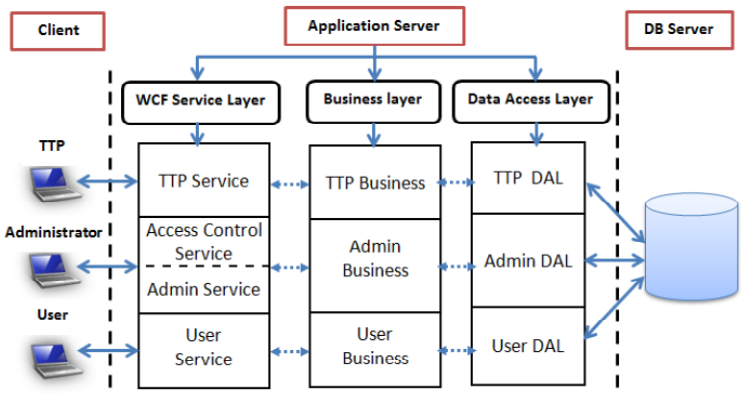

Figure 2. Three-tier security architecture

Encrypted secret key and password for each (DBA).

• Entire database entities, attributes, and records.

• All DML operations performed on database.

5.2. WCF service-oriented system mechanism

The mechanism of the proposed architecture depends on a harmonious interaction between the super administrator (Trusted third party) and database administrators. This will be illustrated in the following processes:

5.2.1 Trusted third party (TTP) processes. The Super administrator (SA) who is the trusted third party (TTP) in the database system is found on the top hierarchy of the system. He has all authorizations and capabilities to control and monitor database administrators, users, and transmissions inside the database system. He begins the system by performing the following operations:

1. Creating his account with a secret password and an additional secret key for more privacy.

2. Providing the maximum number of operations (insert-update-delete-select) allowed for each user in order to differentiate between active, intermediate and inactive user profiles. For example, the super administrator fills in the parameters for (Insert) operation as follows {Insert: Max (20), Active (15), Intermediate (8), Inactive (1)}. This means that any active user can perform between 15 and 20 insert operations. Intermediate users can execute insert operations >= 8 and < 15. Inactive users can execute insert operations >=1 and <8.

3. Determining the total number of database administrators (N DBAs) who can connect to the database system and the number of shadows (K DBAs) that must be found to provide a secret sharing process for every request from a single administrator. The maximum number of shadows (K DBAs) must be less than or equal to the maximum number of administrators connected to the system (N DBAs).

4. Providing the username, password, and a secret key certificate for each database administrator.

The secret key certificate will be generated as follows:

• If the database scheme contains N DBAs such that any K DBAs shadows can combine together to control the transmission process, the DBMS generates the following polynomial equation:

$$F(x) = (ax^{k-1} + bx^{k-2} + cx^{k-3} + \ldots.. + nx^{0}) \bmod p \quad (1)$$

Where (a, b, c,…, n) are random coefficients and

p is a prime number. These parameters are inserted into the system by the super admin (SA). Any K shadows can be used to create K equations by evaluating the polynomial at n different points such that:

$$k_i = F(x_i)$$

These secret keys must be kept private in a smart card such that any K DBAs out of N DBAs can make a decision for any policy modification. The implemented system encrypts both the password concatenated with the secret key certificate for each database administrator using the AES algorithm E (password || secret key certificate). Even if two secret keys for two different database administrators are matched and of course the two passwords for the two database administrators are different, then the final encryption result using AES algorithm will be different. The super administrator (SA) operations are sent as a request to the application server in a form of a TTP WCF Service. The WCF service is considered as an intermediate layer between client and the application server to pass all operation

requests. The TTP WCF services have two main objectives during the transmission of data:

− To pass all requests from the client side to the database server across the application server whether the client side was a super administrator, database administrator, or legitimate users.

− To return the respond from the database server back to the client side again for confirmation.

In order to perform the request, a set of binding configurations must be set first to prepare the connection between the client (requester) and the database server. The binding defines the channels that can be used to communicate between clients and database server.

The developed application was implemented using HTTP communication mode as explained in Table.1 because it is based on web services in a client-server system.

In this stage, a one-way communication pattern is used into which the requested messages are sent in one direction, from the client to the database server. One-way communication is common when the sender does not need an informational response back; the sender just needs an acknowledgement that the message was sent.

The developed system uses "BasicHttpBinding" class to perform the one-way communication pattern. The configurations of the "BasicHttpBinding" class are presented in Table 2.

