Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 55 (2021) 118–125

Transportation Research Procedia

www.elsevier.com/locate/procedia

14th International scientific conference on sustainable, modern and safe transport

Increasing Security of Database During Car Monitoring

Roman Ceresnak, Michal Kvet, Karol Matiasko

University of Zilina, Univerzitná 8215/1, Žilina 010 26, Žilina

Abstract

Nowadays, a vast data flow causes quite a problem in several sectors. One of the issues related to information growth and data volume is their appropriate storage and the securing of the data processing related security and its protection. In this era, where the data are a very sensitive commodity, an attack or data loss can mean losing user trust. Various researchers have dealt precisely with these ideas. They figured out several effective methods of preventing data loss and how to reduce attack amount either on a database or on a web site. Big data, which currently has a significant role in many statistics, are susceptible to various data losses, steal, or deletion. Possible data deletion can lead to a wrong decision about a transport situation prediction task, inaccurate result in decision support, or even to lousy diagnosis prediction in medicine. This vast amount of the problems led to the significant data amount and their security problem solution. Our article relates to data security in situations with substantial real data amount in time, which are entering the system. We suggest effective data distribution methods and a mechanism for data transfer from a database to data storage according to individual records' time parameters.

Peer-review under responsibility of the scientific committee of the TRANSCOM 2021: 14th International scientific conference on sustainable, modern and safe transport

Keywords: Real-time data security; data-warehouse; GPS-monitoring

1. Main text

The data were, are, and always will be an essential part of human lives. Already in the past, the people understood data signification, and they were storing the data in various data storages already in the 50s of the last century. Some from the first storages, keeping the data in a certain way, were different text files, which structure was insufficient and did not perform effective operations such as select, insert, update and delete possible.

Several experts realized these lacks, and in gradual steps, the very first idea of a database was created. This data manipulation method was reflected as useful, and many experts evolved this idea further. Concrete relation databases grew in form, we nowadays now, by gradual work and idea implementation of the several experts.

At the beginning of this century, nonrelational databases came to the fore with social development and massive data growth, defined by 3V (Volume, variety, and velocity [1]. The fast growth of popularity was also related to significant data potential in many ways, to improve various processes in many sectors. Using relational databases as their primary data storage, several companies saw great benefit in data flexibility, which can be stored in nonrelational

Peer-review under responsibility of the scientific committee of the TRANSCOM 2021: 14th International scientific conference on sustainable, modern and safe transport
10.1016/j.trpro.2021.06.013

databases. As the nonrelational database popularity was still growing, several types, where the most common are graph databases, vast column stores, and key-value stores, were created [2].

Security plays the most crucial role in the data field. Data hacking can cause a data leak. This is why society, dealing with data processing, can be highly costly. A hacker can even change or delete the data we got and processed after the big battle. Reasons for the security violation in the database are mainly because of these:

- The majority of the systems for data processing have one security layer.
- None encryption of incorrect data or results/ output data.
- Access to the data of unethical experts in the IT field, which represents the data loss risk.

Database security in the contemporary practice of corporate IT represents an often vulnerable spot. Independent Users Group research (IOUG) showed, 58% of respondents consider the databases as the actual main security weakness. However, while doing so, most investments go to the security on the network level, followed by servers, and the databases are only in third place.

The data stored in the databases used to be valuable, and so, the attacker motivation is vast, lately including ransomware creators; cybernetic crime industry is very innovative. In the case of the vulnerability of abuse through SQL injection, the fact is approached that the attackers can do these actions relatively quickly (without high costs or special technical tools) and search such vulnerable database instances. From another perspective, it is relatively hard for companies to defend themselves against these attacks, regarding database application complexity and the amount of existing code, to audit whole code or eventually replace it. Only 37% of organizations have set some systematic protection against SQL injection. While doing so, the database administrators dispone with standard tools, which allow them to increase background security – for example, database firewalls, which would enable them to watch and filtrate (block) particular SQL demands and further communication between the database and the background.

