Safety of data warehouse for remote laboratories in Laboratory Management System and network of road junctions

L. Pálka, L. Váňová, F. Schauer, K.Vlček and R. Jašek

***Abstract*—** In spite of the fact that remote laboratories and management of road intersection have been existing for at least three decades, virtually no attention has been devoted to the security of this new subject. The paper deals with the security of the data storage of the Datacenter (DTC), with remote laboratories working under the Laboratory Management System (LMS) and to the construction of central management and analysis of collected data using dynamic modeling information obtained for centralized streamline the flow of traffic in the city. Especially, the security risks for the data storage and corresponding data processing to ensure the operation of the data warehouse are described in detail. This article describes the newly proposed system in terms of communication data streams using the tools of the modern age, based on the proposed dynamically controlled system with real time feedback. DTC also will serve as the central system for analysis and storage of data from LMS and system for network of road junctions, therefore it is necessary to provide the service, as described in the following text article.

***Keywords*—** data security, database security, data storage, rig, remote experiments, work with data in the data warehouse, data warehouse design, database security services, database firewall, C2 auditing, Network of road.

I. INTRODUCTION- REMOTE LABORATORIES AND LABORATORY MANAGEMENT SYSTEMS - STATE OF THE ART

At the present stage of the development of Information Communication Technologies (ICT) there are plenty simulations and remote experiments for science and education purposes [1][3][9]. Remote experiments and informatics resources are tools that are closely related and definitely need to process and store substantial amounts of data. Data, used with remote laboratories (RL), may have the form of simple queries, data analysis, comparative analysis and data mining for associative analysis, extrapolation or predictive trend analysis. Surprisingly, in spite of the fact the RL have been existing for at least three decades [1], virtually little attention has been devoted to the security of this new ICT subject [8].

The present paper deals with the security and safeguarding of the data processed and especially stored in the DTC with remote laboratories, especially that with Laboratory Management System (LMS).

In this connection we will use the term data warehouse (DW) (see Figure 1 data warehouse functioning) [4][6], referring to a complex system that allows to collect, organize, store and share consolidated data from all available operating systems, optimized for reporting, analysis, and data archiving. Users exploit the data warehouse for reporting, in this respect synonymous for business intelligence technology, based on the use of the data and its accumulation, preservation and presentation. The working principle of the DW is that the data we need to process is first stored into the database in a raw state, then follows data classifying using OLAP[1] in data cubes l (see Figure 1) and then, using architecture model (e.g. experiments evaluation or search), and subsequent results storing and reporting.

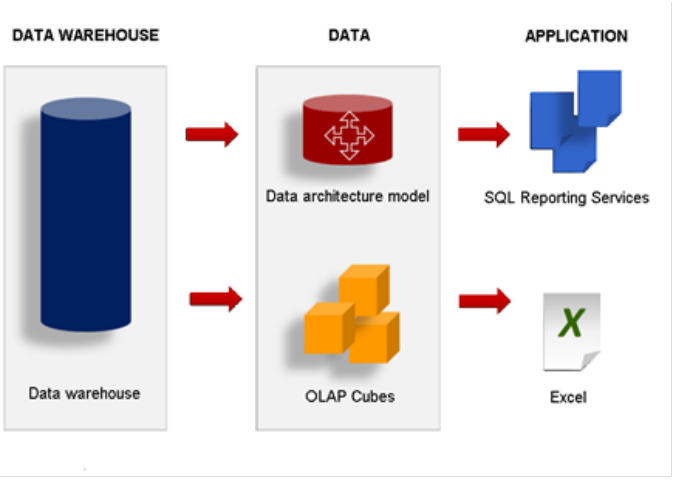

Fig. 1. Schematic representation of the data warehouse functioning

The layout of the paper is following. In Chapter 1, the typical scheme of the communication of a typical remote experiment (RE) , built as the finite-state machine (FSM) [2], using the Internet School Experimental System (ISES) physical hardware, is described[10] (for the ease of reading we will next denote the set of the individual remote experiment by the word rig). Also, the control program compiling and the type of the data generated and transferred is

[1]Online analytical processing: OLAP tools enable users to analyze multidimensional data interactively from multiple perspectives.

shortly described. More detailes may be found in corresponding literature [2][8].

The Chapter 2 is devoted to describing the architecture Remote Experiments and the corresponding integrating management system, called for our purposes Remote Laboratory Management System (RLMS) [10]. The Chapter 3 is deals with the management of intersection, and "Management intersection as needed". The Chapter 4 is devoted to the actual risks, the remote laboratories are exposed. [13][14] The Chapter 4 is then focused on the corresponding security of a typical DW of a university datacentre (DTC) with LMS for remote laboratories. The Chapter 5 and 6 is devoted to describing the services of security database and firewall services of database. The final Chapter 7 and 8 is then focused on security roles of database, followed by Datastore in datawarehouse in the future and conclusions.

II. ISES Remote experiment (RE) and Remote laboratory management system (RLMS) – tools used

Only recently has emerged a serious problem stemming from analysis of research data. ISES is a powerful tool for process and experiments control, acquisition, collecting and data processing in real time. Let us mention the basic features of the ISES system, more detailed description may be found elsewhere [2][10]. The basis of the system is ISES board, which is available in several versions, differing depending on the number of inputs/outputs and also on type of communication with the control PC (by PCI card, USB connector, Wi-Fi). To this board are, by a unique connector, plugged in sensors like: ammeter, voltmeter, thermometer, position sensor, ohmmeter, load cell, anemometer, microphones, sonar, light gate, pH meter, conductivity meter, heart rate monitor, etc. [8,22]. The layout arrangement of the RE is in Figure 2.

