INSTITUTE FOR DEFENSE ANALYSES

Database Access – Application-Driven versus Data-Driven

William R. Simpson

28 April 2015

IDA Non-Standard
D-5403
Log: H15-000019
Copy

INSTITUTE FOR DEFENSE ANALYSES
4850 Mark Center Drive
Alexandria, Virginia 22311-1882

The Institute for Defense Analyses is a non-profit corporation that operates three federally funded research and development centers to provide objective analyses of national security issues, particularly those requiring scientific and technical expertise, and conduct related research on other national challenges.

About This Publication

This work was conducted by the Institute for Defense Analyses (IDA) under contract HQ0034-14-D-0001, Task BC-5-2283, "Architecture, Design of Services for Air Force Wide Distributed Systems," for USAF HQ USAF SAF/CIO A6. The views, opinions, and findings should not be construed as representing the official position of either the Department of Defense or the sponsoring organization.

Database Access – Application-Driven versus Data-Driven[1]

William R. Simpson
Institute for Defense Analyses, 4850 Mark Center Dr.
Alexandria, Virginia 22311

[1] The publication of this paper does not indicate endorsement by the Department of Defense or IDA, nor should the contents be construed as reflecting the official position of these organizations.

ABSTRACT

A multitude of commercial applications rely on Database Management Systems (DBMS) that provide an organized collection of data; for example, modelling the availability of flights and seating in a way that supports reservation and sales of air transportation. DBMSs are specially designed software applications that interact with other applications and users to capture and analyze data. A general-purpose DBMS is a software system designed to allow the definition, creation, querying, update, and administration of databases. For the purposes of this paper we assume that the database is front-ended by web services for database access and query. This paper discusses the current approach to database access and privilege by web services and the changes that are required by a high assurance end-to-end approach. The latter rely on a well-formed security paradigm for the enterprise.

Keywords: Database Access, DBMS, Access Control, IT Security, Integrity.

1. INTRODUCTION

Database security deals with all aspects of protecting the database content, its users, and its owners. It covers protection from intentional and unintentional unauthorized database activities by authorized privilege limited entities and unauthorized entities (e.g., a person or a computer program). Database access control deals with controlling who (a person or a computer program) is allowed to access what information in the database and what privilege is provided. The information may comprise specific database objects (e.g., record types, specific records, data structures), certain computations over certain objects (e.g., query types, or specific queries), or use of specific access paths to the former (e.g., using specific indexes or other data structures to access information). [1-9]

This may be managed directly on an individual basis, or by the assignment of individuals and privileges to roles that are then granted entitlements.

Data security prevents unauthorized users from viewing or updating the database. For example, an employee database can contain all the data about an individual employee, but one group of users may be authorized to view only payroll data, while others are allowed access to only work history and medical data.

Data security in general deals with protecting specific data, from corruption, destruction, or removal.

Our basic security model requires that all functionality be realized by web services. This precludes database grazing in which the requester can peruse most of the database at once. This is to be preceded by public key infrastructure (PKI)-based mutual authentication and a transport layer security (TLS) pipeline followed by a security assertion markup language (SAML) token for access and privilege. The database is organized by columns and each identity or role has permission that allow Create, Read, Update or Delete (CRUD) functions. However that still leaves two paradigms for database operations (application-driven and data-driven). To illustrate the difference an example is provided whereby a financial database is accessed by an individual who has credentials of a Financial Analyst.

2. AN EXAMPLE – ROLE-BASED ACCESS

The enterprise financial database (EFD) has many predefined roles. These are determined by the data owner, and placed in the format of an Access Control requirement (ACR) for storage in the enterprise service registry. The roles may be arbitrarily complex since the claims engine will compute whether or not they are satisfied and provide any variables or restrictions requested. A few are defined below:

1. Financial analyst is determined by position, training, and job identifier.

Financial Analyst =>
   a. manager and above, AND
   b. job identifier=xxx12, AND
   c. training=[basic finance (within last 5 years) AND financial Analysis (within last 5 years)] OR [BS, accounting or finance (within last 10 years)] OR waiver.

   RESTRICT
   a. sub area q unless supervisor is corporate director or above.
   b. data restricted to current location code. AND
   c. cannot update any project over $5M UNLESS a waiver is issued for the individual AND
   d. Additional restrictions may be included.