The security strategy should also protect sensitive data in test and development backgrounds, whose use to be relatively less protected. Practice, when the testing is done, for example, with by no means protected data about costumers and employees from the primary system (tables contain credit cards, personal identification information), is still quite common. Up to 71% of organizations from IOUQ research, while doing so, do not use any data masking techniques that could prevent this – the attackers then, as a result, can also avoid, for example, a robust primary system security.

The matter should also be the proactive access to the database's security and monitoring their operation, capable of pointing out suspicious activity on time; this access could significantly reduce damage caused by ransomware. Even if the tools for watching anomalies during the database operation are available for 48% of organizations, only 32% use them consistently and automatically.

Summarized, nowadays the database security could be a priority for various sectors. By this, it is possible to protect the sensitive data often, also in trespassing of attackers through the perimeter, i.e., the security separating corporate IT from the background. To build up another layer of protection directly on the critical data level, which is the closest to their source, is essential. The detection of SQL injection attacks, guard against them, mapping sensitive data, encryption, and masking while using out of the production background. These various techniques and their correct use (for example, in the form of best practices) can significantly limit the security risks related to the databases.

Based on several studies, we also decided to focus on security data problematics, but in a different way than it was described. We will solve the security already when the data are entering the system to the first data control in our contribution. Except for the very first control, we see the advantages of our article in another 3 points:

- Multistage data control
- Insert reduction with the help of data transferring on the basic of time aspect
- Information transferring for control increase

The rest of the article is structured as follows. The works we examined during the problematics solution, related to the incoming data security in real-time, are included in the Related chapter works. The third section consists of our research and shows us the designed methods and algorithms implemented to improve incoming data security. We created the experiments in the fourth part, which helped us prove the devised method's efficiency and set the trend in further research. The fifth part is devoted to a full article summary and possible improvements of the created processes.

2. Related Work

Whether we talk about the relational or nonrelational database, the database types have their advantages and disadvantages. The characteristics of one or second type have a crucial point in the right database selection. Nowadays, the vast part of companies is transferring to cloud solutions. This recent development in the cloud computing field and distributed web applications has evoked the need to store essential data to distributed databases, providing high availability and scalability. In recent years, an increasing number of companies have adopted various nonrelational database types, commonly called NoSQL databases, and whereas the provided applications appear, they get massive interest in markets. These new database systems are not from the definition "relational," Thus, they do not support SQL's full functionality. Besides, unlike relational databases, they trade with consistency and security for performance and scalability. The security problems become more and more disturbing because more and more sensitive data are stored in the database NoSQL. In contribution [3], the researchers examine two of the most favorite NoSQL databases (Cassandra and MongoDB) and describe their main security elements and problems.

The statements, whose authenticity already was proven in real-time, claim the phenomenon NoSQL took the database world and IT applications by storm. The growth and penetration of NoSQL applications, powered by giants from Silicon Valley such as Facebook, Twitter, Yahoo, and LinkedIn, created an unprecedented database revolution to inspire smaller companies to join the running train of NoSQL. Even if these databases' expansion and growth bring fame and success, it is essential to research the security aspects of these new era databases to various corporate IT sectors. Trust, integrity, and availability (CIA) are the basics of data security and privacy. In article [4], the researchers are dealing with the research and analysis and evaluation of NoSQL development by optics of CIA triad. While the CIA concept is originated in relational databases, it is crucial to understand, examine, and define the security possibilities of this new generation database from a fulfilling CIA perspective.

The security models developed for the databases differ in various aspects because they are focused on different characteristics of the security database problem or because they create different assumptions about what represents the security database. This leads to an incoherent and incomplete understanding of an organization's security strategy. This makes it harder for the harmonization of various security demands. In article [5], the researchers examine various security problems in the databases. The report is useful for planning precise and directive-based security demands on the database.