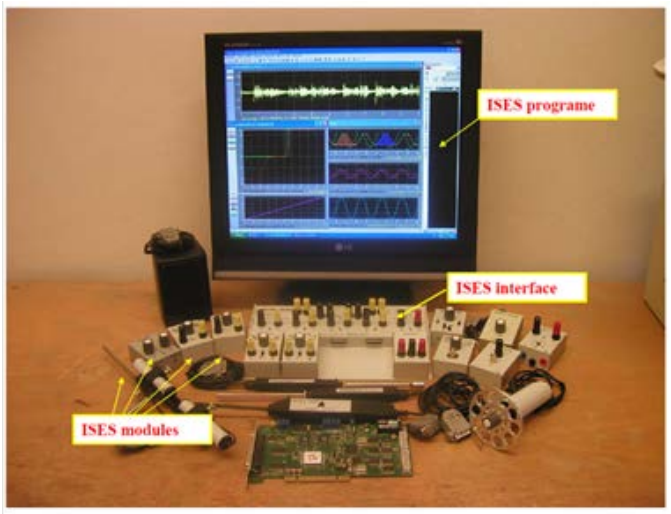

Fig. 2. ISES – Internet School Experimental System.

The most important component is the Measureserver module, functioning as finite-state machine (FSM) controlled by the controlling program of the PSC script file. The main feature of the Measureserver, is to communicate with the physical hardware and to check the setup of the ISES panel and its sensors/meters and to take care about their data collection and processing. Other parts of the system are ImageServer for life view of the remote experiment, Web server for the communication between RE and the client. Also, aprt of the RE is the communication web page as the interface communicating with the RE over the Internet by the client.

The invevitable part of the RE system is the data warehouse for the storage of data for all above systems. It is a centralized repository service to Measureserver, web server, image server and other components of the solution.

In this article we will discuss this last part of the system with respect of data security, but not only from the perspective a single RE, but of the whole RLMS. The layout arrangement of the RE is in Figure 3[23,24].

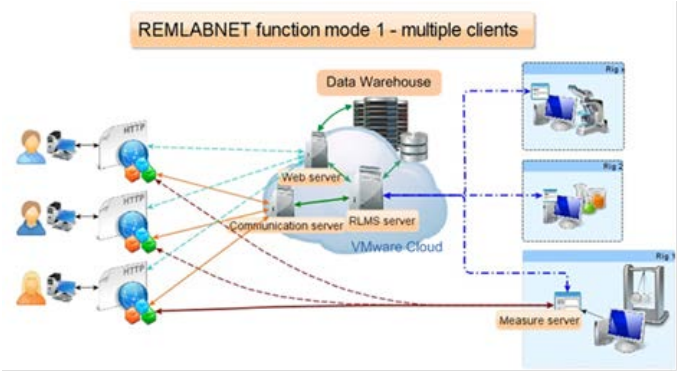

Fig. 3. REMLABNET function mode 1 – multiple clients.

A serious problem stemming from security aspects of e-laboratories has emerged only recently. Let us describe first the data generated and that are processed in every ISES rig working on the communication principle server-client and the functioning of the superordinate Remote Laboratory management system (RLMS). The controlling of every rig by client is enabled via Web interface, by means of which the user can perform the appropriate settings, options, and starting or stopping of the remote experiment (RE).

The measured data from the experiment delivered from the MeasureServer are stored in the data storage . RLMS is a system for a database-driven Web application. This is seen from the figure 3, where LMS is divided terms of safety in three parts. Database-driven Web applications are very common in today's Web-enabled society. LMS consist of a back-end database with Web pages that contain server-side script written in a programming language that is capable of extracting specific information from a database depending on various dynamic interactions with the user.

Remote experiments problematic is the topic of scientific activities of the group since 2005, when the first remote experiment started to be built. We relied on the enormous know-how of Assoc. Prof. F. Lustig from Department of Physics Education of Faculty of Mathematics and Physics, Charles University in Prague, where the universal and very

useful modular computer oriented set Internet School Experimental System (ISES) was designed at the beginning of 90th [Lustig, F. and Schauer, F.: "Creative laboratory experiments for basic physics using computer data collection and evaluation exemplified on the ISES", Proceedings first european conference on Physics Teaching in Engineering Education, 125-131, Copenhagen, Denmark, ed. Oehlenschlaeger, 1997].

A database architecture for Remote Experiments

A database-driven Web application for LMS has three tiers: presentation, logic, and storage. To help you better understand how Web application technologies interact to present you with a feature-rich Web experience, Figure 4 illustrates the three-tier schema.