Table 3. BasicHttpBinding configurations

Parameter	Description
BindingName	The name of the communicated channel.
OpenTimeOut	The time to open the connection.
CloseTimeOut	The time to close the connection.
ReceiveTimeOut	The time to receive the data.
SendTimeOut	The time to send data.
MaxBufferPoolSize	The maximum amount of data to be sent.
MaxReceivePoolSize	The maximum amount of data to be received.

As presented in Figure. 2, the TTP WCF Service passes the request to the TTP Business Layer. The main objectives of TTP Business layer are as follows:

- Perform any mathematical operations that the trusted third party (TTP) may execute inside database.
- Perform the mapping process because the trusted third party (TTP) can't send a request in a form of user entity, so the TTP business layer is used to map and convert the user entity into entity attributes.
- Perform security operations on the transmitted data by encrypting the passwords of all connected users whether they are trusted third party (TTP), database administrator, or system users.

After performing the previous operations, the TTP Business Layer passes the request to the last layer in the application server which is the TTP Data Access Layer (DAL).

The data access layer main objective is to create SQL data parameters and store the transmitted data that the trusted third party (TTP) has requested. The WCF service returns the respond to the trusted third party (TTP) as a confirmation process that all operations are carried out or none are.

5.2.2 Database administrator (DBA) processes. The database administrator is the second level of the security hierarchy. Once the trusted third party (TTP) stores all database administrators' accounts, each database administrator can now access the security system separately by providing the system with his authentication parameters (username, password, and secret key certificate). Once the database administrator (DBA) proceeds with the system he can perform the following operations:

• Adding users

The database administrator (DBA) first operation is to add all users that can connect to the database. The database administrator (DBA) must provide three categories of information for each created user:

− Login Information such as (Username and Password).

− Related Information such as (SSN, email, and address).

− System Information that determines the user profile. This policy determines whether the user can be (Active, Intermediate, or Inactive user).

• Adding database roles

The second operation of the database administrator (DBA) is to create database roles to be used in the security system as presented in [1]. This policy creates each role and determines which DML operations (Insert – Update – Delete – Select) to be granted to each role.

• Adding user roles

The third operation of the database administrator (DBA) is to grant database roles to the legitimate users.

• Adding user authorizations

The last process for database administrator (DBA) has two main stages:

- Determine the authorizations for each user and determine whether there are any sensitive attributes that must be hidden from the user.

- Fill in the actual parameters to be granted to each user in the authorization block to determine the number of (Insert - Update – Delete – Select) operations allowed for each user based on the classification parameters that have been inserted by the super administrator.

The main important point here is to distinguish between the main DML parameters stored by the super administrator (SA) and the actual DML parameters to be granted to each user by the database administrator (DBA). The database administrator (DBA) must take into his considerations the main DML parameters created by the super administrator (SA) as presented in section 5.2.1. For example, if the database administrator (DBA) passes the parameter for an active user (Max Insert = 13). This value will be checked by the TTP Business Layer to determine its validity to be saved in the database server. The database server will raise an error because the inserted parameter violates with the stored parameter which must be between 15 and 20 insert operations.

All database administrator operations (adding users, adding roles, granting roles, and user authorizations) are sent as a WCF service request from the client (requester admin) to the database server.

In this stage, a two-way communication pattern (Duplex) is needed because each requester (DBA) must open a secret sharing connection to all connected database administrators to handle the request. Then each connected database administrator must respond to the request (Callback) to grant or deny this request. The developed system uses "wsDualHttpBinding" class to perform the two-way communication pattern in order to send and receive messages. Before establishing the two-way communication pattern between the client and the server, the WCF configurations must be prepared. This is explained in Algorithm 1.

Algorithm 1 Host Server Behavior

1. / / Set up the isolation principle
2.<Service Behavior> (InstanceContextMode = InstanceContextMode. Single)
3. / / Set up the concurrency principle
4.<Service Behavior> (ConcurrencyMode = ConcurrencyMode.Multiple)

As presented in Algorithm 1, the host server WCF service behavior has two main parameters. The first parameter which is the instance context mode is set to "Single", this means that all database administrators who have a secret sharing on a specified operation must pass their requests and responses on this operation in a single line such that no intervention of any other operation will be found.

This is called the principle of "Isolation". The second parameter which is concurrency mode is set to "Multiple", this means that several database administrators can send requests or set permissions at the same time. This is the principle of "Concurrency". When the database administrator connects to the system with his authentication parameters (username, password, and secret key certificate), a log in dictionary must be created to save all connected admins along with a connection channel (callback) into which the database server will respond to each admin in his own channel. This is presented in Algorithm 2.