We noticed a study while examining the problematics that have a different point of view opposite publicized articles. The study provides a multistage secured database system. The modeling of the application domain data's semantics has been the research subject for many years in the database community. However, these afford only to solve the data integrity characteristics. The researcher's contribution in the article [6] demonstrates the extended data model's use, representing the integrity and aspects of data secrecy. This modeling technique is possible to ause as a tool for the design of the database and, what is more important, as a tool for domain experts, database designers, and security service workers, to precisely define security demands for the application domain. The second contribution of their contribution is a complicated taxonomy of the security data semantics that has to be caught and understood to implement the multistage security automatized information system.

Today, popular distributed data processing was not unnoticed, even with security. While the research of origin is standard in the distributed systems, many suggested solutions do not deal with system security and responsibility for the data stored in these systems. In contribution [7], the researchers examine preventive solutions designed for problem-solving with the system security and responsibility for the distributed systems' data. We deduce the file of minimal demands from our research, which are needed to make the provenience system more effective in solving these two problems. In conclusion, we identify some spaces in examined solutions and introduce them as challenges that should be reviewed by future research workers of provenience. The authors claim these spaces are necessary to be solved sooner than possible to reach a full and reliable provenience solution.

Several data damages can happen during various attacks, whether it is a web application or a direct attack on the database. The central aspect of the majority of companies is the reliability of data correctness in the system. When a user inserts the records into the database, they await they will find them in the database any time. While the attack data edit, respectively, data delete can happen. The researchers in the article [8] dealt with these problematics, where they examine solutions when the data were always actualized synchronously in precisely set time points, frequencies, and granularity. The researchers state, it is necessary to make a solution appropriate to the concrete system to achieve

the best performance. They refer to the attribute oriented temporary model with a reflection for technologies of data grouping. When the object is defined only partly, or n of the data, the case is not present at all in many times. Because of these reasons, the values stored in the database in the form of very time or attribute, representing the object state, have to be defined. Publicized contribution deals with time-oriented database architectures, manages undefined values and designs complex system classification based on transactions, accesses, and indexes. They deal with the techniques of undefined values modelling and cover synchronized processes with data groups' help. The researchers designed the solutions, useful for the data getting with emphasis on undefined values and states.

The data security shared on various servers is a difficult problem in non-relational databases because the data distributed and transferred through an unsecured network. The extensive research of the mechanism of NoSQL database division was done, but no concrete criteria for analysis of divided architecture security were defined. The researchers in the contribution [9] design valuation criteria. The submitted analysis helps various organizations' inappropriate and reliable database selection by their preferences and security demands.

While examining the security, the scientists examined the solving of the problematics with trend technology blockchain. The researchers in the document [10] design to use a distributed scheme based on blockchain-knows also as public accounting book – to create VWN, where the owners of primary wireless sources (PWRO) rent their wireless heads (for example, part of the high-frequency spectrum, infrastructure) to mobile virtual servers by network operators (MVNO). Researcher using communication between machines based on agreements about service level (SLA) between PWRO and MVNO. Designed distributed scheme based on the blockchain provides the security to include PWRO and MVNO and prevent PWRO from excessive use of their sources (what will stop double costs) and help MVNO fulfill QoS demands of their users. Federal communication commission USA (FCC) or similar regulation organs in other countries participate in this framework by providing the instructions and prescriptions about maximal power supply levels, licenses, and geographical coverage. Essentially, this helps users to fulfill demanded QoS demands while keeping principles of government regulation. Performance is rated with the help of number results.

As it is possible to see, several studies were dealing with security problematics. While designing our new method, we were inspired by the work [11]. The author focused on granularity management of the temporary system and developed data sharing models based on reliability, sensitivity, and accuracy of data providers. A new conception of a time advantage is introduced, which is subsequently evaluated in the experimental part. Data flow optimization by historical data aggregation, and data amount limitation is a crucial part of the system's decision-making process, while the time for data transfer is strictly limited.

We had to use a tool to transfer the data from the database to the database storage for data migration purposes. We examined the method, because of this purpose, that is currently the most widespread. In their article [12], the researchers described the ETL method as a process of data storing, migrating the data from the source database by performing specific rules of transformation over extracted data. This transformed data is loaded back to the target database. Various organizations with significant data occurrences lead to big data technologies such as Hadoop, Hadoop Ecosystem Project such as Hive and HBase to store their data. An organization uses the data migration process for the data migration from RDBMS to Hadoop, and these data are further used for various analytic purposes. However, the data migration role sometimes causes irregularities in the data because of multiple reasons, which can lead to inaccurate data analysis. This contribution talks about a generalization framework for data verification between RDMBS and Hadoop.