2. Financial Supervisor is determined by position, training, and job identifier.

Financial Supervisor =>
   a. manager and above, AND
   b. job identifier=xxx14, AND
   c. training=[basic finance (within last 5 years) AND Financial Analysis (within last 5 years)] OR [BS, Accounting or finance (within last 10 years)] OR waiver is issued for the individual.

RESTRICT
   a. cannot update any project over $5M until he has been using the system 6 months, OR
   b. waiver is issued for the individual.

2. Financial Auditor is determined by training and job identifier.

Financial Auditor =>
   a. job identifier=xx316, AND
   b. training=[basic finance (within last 5 years) AND Financial Analysis (within last 3 years) AND Financial Audit (within last 3 years)) OR (MS, Accounting or finance (within last 15 years)) OR waiver.

RESTRICT
   a. data restricted to audit location code.
   b. …

4. Bookkeeper …
5. Quality Control Specialist …
6. Administrator …

7 ….

3. APPLICATION-DRIVEN ACCESS

Each of the roles must be coded for operations. For illustration we will deal only with the Financial Analyst in this example - who, in this case, is Fred2345432, or just Fred. The evolution is the normal preparation for the access and privilege associated with an application or service. The figures below show the evolution of the access control, which involve most of the services in the enterprise attribute ecosystem.

The process begins with the generation of a SAML token.

Table 1 Basic SAML for Database Operations

SAML: Assertion		
Version ID	Version 2.0	Required
ID	SAML ID	Required
Issue Instant	Timestamp	Required
Issuer	(content)	Required
Signature	(content)	Required
Subject	User	Required - X.509 Identity
SAML: Attribute Statement		
Subject	User	For local use
Claims include Roles: and restrictions	(content)	may include parameters.
SAML: Conditions		
NotBefore	(content)	Timestamp
NotAfter	(content)	Timestamp
Audience	(content)	Target Service

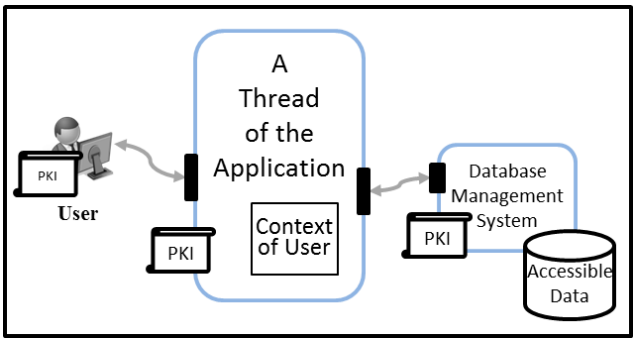


Figure 1 Application-Driven Access

The application (through the use of SAML) has the security context of the user. The application has full privilege with the database and is trusted to limit the user to his/her security context.

Application-Driven Annotated Example
Fred is the Chicago Branch Manager. The definitions of the various roles can be used to compute Fred's claims. Fred is evaluated based upon the enterprise data and he is provided a claim of Financial Analyst but with some restrictions as shown in the table below.

Table 2 Basic Data Evaluation for Fred

Financial Analyst =>	Fred	Claims Engine for Fred
manager and above, AND	Chicago Branch Manger	True
job identifier = xxx12, AND	Job code =43212	True
training=[basic finance (within last 5 years) AND (Financial Analysis (within last 5 years)] OR [BS, Accounting or finance (within last 10 years)] OR waiver.)	Training = training on basic AND finance(8/4/2012), AND (Financial analysis (6/5/2010), BS Mathematics Purdue (6/1/2000),) OR On the enterprise Training Waiver list group for Financial Analysts (TWFIN)	True AND True AND (False or True) =True Overall Fred = Financial Analyst
RESTRICTIONS		
sub area q unless supervisor is corporate director or above.	supervisor (all billets report to 43200 or 43201) is Field Office Manager	False Supply Notq token to application
Data restricted to location code. AND	Location Code = Chicago	Supply Chicago token to application
Cannot update any project over $5M UNLESS a waiver is present in the enterprise stores	Not in enterprise group for ($5Mupdatewaiver)	False supply Not$5M+ token to application

Table 3 SAML for Fred (Application-Driven)