Algorithm 2 Log In Dictionary

1. // Starting a new system
2. If (callback != null) then // check whether there are
any old connections
3. Lock (connectedAdmins) // lock all admins in this
connection
4. {
5. // Search for the user name of each old admin
6. If (connectedAdmins.ContainsKey(username)) then
7. // Remove old admin user name from the dictionary
8. Remove(username)
9. End if
10. // Add any new connected admin with his user
name and connection channel
11.Add(username, callback)
12. }
13. End if

6. WCF service-oriented infrastructure

As presented in Figure 2, the database administrator performs two WCF services in the whole process starting from requesting an operation and ending to execute the operation or not. The two WCF services are: Access control WCF service and admin WCF service.

6.1 Access control WCF service

The first operation of Access Control WCF service is to count the maximum number of K DBAs shadows who can combine together in a secret sharing process as presented in Algorithm 3. The process of shadows count identification is a central process by which the request from any database administrator will be granted or denied. For example, if the number of K DBAs shadows is 5, then there must be 5 agreements to execute the process. When a secret sharing process is established and the number

of granted DBAs shadows is less than 5, then the process is denied.

Algorithm 3 Shadows Identification

1. If (systemShadowsCount = 0) then
2. // Retrieve the number of shadows from the new system
3. GetShadowsCount(systemID)
4. End if
5. Return systemID

All operations that the database administrator, say (DBA1) has just created whether adding user, adding DB Roles, adding user roles, or adding user authorizations are implemented as a request. This request is sent to all database administrators who are created by the super administrator (trusted third party) and are connected to the system (N DBAs). This request is sent from the client side using Access Control WCF service to the database server as presented in Algorithm 4. Algorithm 4 builds a "library table" which stores the username of each database administrator and the number of K DBAs shadows who must grant the request to him. Once the request is sent from the requester DBA, say (DBA1), the number of granted shadows are initialized with zero. Each granted DBA increments the library table with 1, until the K DBAs shadows are satisfied. The Access Control WCF service performs the broadcasting process to all connected database administrators from (DBA2) to (DBAn) to provide a secret sharing operation as presented in Algorithm 5.

The broadcasting process is executed by building a queue for all database administrators (N DBAs) who will receive a message to grant or deny the request. As explained before, at least K DBAs from the total N DBAs must approve the operation to grant the request.

Broadcasting is a kind of routing algorithm used in computer networks which makes sure that every device in the network will receive a (broadcasted) packet. Routing is the process of choosing which paths to be used to send network traffic, and sending the packets along the selected sub-network. Another routing algorithm is called flooding. Flooding is a very simple routing algorithm which sends all incoming packets through every outgoing edge [19, 20].

This paper performs the WCF services using integration between flooding and broadcasting algorithms for many reasons. First, in flooding algorithm packets are sent through every outgoing link. As a result, the bandwidth is obviously wasted. Second, due to the wasted bandwidth, flooding can actually degrade the reliability of a computer network. Third, broadcasting sends a packet along a link at most once. Fourth, unlike flooding, broadcasting is done by specifying a special broadcast address on packets.

Algorithm 5 performs hybrid integration between broadcasting and flooding algorithms into which the request is sent to all database administrators (N DBAS) across the network except the requester administrator (Flooding Algorithm) and the request is sent all DBAs only once because the message is sent to the username which is unique in the system (Broadcasting Algorithm). This is presented in Algorithm 5.

After broadcasting the request to all connected database administrators (N DBAs), each database administrator responds with a Boolean variable to the request from his own connected session and provides a response whether to grant or deny the request. This is presented in Algorithm 6. In Algorithm 6, each granted DBA increments the library table with 1, until the K DBAs shadows are satisfied.

All responses from all connected database administrators are sent again to the Access Control WCF service to perform the checking process. If the number of permissions from database administrators equal to the number of permitted shadows (Algorithm 3), the request is granted, otherwise the request is denied. This is called the reply operation in Algorithm 7.

The overall operations of access control WCF service-oriented system are presented in Figure.3 starting from sending the request and ending with replying operation to grant or deny the request.