3. Our contribution

The chapter of our contribution shows two methods we needed to implement to increase the data's security entering the system.

3.1. Data adjustment

The data coming to the system from every car are always edited. Every car that sends the information about a position with the help of a watcher has only three values. The first value is a unique car value, the second value is an x coordinate on the map, and the third is a y coordinate on the map. The big problem, regarding the attack, is seen in changes of individual data and their next insert to the database. Thus, we added another parameter to the table

displayed in picture 1. They are the time of addition, respectively edit of the data, and the unique record value. The result table is showed in picture 2.

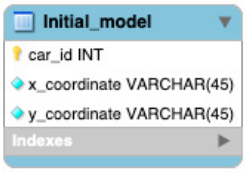

Fig. 1. Base model

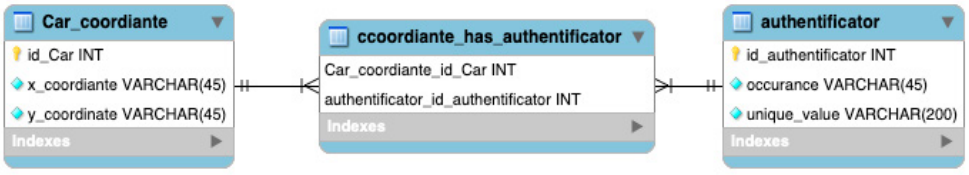

Fig. 2. Edited model

We inserted the values we recorded into the relational database MySql. Even if we also tried the nonrelational database DynamoDB based on firm structure, we decided to use the right relational database.

The values, entering the tables, are allowed only for a concrete application that monitors individual transport vehicles' roads. In our case, they are cars. Because of this purpose, we created a role, concretely IAM Role, allowing us to insert the records only to the application.

The main idea of the table extension was the control of the operations state of individual records. We think about operations update and delete. Because of this purpose, we created a notification above the database. We always want every operation *update* and *delete* to be controlled before the individual operations performance. If the application performs the mentioned operations *update* and *delete*, their role is handled at first, assigned to the application. Subsequently, the unique value from the table authenticator is controlled. Five random tokens are created for every role, which head is sent automatically while calling the operation update and delete. These five randomly created tokens are also dependent on record creation, and their automatic edit happens every 12 hours. The token improvements work until an unauthorized attempt of the data edition will happen. If this situation occurs, the tokens are regenerated automatically, and the change for creating a new role for the application is stimulated. The notification edit is the second obvious thing performed automatically during the unauthorized attempt to get the data.

```
aws rds create-event-subscription \
    --subscription-name myeventsubscription \
    --sns-topic-arn arn:aws:sns:us-east-1:802#########:myawsuser-RDS \
    --enabled

aws rds delete-event-subscription --subscription-name myrdssubscription

aws rds remove-source-identifier-from-subscription \
    --subscription-name myrdseventsubscription \
    --source-identifier mysqldb
```

3.2. Data transfer

In our opinion, the data are the most sensitive to be attacked on the database level, so we decided to protect this data with their transfer to data storage. This transfer is done automatically based on the latest changes with the data. If the value of the data existing in the database breaks the value of 1 month, so the data are automatically transferred in the following way:

- Script car_coordinate above table is stimulated, looking as follows
- *mysql> SELECT * FROM car_coordinate JOIN ccoordinate_has_authenticator USING id_car JOIN authenticator USING (id_authenitcator) WHERE DATE_SUB((DATE_SUB(curdate(), INTERVAL 1 MONTH)). LIMIT 0,1000000 INTO OUTFILE '/tmr,/car_coordinatel.csv FIELDS TERMINATED BY ' ENCLOSED BY '"' LINES TERMINATED BY '*
- Subsequently, the procedure is stimulated
- *mysql> call export_csv_split('tbl_product',1000000);*