SAML: Assertion	
Version ID	Version 2.0
ID	X34.?thik045ml23
Issue Instant	12:11:00 06 May 2014
Issuer	www.securitytokenserver3.net
Signature	(content)
Subject	Fred- X.509 Identity
SAML: Attribute Statement	
Subject	Fred2345432
Claims: Role = Financial Analyst Data= Notq, Chicago, Not$5M+	
SAML: Conditions	
NotBefore	12:11:00 06 May 2014
NotAfter	12:16:00 06 May 2014
Audience	www.mysqldata2.net

For databases, the application-driven approach has the following advantages and disadvantages:

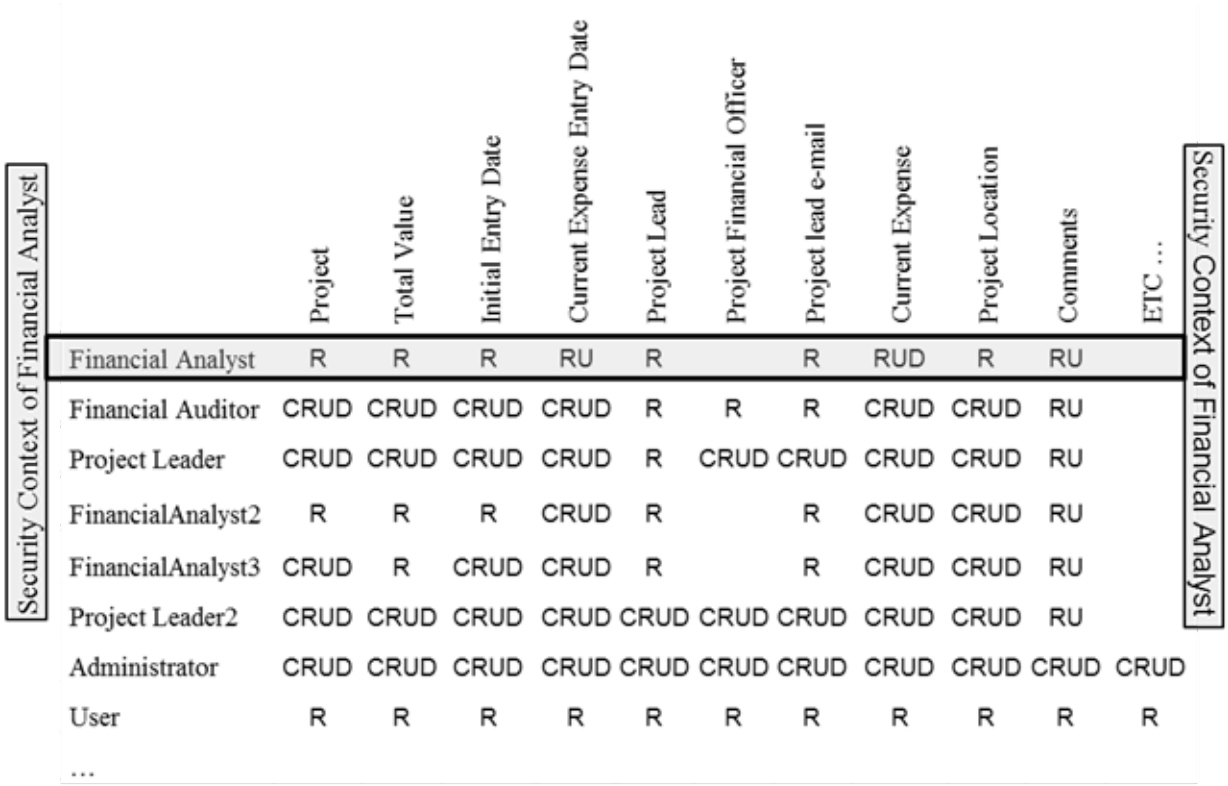
Advantages:
1. The data owner does not have to know the database schema in order to specify access and privilege.
2. The service controls Fred's interaction with the database.
3. Database administrators may or may not establish CRUDs for the role in question.

Disadvantages:
1. The service developer must know the database specifics
2. The service is granted full access to the database (to accommodate the different users.
3. The service computes what is allowable (CRUD) and send computed SQL for what it believes are reasonable requests consistent with Fred's authorities.