Algorithm 4 Sending Request (Requester Client→Server)

1. // The request contains the user name of requester and action message
2. // Build a request library table that contains the user name of requester plus the number of DBAs shadows permitted to him.
3. If (requestsLib.ContainsKey(username) then
4. Remove (username) // Delete all old user names from library table
5. End if
6. // Add the requester user and start initialization with Zero
7. Add (username, 0)
8. // Broadcast the request message to all connected DBAs shadows
9. Broadcasting Request (username , Message)

Algorithm 5 Broadcasting (Server→All DBAs Shadows)

1. // Build a queue for all DBAs Shadows for broadcasting to them
2. // Broadcast to all DBAs shadows except the requester client
3. If (client.Key != username) then
4. // Broadcast username and message of requester to all DBAs Shadows

5. Client.Broadcasting (username, message)
6. End if

Algorithm 6 Setting Permission (All DBAs Shadows→Server)

1. If (grant = Yes) then
2. // Increment the number of shadows permitted to the requester by 1 in the request library table //
Algorithm 4
3. Requestslib [username] ++
4. End if
5. // Check if the total number of permitted admin equal to the number of shadows in the system //
Algorithm 3
6. If Requestslib [username] = systemShadowsCount then
7. // Reply a notification message to the requester
8. Reply (username, True)
9. End if

Algorithm 7 Reply (Server →Requester Client)

1. // The server scans for the requester client
2. // Send a notification message for granting the operation
3. Username. Reply (Permitted)

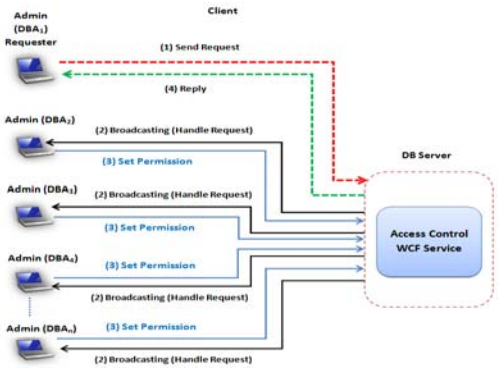

Figure 3. N-tier access control service architecture

As presented in Figure. 3, there are N-tier access control services that are sent to N DBAs. The Access Control WCF service is initialized when the requester administrator, say (DBA1) sends the request to all database administrators in the system. The request is sent to the Access Control WCF service in the DB server. The DB server handles the request and the Access Control WCF service performs the broadcasting process by sending the request to all N DBAs except the requester in order to avoid the return of the message to the requester again.

Each administrator from N DBAs responds to the request by verifying whether to grant or deny the request. Algorithm 6 counts the number of granted DBAs until at least K DBAs shadows are satisfied. At this point, a reply message is sent back to the requester administrator from the Access Control WCF service to grant the request. If the number of K DBAs shadows is not satisfied the request will be denied.

After completing all operations, each database administrator can close his connection and log off from the system. Algorithm 8 presents a log off dictionary to remove all database administrators from the system and reset the number of shadows for each operation in order to be ready to start any new connection again.

As presented in Algorithm 8, when any database administrator logs off from the system, the system searches for his username and removes it from the library table that has been presented in Algorithm 4. The system also resets the number of K DBAs shadows to zero to be ready to start any operation again.

Algorithm 8 Log off Dictionary

1. Lock (connectedAdmins) // lock all admins in this
connection
2. {
3. // Search for the user name of each admin
4. If (ConnectedAdmins.ContainsKey(userName)) then
5. Remove old admin user name from the dictionary
6. ConnectedAdmins.Remove(userName)
7. End if
8. }
9. // Check for the number of connected admins in the
system
10. If (ConnectedAdmins.Count = 0) then
11. // Reset the number of shadows
12. SystemShadowsCount = 0
13. End if

6.2 Admin WCF service

As presented in Figure.2, the database administrator performs two WCF services in the whole process. The first WCF operation is Access Control WCF service which is responsible for sending and receiving requests from the N DBAs to grant or deny the request as explained in the previous section.

If the request is granted to the requester administrator, say (DBA1), the Access Control WCF service returns the response back to the database administrator (DBA1) to continue and proceed in the transmission process.

As presented in Figure.2, the Admin WCF service is used to accomplish this transmission by sending the granted request again to the Admin Business Layer to perform any required mathematical operations or DML operations. The

Admin Business Layer passes the request to the Data Access layer to store the requested data in the database server. Finally, the operations are successfully completed and are stored in the database server.

7. Conclusions and future work

Database security mechanisms and techniques remain important goals in any data management systems whether these systems are commercial, industrial, educational, medical, or even military systems. The main goal of this paper is to build a set of security policies to provide a strong database system and provide a harmony between all participants in the system whether were super administrator, database administrators, or legitimate users. This harmony was built based on WCF service-oriented system to provide a secret sharing operation to grant or deny any request. We plan to extend our work on the following lines. First, the security system must be applied to different environments to determine the efficiency of the system and determine whether there is a need to add any new security policies or modify existing security policies to be applicable to any changing environment. Second, the number of tested database administrators must increase to determine the efficiency of WCF services in large organizations such as military institutions.