- The generated files CVS transfer is done gradually to address book with the name s3-redshift:
- *mkdir ~/s3-redshift*
- *mv /tmp/*.csv ~/s3-redshift/*
- Because the running instance has to have the right to start and manipulate with S3, so we had to install s3cmd command line for these purposes
- *rpm -Uhv http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch.rpm*
- Subsequently, we created a new bucket for Redshift
- *s3cmd mb s3://s3-rshift*
- After the bucket creation, we started to synchronize data address book CSV to bucket S3:
- *s3cmd sync s3-redshift s3://s3rshift*
- In the next steps, we prepared our Redshift.
- Redshift runs as the version on PostgreSQL 8X. For access to cluster Redshift with the provided ending point and credentials, we can use standard client PostgreSQL..
- We used the following command for the installation of the command PostgreSQL.
- *yum install -y postgresql #Redhat/CentOS*
- *sudo apt-get install -y postgresql #Debian*
- We used program psql for access to cluster Redshift.
- psql --host=mysql-redshift.cd23imcbfhbd.ap-southeeast-1.redshift.amazon.com --port=5
- Subsequently, we created a table similar to the table in MySQL, which script looks as follow:*CREATE TABLE car_coordinate (car_id INT PRIMARY KEY NOT NULL, x_coordinate VARCAHR2(45) NOT NULL, y_coordinate VARCAHR2(45) NOT NULL)*
- Then, command COPY was started to get access to CSV files in our segment S3 and parallelly loaded them to the table.
- copy *car_coordinate FROM ‚s3://s3-rshift/s3-redshift/ car_coordinate ‘ credentials ‚aws_access_key_redshift‘*
- After the very first loading of the data from segment S3, we have to start with the command VACUUM to arrange your data and analyze commands for table statistic actualization.
- *vacuum car_coordinate*
- *analyze table_product*
- After we did every step, the table structure verification happened.
- *analytical=# \d car_coordiante;*
- *TABLE “public.car_coordinate”*

After applying all steps, which help us transfer the data from the relational database MySql to data storage Amazon Redshift, the deletion of shared records from the tables *car_coordinate*, *authenticator*, and subsequently *ccar_coordinate_has_authetificator* happened. The main goal was to reduce the database overload and reduce the possible attack on huge data in the database that monitors car position on the map.

3.3. Experimenty

We created an application for the experimental activity, portrayed in picture 3. The developed application monitors the cars delivering the goods around the whole of England.

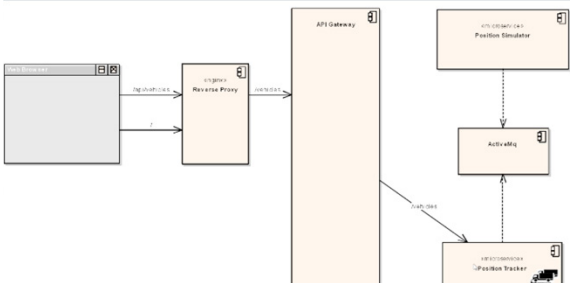

Fig. 3. Architecture of application

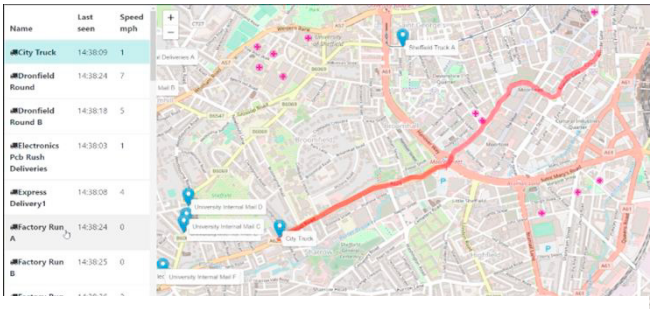

Fig. 4. GPS tracking

Approximately 10 000 records were inserted into the system for up to 10 minutes. During the experimental activity with the cars' help situated in the system. Even if some positions already were in the design, what we solved in another contribution, so the database worked effectively and effectively process the incoming data. The system on which we deployed the application and database is a cloud from Amazon company. Any performance lack did not happen ever, because calculation units EC2 were scaled by automatic scaling by service Amazon Auto Scaling Group.