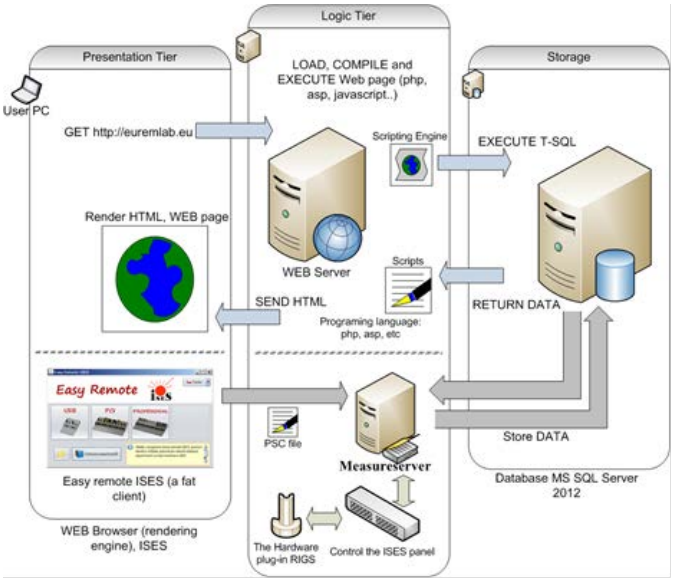

Fig. 4. Three-Tier Architecture Remote Experiments.

The presentation tier is the topmost level of the application. It displays information related to such services such as information web page about LMS, reservation system, and it communicates with other tiers by outputting results to the browser/client tier and all other tiers in the network.

The logic tier is pulled out from the presentation tier, and as its own layer, it controls an application's functionality by performing detailed processing. The data tier consists of database servers. Here, information is stored and retrieved. This tier keeps data independent from MeasureServer, reservation system and web server. Giving data their own tier also improves scalability and performance. In Figure 4, the Web browser (presentation) sends requests to the middle tier (logic), which services them by making queries and updates against the database (storage). A fundamental rule in a three-tier architecture is that the presentation tier never communicates directly with the data tier; in a three-tier model, all communication must pass through the middleware tier now. Conceptually, the three-tier architecture is linear.

In Figure 4, the user open his Web browser and connects to web page for remote experiments (http://euremlab.eu). The Web server that resides in the logic tier loads the script from the file system and passes it through its scripting engine, where it is parsed and executed. The script opens a connection to the storage tier using a database connector and executes an SQL statement against the database. The database returns the data to the database connector, which is passed to the scripting engine within the logic tier. The logic tier then implements any application or business logic rules before returning a Web page in HTML format to the user's Web browser within the presentation tier. The user's Web browser renders the HTML and presents the user with a graphical representation of the code. All of this happens in a matter of seconds and is transparent to the user.

In this slide show we describe the html code displayed on the station.

This component is represented in the LMS applications REMLABNET web server [2][16]. Web Server takes care about available ports for the LAN communication. It is a common habit by the system administrators to block all ports for security reasons except ports 80 and 443. HttpRelayServer dynamically changes communication port to that available.

We extend this model by MeasureServer. MS is most important informatics SW component of the remote experiment, functioning as finite-state machine (FSM). It communicates with the apparatus, processes the measured data and control commands. The main feature of the MeasureServer is setting of the ISES panel, the sensors/meters for data collection and processing the control commands. [16] The module dynamically responds to signals from the physical HW, as well as the commands transmitted from the client´s interface. MeasureServer has three main components - the MeasureServer core, Hardware plug-in and PSC script file. The MeasureServer core is responsible for the data and command transfer, client handling and for execution of all the controlling commands. The execution of process is controlled by the PSC script file which is directly imported to MeasureServer. The PSC script is a unique programming language specially designed for the ISES system. The PSC script is non-compliable language defining the MeasureServer's core behavior of remote experiment. The Hardware plug-in provides required functionality to control the ISES panel translating signals from/to the physical HW apparatus.

In Figure 4, the user open his rig(s) and connects to MeasureServer. The MeasureServer that resides in the logic tier loads the script (PSC file) from the file system and passes it through its scripting engine where it is parsed and executed. The MS opens a connection to the storage tier using a database connector and executes an SQL[2] statement against the database. The database returns the data to the database connector and then returns the requested data to MS. Before returning the data to the Web server. The Web server then implements any final logic (results and details of the

[2]Structured Query Language: SQL is a special-purpose programming language designed for managing data held in a relational database management system.

experiment) before presenting the data in HTML format to the user's Web browser within the presentation tier. The user's Web browser renders the HTML and presents the user with a graphical representation of the code. All of this happens in a matter of seconds and is transparent to the user.

Based on the system of RE, clients have a non-stop accessibility to enter RL through their web browser's interface connected to Internet from anywhere.[16] The most significant advantage is the real-time experimenting with the apparatus installed in a laboratory. An authorized student can comfortably communicate with the remote apparatus of the visualized RE via a web page. The apparatus promptly reacts and sends adequate responses/signals/data through particular subsystems back to the target client/student. After the completion of RE the student will arrange the data in formatted, sorted and filtered form and also in graphic charts displayed on the screen and find the corresponding answers regarding the phenomena observed.

III. A DATABASE ARCHITECTURE FOR NETWORK OF ROAD JUNCTIONS

In this article we will focus on the proposed system as a functional unit will be described in detail and communication of information management and data transfer via data warehouse. The system as a whole uses technology already in place at intersections and implements this system a new element to control intersections as a whole to streamline the flow of traffic. It is therefore an effectively controlled system with a new factor for modeling data in order to increase efficiency. The benefits of this proposed method is its management efficiency and increase efficiency with minimal cost to use in order to optimize and increase efficiency.

Controlled system, there is a system of intersections controlled by traffic lights. Intersections exist in various designs and graphic shapes, but we will only deal with intersections controlled by traffic lights, because we would like to use sensors on these intersections to increase the flow of road traffic in cities and data from these sensors to analyze in the data warehouse.