References

[1] M. Hashem, I. Henawy, A. Mostafa, "Interactive Multi-Layer Policies for Securing Relational Databases", Proceeding in the IEEE International Conference on Information Society, London, UK, 2012, pp. 65-70.

[2] J. Yu, P. Yan, X. Zheng, "Database Encryption and Confirmation Mechanism Research", Proceeding in the IEEE International Conference on Multimedia Technology, 2010, pp.1-4.

[3] E. Bertino, R. Sandhu, Database Security-Concepts, Approaches, and Challenges, IEEE Transactions on Dependable and Secure Computing, vol.2, no.1, 2005, pp.2-19.

[4] A. Bouchahda, N. Thanh, A. Bouchahda, F. Labbene, "Enforcing Access Control to Web Databases", Proceeding in 10th IEEE International Conference on Computer and Information Technology (CIT), 2010, pp.612-619.

[5] H. Kadehm, T. Amagasa, H. Kitagawa, "A Novel Framework for Database Security based on Mixed Cryptography", Proceeding in the IEEE 4th International Conference on Internet and Web Applications and Services, Japan, 2009, pp.163-170.

[6] J. Hou, "Research on Database Security of ECommerce Based on Hybrid Encryption", Proceeding in the International Symposium on Computational Web Information Systems and Applications, China, 2009, pp.363-366.

[7] W. Xing-hui, M. Xiu-jun, "Research of the Database Encryption Technique Based on Hybrid Cryptography", Proceeding in the IEEE International Symposium on Computational Intelligence and Design, China, 2010, pp.68-71.

[8] K. Kaur, K. Dhindsa, G. Singh, "Numeric to Numeric Encryption of Databases: Using 3Kdec Algorithm", Proceeding in the IEEE International Conference on Advanced Computing, India, 2009, pp.1501-1505.

[9] S. Antoshchuk, A. Blazhko, E. Saoud, "Automated Design Method of Hierarchical Access Control in Database", Proceeding in the IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems, Italy, 2009, pp.360-363.

[10] L. Pan, "Using Criterion-Based Access Control for Multilevel Database Security," Proceeding in the IEEE International Symposium on Electronic Commerce and Security, 2008, pp.518-522.

[11] C. Shyni, S. Swamynathan, "Purpose Based Access Control for Privacy Protection in Object Relational Database Systems", Proceeding in the IEEE international conference on Data Storage and Data Engineering, 2010, pp.90-94.

[12] Z. Rashid, A. Basit, Z. Anwar, "TRDBAC: Temporal Reflective Database Access Control", Proceeding in the IEEE 6th International Conference on Emerging Technologies (ICET), 2010, pp.337-342.

[13] B. Qing-hai, Z. Ying, "Study on the Access Control Model", Proceeding in the IEEE International Conference Cross Strait Quad-Regional Radio Science and Wireless Technology, 2011, pp.830-834.

[14] W. Zhang, J. Li, "Research and Application of WCF Extensibility", Proceeding in the IEEE International Conference on Information Systems and Mining (WISM), 2010, pp.363-367.

[15] S. Kamran, O. Haas, "Semantic agent-based controls for Service Oriented Architecture (SOA) enabled Intelligent Transportation Systems (ITS)", Proceeding in the IEEE International Conference on Intelligent Transportation Systems (ITSC), 2011 , pp.937-942.

[16] C. Liu, B. Zhang, L. Guo, W. Zhang, "Realization of Vector Data Service based on Windows Communication Foundation", Proceeding in the IEEE International Conference on Geo-Informatics, 2011.
pp.1-5

[17] K. Bhargavan, C. Fournet, A. D.Gordon, S. Tse, "Verified Interoperable Implementations of Security Protocols", Proceeding in ACM Transactions on Programming Languages and Systems, vol. 31, no. 1, 2008.

[18]W. Zhang, "A Service-Oriented Distributed Framework-WCF", Proceeding in the IEEE International Conference Web Information Systems and Mining, 2009, pp.302-305.

[19] F. Ferrari, M. Zimmerling, L. Thiele, O. Saukh, "Efficient Network Flooding and Time Synchronization with Glossy", Proceeding in the ACM International Conference on Information Processing in Sensor Networks (IPSN), 2011, pp. 73-84.

[20] W. Lou, J. Wu, "On Reducing Broadcast Redundancy in Ad Hoc Wireless Networks", Proceeding in IEEE Transactions on Mobile Computing, vol.1, no.2, 2002, pp.111-122.