4. DATA-DRIVEN OPERATIONS

A number of additional are requirements needed for data-driven applications:

1. Database schema must be known to the developer of the access control requirements. Assume column authorization defined CRUDs
2. Elements in the database (when they represent the same thing in the enterprise attribute store (EAS) must be identical (and common definition) to the elements in the EAS. (Example: Location code in the database is a three character code. It must be the same code in the EAS – when multiple databases use the same value, they must all have the same representation as the EAS).
3. The database must be prepared: The column Create, Read, Update and Delete [CRUD] permissions are set in the database for each role (Figure 2).
4. CRUD by role:



Security Context of Financial Analyst	Project	Total Value	Initial Entry Date	Current Expense Entry Date	Project Lead	Project Financial Officer	Project lead e-mail	Current Expense	Project Location	Comments	ETC …	Security Context of Financial Analyst
Financial Analyst	R	R	R	RU	R		R	RUD	R	RU		
Financial Auditor	CRUD	CRUD	CRUD	CRUD	R	R	R	CRUD	CRUD	RU		
Project Leader	CRUD	CRUD	CRUD	CRUD	R	CRUD	CRUD	CRUD	CRUD	RU		
FinancialAnalyst2	R	R	R	CRUD	R		R	CRUD	CRUD	RU		
FinancialAnalyst3	CRUD	R	CRUD	CRUD	R		R	CRUD	CRUD	RU		
Project Leader2	CRUD	CRUD	CRUD	CRUD	CRUD	CRUD	CRUD	CRUD	CRUD	RU		
Administrator	CRUD	CRUD	CRUD	CRUD	CRUD	CRUD	CRUD	CRUD	CRUD	CRUD	CRUD	
User	R	R	R	R	R	R	R	R	R	R	R	
…												

Figure 2 CRUD by Role

   a. A view template is created (Figure3) by Role showing all columns that a role can view.
   b. Creation of a stored program that will provide a tailored view for each role as tailored by the individual attributes in the CRUD security of the role. [10, 11].
   c. This view can only be restricted, not enhanced. If enhancement is desired, a new role must be defined.
5. View restrictions are by column but apply to rows (Example: Project Location = 'Chicago').

6. When more than one role is in SAML, the application must ask the requester which role is being exercised

7. We assume for this example a column organized relational database. The claims can be built for any database and the former is for illustration only. For database, the permissions are defined in terms of CRUD, normally by columns. The database also applies these CRUD elements for the role. In an identity-based access control system, they would be written for each identity. The use of roles and restrictions simplifies the definitions for appropriate view to be computed.

Transfer of the SAML to a stored program in the database to set the view for the role as limited by other factors.Example – Columns and their CRUDSs are set in the stored view for each role. Rows are restricted by setting acceptable values in various columns. The stored program shall validate the SAML, resolve the role and set the view in the security context of the role (for application of CRUDs to be transferred to the application for further transmittal to the user. The application must have at least four SQL queries programmed in. These include:

1. Execute stored program for view and security context.

2. Create – New entry in the stored view and security context.

3. Update – (column, row) in the stored view and security context.

4. Delete – (column, row) in the stored view and security context.

Any violation of the CRUD for the context view should return an error.

Security Context of Financial Analyst	Project	Total Value	Initial Entry Date	Current Expense Entry Date	Project Lead	Project lead e-mail	Current Expense	Project Location	Comments	Security Context of Financial Analyst
	CRUD	R	R	R	RU	R	R	RUD	RU	
	123400r	5,500,000	12/11/2013	02/04/2014	George Henry	ghenry345@ent.org	3,450,000	Chicago	Initial contracts provided on 02/04/2014 Initial contracts provided on 10/01/2012	
	137800q	2,500,000	08/02/2012	02/04/2014	Helmut Smith	hsmith123@ent.orgl	2,450,000	Chicago	Awaiting final deliverable sign off on 02/04/2014	
	567400r	4,500,000	09/10/2013	12/06/2013	Rita Jones	rjones345@ent.org	3,450,000	Chicago	Initial contracts provided on 12/06/2013 Initial contracts provided on 10/01/2012	
	713200q	3,000,000	08/02/2012	02/04/2014	Janet Smith	jsmith456@ent.org	2,450,000	Chicago	Awaiting final deliverable sign off on 02/04/2014	
	456200r	4,500,000	12/11/2013	02/04/2014	George Henry	ghenry222@rb.com	2,450,000	Chicago	Initial contracts provided on 02/04/2014 Initial contracts provided on 10/01/2012	
	912400t	3,500,000	08/06/2011	02/04/2014	Mike Frank	hmfrank199@fnc.tl	2,450,000	New York	Awaiting final deliverable sign off on 02/04/2014	
	778800r	4,500,000	09/10/2013	12/06/2013	Harry Ga	hga778@chi.com	3,450,000	Chicago	Initial contracts provided on 12/06/2013 Initial contracts provided on 10/01/2012	
	657800s	3,000,000	08/02/2012	02/04/2014	Jim Rich	jrich657@fnl.net	2,450,000	Chicago	Awaiting final deliverable sign off on 02/04/2014	
	. . .									