Traffic-light controlled intersections are intersections controlled by traffic light (crossing signal is periodic on leave); are suitable for driving any intersections, the disadvantage is the reduction of traffic flow; traffic light controlled intersections can be influenced by appropriate timing [25].

The System intersections controlled by traffic lights, as we all know, can be viewed as a decentralized system which has a higher organization and their direct function is not controlled, and therefore that part of the system works independently, which is the default state of a typical intersection. Factor decentralization works here in the form of information flow, the headquarters knows only whether the management system itself intersection is functional or not. Our task is to use sensors to centralize these parts and connection system into a controlled and centralized.

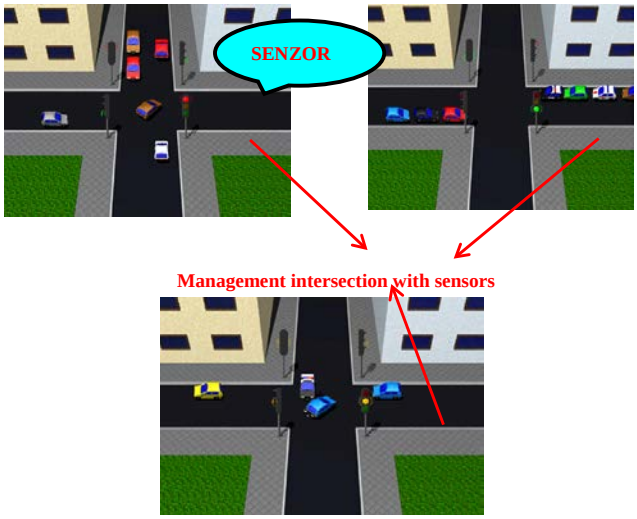

Fig. 5. Centralized control of intersections with sensors.

This system would use real data obtained from sensors intersections (see Figure 5), based on which we can control the intersections as needed. Traffic control at intersections would become more efficient, which would result in an increase in traffic flow and congestion relief with which continually point out the bottlenecks in the infrastructure.

We have to bring the situation in which the car passes smoothly through one, two intersections and at the third intersection remains trapped in the convoy. As mentioned above, each intersection is timed. Some intersections are programmed independently of the other intersections, but in larger cities are timed intersections, depending on the downstream intersection. From nearby, this situation was addressed in many works on roads in Otrokovice. This method of controlling traffic lights is very effective because it enables drivers in speeding smoothly ride the city, without causing unwanted congestion and environmental pollution by exhaust gases.

When you think but think about it, so we come to the question: "Will it be sufficient

to control the intersection?" The answer is that, of course, integration management intersections will increase the flow of traffic, but not certainly not what we wanted to achieve. And what about the railroad crossing, traffic accident, road closure, an accident on the road?

It is logical that it is important not only to interconnect the intersection between them, but also intersections with other possible modes (see Figure 6). It is also important

to connect both real data obtained from sensors intersections, other modes of transport, but also information from the Integrated Rescue Service (police, fire brigade, emergency medical service). The aim is for the system over time to connect as many organizations, both public and private.

This system would allow drivers to seamlessly pass through the city on the basis

of data obtained from sensors intersections, eliminate congestion and be more environmentally friendly.

At this point, we will explain the communication between the control and controlled system (see Figure 6). Data obtained from sensors intersections, other modes of transport, integrated Brigade and other organizations would be transmitted to the data store from which data would be transmitted to the control system, which will be explained in more detail in the next chapter. Based on the evaluated data from the control system we can ensure the effective management of intersections according to real situations taking place at the crossroads at the moment.

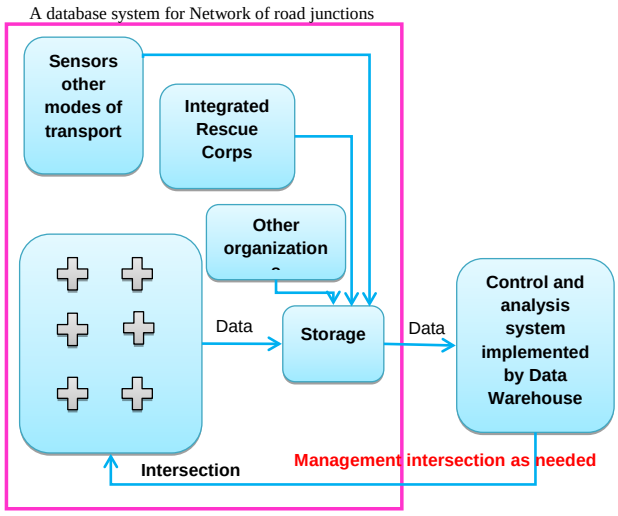

Fig. 6. Control and controlled system.

IV. DataWarehouse Security Risks of Remote Experiments and Security Risks of Network of Road Junctions

LMS is a specific technically sophisticated complex system. The availability of these system and the sensitivity of the data that they store and process are becoming very important. Web page that presentation LMS on Internet contains supporting infrastructure and environments use diverse technologies and can contain a significant amount of modified and customized codes.[19,20] The very nature of their feature-rich design and their capability to collate, process, and disseminate information over the Internet or from within an intranet makes them a popular target for attack.[21] Also, since the network security technology market has matured and there are fewer opportunities to breach information systems through network-based vulnerabilities, hackers are increasingly switching their focus to attempting to compromise applications.

SQL Injection of database attack

SQL injection is an attack in which the SQL code is inserted or appended into application/user input parameters that are later passed to a back-end SQL server for parsing and execution. Any procedure that constructs SQL statements could potentially be vulnerable, as the diverse nature of SQL and the methods available for constructing it provide a wealth of coding options. The primary form of SQL injection consists of direct insertion of code into parameters that are concatenated with SQL commands and executed. A less direct attack injects malicious code into strings that are destined for storage in a table or as metadata. When the stored strings are subsequently concatenated into a dynamic SQL command, the malicious code is executed. When a Web application fails to properly sanitize the parameters which are passed to dynamically created SQL statements (even when using parameterization techniques) it is possible for an attacker to alter the construction of back-end SQL statements. [15] When an attacker is able to modify an SQL statement, the statement will execute with the same rights as the application user; when using the SQL server to execute commands that interact with the operating system, the process will run with the same permissions as the component that executed the command (e.g. database server, application server, or Web server), which is often highly privileged.

It is important to have a clear understanding of how your data entry influences a SQL query and what kind of response you could expect from the server.

Figure 7 shows how the data sent from the browser are used in creating a SQL statement and how the results are returned to the browser.

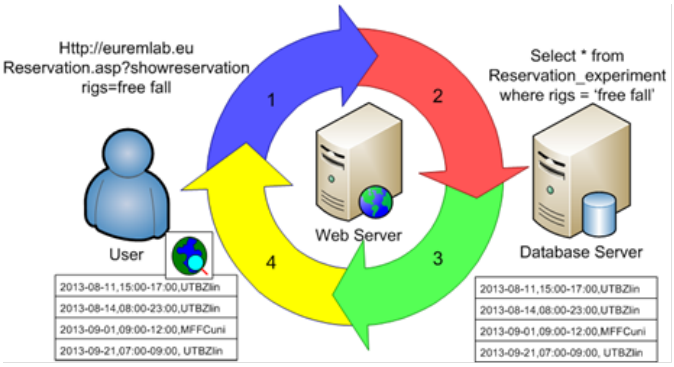

Fig. 7. Information Flow during a SQL Injection Error.

If an attacker to modify the website to query the database, such as a change in url, web server just creates a SQL query, parses the results, and displays the results to the user. The database server receives the query and returns the results to the Web server. This is very important for exploiting SQL injection vulnerabilities because if you can manipulate the SQL statement and make the database server return arbitrary data (such as usernames and passwords from the Web site) the Web server has no means to verify whether the data are legitimate and will therefore pass the data back to the attacker. This is one of the most common attacks on the database and need from development to propose the necessary measures.
Of recent history:

In 2011, Sony suffered a 23 day network outage after a breach of security that allowed the theft of approximately 77 million registered accounts from its PlayStation Network. It is to date the largest computer data exploit in history. A month later, hackers claimed in a press release to have stolen personal information of 1 million users from the website of Sony Pictures by a single SQL injection attack [17].

DoS and DDoS of database attack

Another attack, more difficult to prevent, is a DoS, or DDoS (Denial of Service, or Distributed Denial of Service), the overloading of a website or any kind of server with requests, for the purpose of bringing it down.

Denial of Service (DoS) is an inelegant but effective attack against web, database, and any type of public server. The goal is to overload the server with requests to crash it or make it unavailable for normal operations. A DoS is most of the time targeted towards a web server, and affects SQL server on the rebound. The first way to handle this is to protect the web server, for example, with a network firewall, which will automatically block suspicious IP addresses, or a Web Application Firewall (WAF). Here, we will provide some recipes to increase protection in SQL Server itself.

How to do it...:

DoS risks are increased when you allow queries to be created dynamically in the client application, especially when you offer multi-criteria search forms. Since the user can search with any combination of criteria, it can lead to complex queries where it will take time to execute and exploit the resources of the server. A few of these queries running simultaneously can effectively decrease the performances of the whole server.

V. SECURITY OF SCIENTIFIC REMOTE EXPERIMENTS AND SECURITY RISKS OF NETWORK OF ROAD JUNCTIONS

A greater level of baseline, hardware-enforced security features are important in all categories of remote laboratory system for part systems such as database server, MeasureServer, web server, reservation system, control system etc. These capabilities will protect the information on the device itself, and the information that is accessed from the device. They'll enable greater trust in the device, and because of this trust we'll be able to provide users of the device with access to more resources.

For LMS security, these baseline hardware security capabilities will provide help in key focus areas, including threat management, ID and access management, data protection, and remote monitoring. Some expected baseline capabilities include protected environments, encryption, hardware acceleration, enhanced recovery, and integration with security software.

SQL Injection of database based defense

SQL injection is the action of adding characters to a SQL query in order to modify its action and execute an exploit, such as getting more information, modifying data or data structures, or even getting access to the underlying operating system of the database server. [17][6][7] It can happen when a dynamic ad-hoc SQL query is built in the application code.

Let's see how it is necessary to protect the data storage LMS and what needs to be done on the website of remote experiments.

The SQL query is built dynamically in a string in web page of remote experiment in browser on client PC. [4][12][15] It leaves a possibility of something to be added to cause harm. we can find numerous and cunning ways to manipulate a query, by using SQL operators, functions, or constructs that can circumvent basic protection. For example, in T-SQL, the BULK INSERT command could be used to read the content of a file on the disk and return the result as a result set, or the xp_cmdshell extended stored procedure could be used to run Windows or even Active Directory commands. To eliminate the threat, replace those strings with parameterized stored procedures. The parameters will never be evaluated as a part of the query syntax and cannot be used for adding other behavior to it. They could still be used to get more information than expected, but not to run commands.