Figure 3 View Template for Financial Analyst

Data-Driven Annotated Example

Fred is evaluated by the claims engine and claims are slightly modified based upon the database schema and the instructions to the stored program as shown in the table below.

Table 4 Modified SAML Data for Fred (Data-Driven)

SAML: Assertion	
Version ID	Version 2.0
ID	X34.?thik045ml23
Issue Instant	12:11:00 06 May 2014
Issuer	www.securitytokenserver3.net
Signature	(content)
Subject	Fred - X.509 Identity
Subject	Fred2345432
Claims: **Role = Financial Analyst** **Restrict: "Project" ≠ ??????q** **Restrict: "Project Location"= "Chicago"** **Restrict: "Total Value" >=5,000,000** (content)	
SAML: Conditions	
NotBefore	12:11:00 06 May 2014
NotAfter	12:16:00 06 May 2014
Audience	www.mysqldata2.net

The application authenticates itself to the database and triggers the stored program – the SAML for Fred is transferred as shown in Figure 4.

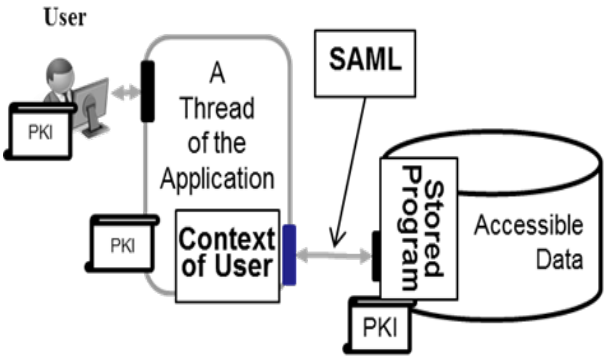


Fig. 4. Posting SAML to Stored Program

The stored program verifies and validates the SAML and pulls up the view template stored in the permissions for Financial Analyst. The stored program then applies the restrictions to the view. This restricted or tailored view is provided to the application for action. Actions are performed but only in the context of the CRUDs in the tailored view. The view is then updated for further work. The stored program modifies the view as shown in Figures 5 and 6.

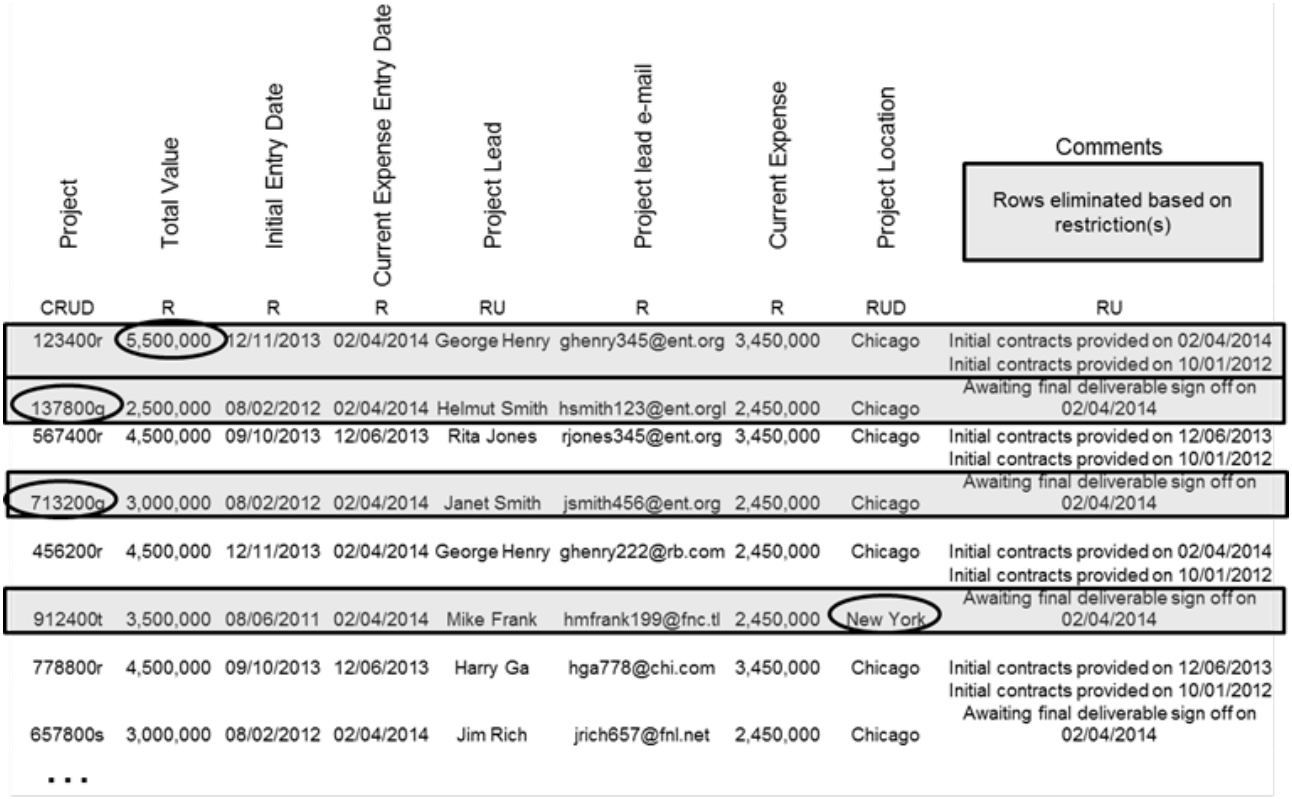


Figure 5 Tailoring the View for Data-Driven Access and Privilege

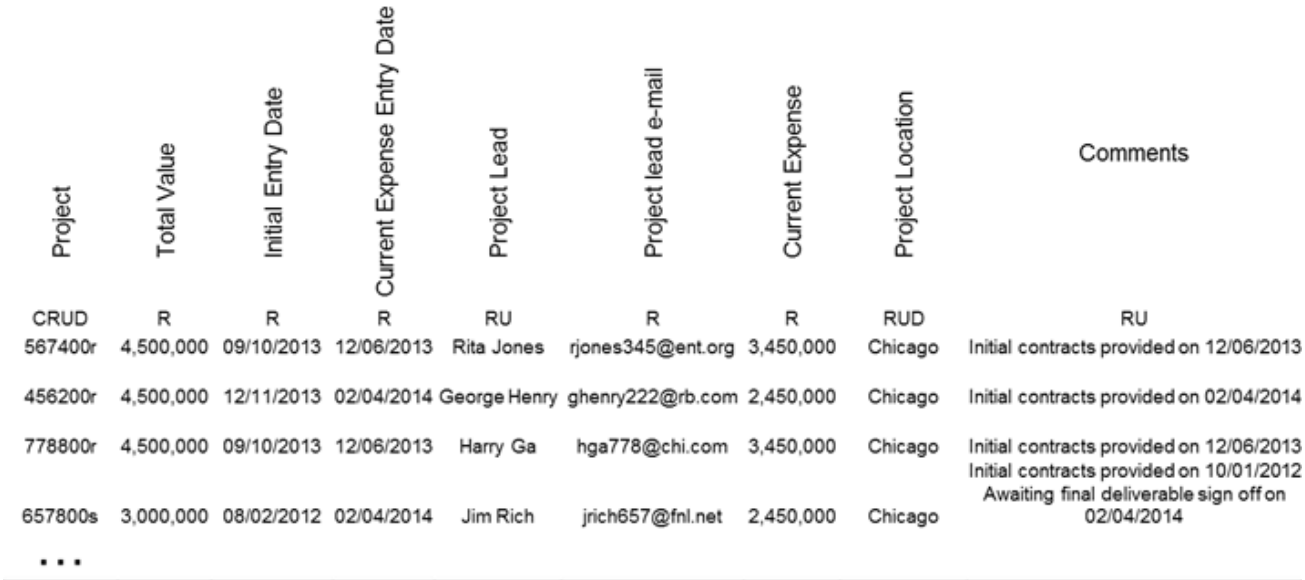


Figure 6 Tailored View for Financial Analyst Fred

The CRUDS in the database will be enforced for the restricted view. Figure 7 shows the exchange with the user. Only accessible data leaves the database.