The best way to stay safe is to encapsulate SQL code inside parameterized stored procedures. But this sometimes defeats the first purpose of building a query dynamically: to fit a multi-criteria sear Chapter For example, if the name of the rig is part of the search, we add a JOIN to the rigs table in the query; otherwise we don't, which simplifies the query and optimizes performances.

Removing the dynamic SQL is the best solution in terms of security, because there will be no chance for an attacker to inject code inside the SQL statement. The variables can now only replace values that are evaluated inside a comparison.

DoS and DDoS of database based defense

The first thing to do is to improve the quality of your code. One simple thing you can do is to ensure you have created the needed indexes on your tables, to avoid costly table scans. You can use the Database Tuning Advisor (DTA) packaged with the SQL Server client tools. You can make a .sql file containing some costly queries, and feed the DTA with it.

You can also limit the number of concurrent connections allowed on SQL Server. By default, the limit is 32,767 in MS SQL Server 2012 (the configuration value shows 0 in this case). You can change the value in the configuration pages of the instance (see the next screenshot, Figure 8) by T-SQL and set:

```
SELECT value_in_use
FROM sys.configurations
WHERE name = 'user connections';
```

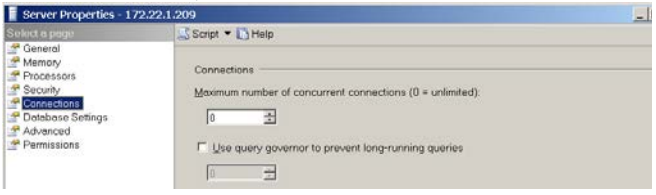

Fig. 8. Set restriction long-running queries.

In the previous screenshot, you can see another server configuration that could be useful: Use query governor to prevent long-running queries. If activated, the SQL optimizer will block execution of any query estimated to cost more than the number of seconds defined in the value. This is far from bullet-proof, as it is simply an estimation from the query optimizer, in pseudo-seconds (just a way of weighing plans to

compare). If you choose, for example, a value of 30 seconds, it simply means that SQL server will not execute a query that it estimates costing more than 30 seconds. The actual query could finally run much faster or conversely run for hours if locks are blocking it. But it is a way to stop queries that are estimated to be heavy, and limit the risk that the server will be overloaded by a few queries.

Another way to limit resource usage is the use of Resource Governor. This addition to the SQL Server administrator toolbox is available only in the Enterprise edition. With it, you can define workload groups inside resource pools. In short, you can limit the amount of CPU and memory allocated to a group of sessions. For that, you create a classifier function that returns the name of a workload group. This function allows you to define the classification rules you want, based on SQL code, system variables, and functions (the login name, the time of day, and so on). Then you declare this function in Resource Governor along with you pools and groups.

VI. Tools communication services of data Warehouse and its implementation

The safety of data in the data warehouse is still the topic of the debate in the field. Generally, its specific features are given by the fact that the data warehouse is a central store of data and contains already implemented functions for data mining and reporting capabilities that stores integrated, validated, detailed data from multiple information systems, which are and should be also readily available. These requirements result in the higher degree of the data protection needs, irrespective of the fact if remote laboratories are involved or not.

General communication policy

• Already in the planning of the data warehouse it is necessary to define the corresponding groups of end users, their key characteristics and the basic settings of safety rules. To end users can be allocated specific groups and subsequently associated into them with specific rights e.g a student, teacher, administrator, developer, systems engineer, etc. This part of the document thus contains elements of security applied to individual users and groups of users. In the result, already starting document of the data warehouse contains the security rules applied to separate groups of users and even individual users themselves. On the other hand, the security policy is not a static affair, and major changes is always to be accounted for, such as:

• Expansion of the data warehouse with regard to its explanatory power, i.e. adding subject areas. In our case, new experiments, lab reports and rigs.

• Creation and modifying of the departmental or specialized databases in response to the expansion for new remote laboratories and their new rigs.

• New versions of the system, etc.

Experience advices to allocate responsibility from the very outset of the project for each part of the security system and who is responsible for the entire system, which creates corresponding roles and permissions for individual groups. For example: For security of the system is responsible ICT Team, which assigns permissions for individual groups and for the system modifications System engineer group.. For new rigs setup is responsible the group Teacher. It thus should never happen that a group intervenes into a system beyond their authorization and corresponding requirement must always be transferred to the specialists with adequate rights.

VII. Firewall of database tool based communication defense

A Web Application Firewall (WAF) is a software or hardware appliance that checks the incoming and outgoing network traffic by analyzing the data packets on a rule set and determining whether they should be allowed through or not, that you put in front of your web application. It can be very effective to detect injection attempts or Denial of Service (DoS) attacks. These attacks are very dangerous for data storage remote experiments.

Attackers are imaginative and it is very difficult to build a 100 percent effective protection against them. Attack like DoS, DDoS and injection is a real issue for remote experiments. The solution might be to invest in a SQL firewall or a Web Application Firewall (WAF). A SQL firewall sits between the client and the SQL Server, and monitors all SQL queries to intercept injection attempts based on suspicious patterns found in the SQL code.

To set the firewall protection as the system remote experiments against these attacks, together with a secure web site inquiries interlock function is used injection in the firewall.