The A in figure 7 is:

Scripted exchange with application about user request related to tailored view (requests are not filtered)

The B in figure 7 is:

SQL Requests (Read is assumed in the view):

1. Create–New entry - stored view and security context
2. Update–(column, row) - stored view and security context
3. Delete–(column, row) - stored view and security context

No other SQL requests are allowed.

For databases, the data-driven approach has the following advantages and disadvantages:

Advantages:

1. The service has limited access to the database.
2. The database controls Fred's interaction with the database based upon Fred's credentials.
3. Database administrators must establish CRUDs for the role in question.
4. The SQL authority of the service is limited and verified by the database.

Disadvantages:

1. The data owner does have to know the database schema in order to specify access and privilege.
2. Views are moved multiple times.
3. The service computes what is allowable (CRUD) and send computed SQL for what it believes are reasonable requests

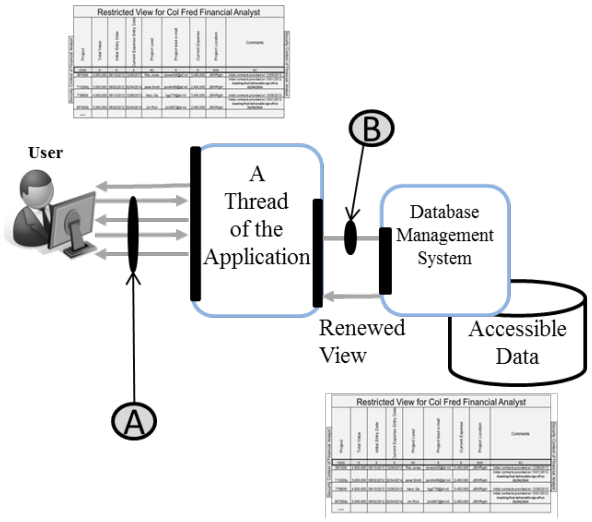


Figure 7 Data-Driven Exchange with User

5. SUMMARY

We have reviewed the basic approaches to the restriction of database access, and the assignment of privilege with databases. The common approach to a web service front end of a DBMS requires the web service to restrict access and privilege based upon the user context. In doing this it must be provided with full access and privilege to the database, and be trusted to limit user access and privilege. The suggested approach builds user-tailored restriction directly into the database and provides the web service fronting the DBMS with the same privilege as the user. At the same time it restricts SQL queries to a fundamental set that will be enforced by the view developed within the database and not at the web service.

This research is part of a body of work for high assurance enterprise computing using web services. Elements of this work include bi-lateral end-to-end authentication using PKI credentials for all person and non-person entities, a separate SAML credential for claims based authorization, full encryption at the transport layer and a defined federation process. Many of the elements of this work are described in [12-18].

REFERENCES

[1]. Jeffrey Ullman 1997: First course in database systems, Prentice–Hall Inc., Simon & Schuster, Page 1, ISBN 0-13-861337-0.

[2]. Tsitchizris, D. C. and F. H. Lochovsky (1982). Data Models. Englewood-Cliffs, Prentice–Hall.

[3]. Beynon-Davies P. (2004). Database Systems 3rd Edition. Palgrave, Basingstoke, UK. ISBN 1-4039-1601-2

[4]. Ken North, "Sets, Data Models and Data Independence", Dr. Dobb's, 10 March 2010

[5]. William Hershey and Carol Easthope, "A set theoretic data structure and retrieval language", Spring Joint Computer Conference, May 1972 in ACM SIGIR Forum, Volume 7, Issue 4 (December 1972), pp. 45–55, DOI=10.1145/1095495.1095500

[6]. Description of a set-theoretic data structure, D. L. Childs, 1968, Technical Report 3 of the CONCOMP (Research in Conversational Use of Computers) Project, University of Michigan, Ann Arbor, Michigan, USA

[7]. Feasibility of a Set-Theoretic Data Structure : A General Structure Based on a Reconstituted Definition of Relation, D. L. Childs, 1968, Technical Report 6 of the CONCOMP (Research in Conversational Use of Computers) Project, University of Michigan, Ann Arbor, Michigan, USA

[8]. "TeleCommunication Systems Signs up as a Reseller of TimesTen; Mobile Operators and Carriers Gain Real-Time Platform for Location-Based Services". Business Wire. 2002-06-24.

[9]. "Structured Query Language (SQL)". International Business Machines. October 27, 2006. Retrieved 2007-06-10.

[10]. My SQL stored Programs and Views, http://docs.oracle.com/cd/E19078-01/mysql/mysql-refman-5.0/stored-programs-views.html#stored-routines-syntax

[11]. Purdue on Using stored procedures to set views, https://www.cs.purdue.edu/homes/ninghui/projects/Topics/DB_FineGrained.html .

[12]. Coimbatore Chandersekaran and William R. Simpson, The 3rd International Multi-Conf. on Engineering and Technological Innovation: IMETI2010, Volume 2, "A SAML Framework for Delegation, Attribution and Least Privilege", pages 303-308, Orlando, FL., July 2010.

[13]. William R. Simpson and Coimbatore Chandersekaran, The 3rd International Multi-Conference on Engineering and Technological Innovation: IMETI2010, Volume 2, "Use Case Based Access Control", pages 297-302, Orlando, FL., July 2010.

[14]. William R. Simpson and Coimbatore Chandersekaran, International Journal of Computer Technology and Application (IJCTA), "An Agent-Based Web-Services Monitoring System" Vol. 2, No. 9, September 2011, page 675-685.

[15]. William R. Simpson, Coimbatore Chandersekaran and Ryan Wagner, Lecture Notes in Engineering and Computer Science, Proceedings World Congress on Engineering and Computer Science 2011, Volume I, "High Assurance Challenges for Cloud Computing", pp. 61-66, San Francisco, October 2011.

[16]. Coimbatore Chandersekaran and William R. Simpson, Lecture Notes in Engineering and Computer Science, Proceedings World Congress on Engineering 2012, The 2012 International Conference of Information Security and Internet Engineering, Volume I, "Claims-Based Enterprise-Wide Access Control", pp. 524-529, London, July 2012.

[17]. William R. Simpson and Coimbatore Chandersekaran, Lecture Notes in Engineering and Computer Science, Proceedings World Congress on Engineering 2012, The 2012 International Conference of Information Security and Internet Engineering, Volume I, "Assured Content Delivery in the Enterprise", pp. 555-560, London, July 2012.

[18]. Coimbatore Chandersekaran and William R. Simpson, International Journal of Scientific Computing, Vol. 6, No. 2, "A Uniform Claims-Based Access Control for the Enterprise", December 2012, ISSN: 0973-578X, pp. 1-23.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE (DD-MM-YY)	2. REPORT TYPE	3. DATES COVERED (From – To)
28-04-15	Non-Standard	

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Database Access – Application-Driven versus Data-Driven	HQ0034-14-D-0001
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBERS

6. AUTHOR(S)	5d. PROJECT NUMBER
William R. Simpson	BC-5-2283
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESSES	8. PERFORMING ORGANIZATION REPORT NUMBER
Institute for Defense Analyses 4850 Mark Center Drive Alexandria, VA 22311-1882	NS D-5403 H15-000019

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR'S / MONITOR'S ACRONYM
Frank P. Konieczny USAF HQ USAF SAF/CIO A6	SAF/CIO/CTO
	11. SPONSOR'S / MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

Project Leader: William R. Simpson

14. ABSTRACT

A multitude of commercial applications rely on Data Base Management Systems (DBMS) that provide an organized collection of data; for example, modelling the availability of flights and seating in a way that supports reservation and sales of air transportation. DBMSs are specially designed software applications that interact with other applications and users to capture and analyze data. A general-purpose DBMS is a software system designed to allow the definition, creation, querying, update, and administration of databases. For the purposes of this paper, we assume that the database is front-ended by a web service for database access and query. This paper discusses the current approach to database access and privilege by web services and the changes that are required by a high assurance end-to-end approach. The latter rely on a well-formed security paradigm for the enterprise.

15. SUBJECT TERMS

Database Access, DBMS, Access Control, IT Security, Integrity

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Frank P. Konieczny
Unclassified	Unclassified	Unclassified	Unlimited	6	19b. TELEPHONE NUMBER (Include Area Code) (703) 697-1308