In the Policies of firewall rulles, click on Create New to add a policy. Select Risk Based IPS/IDS in the Rule Type drop-down box, a database we previously created in the Databases window, or All Databases in the Database drop-down list. The other option will appear after this preliminary choice. You can choose the Action mode: with Active Protection IPS you can automatically intercept detected injection attempts, and with Monitoring IDS you let it go but are notified. For our example, choose Active Protection. Then check SQL Injection Detection and set a Blocking action.

Follows a simple method will protect the system against the majority of known attacks. This setting will have better sleep and remote experiments will work safely.

Port settings for remote experimentation is commonplace. Due to a data warehouse is to communication on port TCP / UDP 1433 with Web Server, open ports for system updates or aplication mechanisms and open ports for MeasureServer.

VIII. Roles of database communication tool based defense

Already SQL roles enable to sort users into a certain groups with pre formulated rights for a user even for a client of remote laboratories. To easily manage the database rights, SQL Server

provides several functionalities for their management, both for the operating system and for the database itself [4].

There are two types of roles for SQL Server: fixed database roles that are predefined in the database and flexible database roles you can create.

Fixed database roles are defined at the database (DB) level and exist in each database. Members of the db_owner and db_securityadmin database posses full rights to manage the DB and change both data and metadata, but only db_owner can add or delete DB account or change the group for a DB user (See Figure 9 for the more detail of the rights of the DB specific group).

Database-level role name	Description
dbowner	Members of the **db_owner** fixed database role can perform all configuration and maintenance activities on the database, and can also drop the database.
dbsecurityadmin	Members of the **db_securityadmin** fixed database role can modify role membership and manage permissions. Adding principals to this role could enable unintended privilege escalation.
dbaccessadmin	Members of the **db_accessadmin** fixed database role can add or remove access to the database for Windows logins, Windows groups, and SQL Server logins.
dbbackupoperator	Members of the **db_backupoperator** fixed database role can back up the database.
dbddladmin	Members of the **db_ddladmin** fixed database role can run any Data Definition Language (DDL) command in a database.
dbdatawriter	Members of the **db_datawriter** fixed database role can add, delete, or change data in all user tables.
dbdatareader	Members of the **db_datareader** fixed database role can read all data from all user tables.
dbdenydatawriter	Members of the **db_denydatawriter** fixed database role cannot add, modify, or delete any data in the user tables within a database.
dbdenydatareader	Members of the **db_denydatareader** fixed database role cannot read any data in the user tables within a database.

Fig. 9. Rights for MS SQL Server groups 2012 [4].

IX. DATASTORE IN DATAWAREHOUSE IN THE FUTURE

Let us predict, in the light of constantly developing ICT, the way of the safety precautions of the future data warehousing. Among the most growing trends, let us mention:

- Local data storage warehouseswillbe integrated into mega-integrated database centres of mega - companies such as Microsoft or Google,
- Authentication will run via a centralized authentication service USERID,
- Data scheme will be managed by mega-datacentres services and edited by higher order instruments. Thus, global consistency and the general design of structures to store and work with data will be ensured.
- Data mining will be integrated into functionalities of data protection and auditing.
- Exiting data warehouses will be more open to the needs of governmental and other institutions to control corruption and terrorism.
- Data warehouses will include integration buses and standards for reciprocal linking will be developed.
- Direct access of individuals and companies to research data warehouses worldwide for teaching, sharing information, and the like will be assured.
- Mega development of remote data centres, simulation technology centres, global centralization of knowledge and clustering into a single unit through integrators.
- Direct and audit subjected access of persons to information in the global knowledge centres (mega-integrated database centres).

X. CONCLUSION

This work describes a series of recommendations and procedures to secure data storage in the scheme of the data warehouse for the needs of remote laboratories and Network of road junctions. The work includes a vision for the future regarding security and direction of data warehouses, aiming to direct readers to the problems of data warehouses from all perspectives and to learn, what are the risks of today and how to comprehend security. In the course of the work on the data warehouse we learned how to realize a safety problem as well as design security. We believe, that the article sheds some light on a number of acute problems but simultaneously opened to us many other questions to consider in connection with data warehouse security. The article also describes the introduction of new terms on issues of security and shares experience in terms of theory and our practical experience.

The advantage of the system are relatively low financial costs of implementation by Microsoft Data Warehouse for the price of the possibilities of secure working with the data obtained. There is presented a modern and secure system that is created by the idea of using information technology for the centralized management of intersections, sysem RLMS or other systems for analysis, which can be relatively easily connected.

ACKNOWLEDGMENT

The paper was published thanks to the Grant of the Internal Agency of UTB No IGA/FAI/2014/042. One of us acknowledges the partial support of the Slovak Research and Development Agency, project no. APVV-0096-11, the Scientific Grant Agency VEGA, project no. 2/0157/12, and the KEGA Agency projects No 011TTU-4/2012 and 020TTU-4/2013.

REFERENCES

[1] The whole system is detail described in the project proposal Submitted Project Grant Agency of the Czech Republic: INFORMATICS MEANS FOR GRID OF e-LABORATORIES – PROJECT REMLABNET, 2013

[2] KRBEČEK, M. Possible_utilization_of_the_artificial_intelligence elements_in_the_creation_of_remote_experiments. [online]. 2012, č. 1 [cit. 2013-06-26]

[3] Grid Remote Laboratory Management System. Sahara Reaches Europe. 2013, č. 1.

[4] Database-Level Roles [online]. 2012 [cit. 2013-06-26], http://msdn.microsoft.com/en-us/library/ms189121.aspx

[5] ALEXANDER, D., FINCH, A., SUTTON, D. a TAYLOR, A. Information Security Management Principles. 2. vyd. bcs, 2013. ISBN 9781780171753.

[6] SCHULZ. Cloud and Virtual Data Storage Networking. teChapterChapterbooks, 2011. ISBN 978-1439851739.

[7] Data Warehouse [online]. 2013 [cit. 2013-06-26], http://en.wikipedia.org/wiki/Data_warehouse

[8] SCHAUER, F., LUSTIG, F. a OŽVOLDOVÁ, M. Innovations 2011: World Innovations in Engineering Education and Research: Internet Natural Science Remote e-Laboratory (INTRE-L) for Remote Experiments. USA: iNEER, 2011, s. 51-68. 1. ISBN 978-0-9818868-2-4.
[9] SCHAUER, F. a OŽVOLDOVÁ, M. Plug and play system for hands on and remote laboratories. In: Proceedings of 8th International Conference on Hands-on Science. Ljubljana: University of Ljubljana, 2011, s. 17-21. ISBN 978-989-95095-7-3.
[10] 1KRBEČEK, M., SCHAUER, F., JAŠEK, R. Security aspects of remote e-laboratories. Zlín: UTB ve Zlíně, Fakulta aplikované informatiky, 2012.
[11] PÁLKA, L., Data Warehouse services [online]. 2013 [cit. 2013-06-26], http://datawarehouse.cz/Data_warehouse
[12] Data Warehouse [online]. 2013 [cit. 2013-06-25], http://www.1keydata.com/datawarehousing/datawarehouse.html
[13] PÁLKA, L., Methods and Tools Related to Data Security and the Protection of Microsoft SQL Servers. Zlín UTB, 2012.
[14] LABERGE, Robert. The Data Warehouse Mentor: Practical Data Warehouse and Business Intelligence Insights. -: 2011. ISBN-10: 0071745327.
[15] CLARKE, J. SQL Injection Attacks and Defense. USA: Elsevier, 2012. ISBN 978-1-59749-963-7.
[16] GERŽA, M., SCHAUER, F., JAŠEK, R. Security of ISES MeasureServer© module for remote experiments against malign attacks, Zlín: UTB ve Zlíně, Fakulta aplikované informatiky, 2013.
[17] BRUCHEZ, R. Microsoft SQL Server 2012 Security Cookbook. UK: Packt Publishing, 2012. ISBN ISBN 978-1-84968-588-7.
[18] HARKINS, M. Managing Risk and Information Security: Protect to Enable. LLC: Apress Media, 2013. ISBN 978-1430251132.
[19] Das, D., Sharma, U., Bhattacharyya, D. K. Rule based detection of SQL injection attack, International Journal of Computer Applications, 2012, vol. 43, no. 19,15—24.
[20] Shang, Y., Luo, W., Xu, S. L-hop percolation on networks with arbitrary degree distributions and its applications, Physical Review E, 2011, vol. 84, no. 3, art. No. 031113.
[21] Tripathi, S., Gupta, B., Almomani, A., Mishra, A., Veluru, S. Hadoop based defense solution to handle distributed denial of service (DDoS) attacks, Journal of Information Security, vol. 4, no. 3, 150—164.
[22] Hamid, R., and Syakirah Afiza Mohammed. 2010. "Remote Access Laboratory System for Material Technology Laboratory Work." In International Conference on Engineering Education and International Conference on Education and Educational Technologies - Proceedings, 311–16. http://www.scopus.com/inward/record.url?eid=2-s2.0-79958730131&partnerID=tZOtx3y1.
[23] Drigas, A. S., Vrettaros, J., Koukianakis, L. G. and Glentzes, J.G. 2006. "A Virtual Lab and E-Learning System for Renewable Energy Sources." WSEAS Transactions on Computers 5 (2): 337–41. http://www.scopus.com/inward/record.url?eid=2-s2.0-33645137921&partnerID=tZOtx3y1.
[24] KRBEČEK, M., SCHAUER, F. and VLČEK, K. "Communication Requirements of Laboratory Management System". In: LATEST TRENDS on SYSTEMS - VOLUME II: Proceedings of the 18th International Conference on Systems (part of CSCC '14), Santorini, Greece, 2014, p. 686-691. ISBN 978-1-61804-244-6, ISSN 1790-5117. http://www.europment.org/library/2014/santorini/bypaper/SYSTEMS/SYSTEMS2-56.pdf
[25] VOŽENÍLEK, V. a STRAKOŠ, V. City logistics dopravní problem města a logistika. 1. vyd. Olomouc: Univerzita Palackého v Olomouci, 2009. ISBN 978-80-244-2317-3.

Lukáš Pálka, Franz Schauer, Karel Vlček and Roman Jašek are with the Tomas Bata University in Zlín, Faculty of Applied Informatics,
Nad Stráněmi 4511, Zlín, CZ- 760 05, Czech Republic (l_palka@fai.utb.cz, fschauer@fai.utb.cz, vlcek@fai.utb.cz, jasek@fai.utb.cz)..
Lucie Váňová, Czech Technical University in Prague, Faculty of Transportation Sciences, Konviktská 20, CZ-111 00 Prague, Czech Republic (lucie.vanova@outlook.com)
Published as resubmitted by the authors 04 October 2014.