We tried to send the same demand as the application sent by various applications and services during the experiment about the value edit, with afford to get, respectively to edit the data. To afford to prove our solution, we copied the access login token that was already supposed to secure access to the data during the ordinary operation. This experiment has a more real contour to the error edit, but the designed method controlled the application authenticity and evaluated our investigation as invalid. The demand was automatically deleted.

We sent the same condition during the turn-off of our control method, and in that case, the data were changed. Our second method, which transfers the data automatically, based on the time range, from relational database MySQL to data storage Redshift, caused only a small number of the records to be edited and other mentioned values stayed protected.

4. Conclusion

Current significantly growing transport demands sophisticated solutions to minimalize the data loss, undesirable value edit by third sides, and also attack reduction for the application or the database with the sensitive data. The number of attacks on sensitive data forces several companies to deal with the system's security in more extensive measures and emphasize the effective way of data manipulation in the application. The methods mentioned in the past and considered sufficient for sensitive data security are either obsolete or unused because of insufficient protection against various attack types. As our designed process proves, the effective way of how to avoid data loss or to reduce unauthorized access is data control based on the time aspect.

In our designed solution, we use value addiction with every incoming value from every car. By the addition of a timestamp, we subsequently control the authenticity of every record. This value is not unique for every vehicle, so we created the structure to which only the system alone has access and is used to verification record authenticity. We also made the information transfer method from the database to the data storage based on the time spectrum, which helps reduce the security risk and reduce the overload of primary data storage because the data storage is the alpha and omega to us. We reduced the surplus of primary data storage by approximately about 34% every month based on the designed method. Our planned approach allowed us to catch 13% of intentional data change whose were not approved for a given shift.

We will deal with concrete modules in the next development for a more complex solution, such as monitoring several cars used in the experimental part. Nowadays, where several data storages are existing, we would want to solve more complex modules that could work with our designed method in another background, such as cloud background from Amazon. On the other hand, to accelerate the control module of the timestamp with the database's data could be demanded by comparing the control values stored in the system.

Acknowledgements

The work is also supported by the project VEGA 1/0089/19 Data analysis methods and decisions support tools for service systems supporting electric vehicles and Grant system UNIZA.

References

E. Hofmann, “Big data and supply chain decisions: the impact of volume, variety and velocity properties on the bullwhip effect,” *Int. J. Prod. Res.*, 2017.

A. Gupta, S. Tyagi, N. Panwar, S. Sachdeva, and U. Saxena, “NoSQL databases: Critical analysis and comparison,” in *2017 International Conference on Computing and Communication Technologies for Smart Nation, IC3TSN 2017*, 2018.

L. Okman, N. Gal-Oz, Y. Gonen, E. Gudes, and J. Abramov, “Security issues in NoSQL databases,” in *Proc. 10th IEEE Int. Conf. on Trust, Security and Privacy in Computing and Communications, TrustCom 2011, 8th IEEE Int. Conf. on Embedded Software and Systems, ICESS 2011, 6th Int. Conf. on FCST 2011*, 2011.

S. Srinivas and A. Nair, “Security maturity in NoSQL databases - Are they secure enough to haul the modern IT applications?,” in *2015 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2015*, 2015.

S. Imran and I. Hyder, “Security issues in databases,” in *2009 2nd International Conference on Future Information Technology and Management Engineering, FITME 2009*, 2009.

G. W. Smith, “Modeling security-relevant data semantics,” *IEEE Trans. Softw. Eng.*, vol. 17, no. 11, pp. 1195–1203, 1991.

Y. S. Tan, R. K. L. Ko, and G. Holmes, “Security and data accountability in distributed systems: A provenance survey,” in *Proceedings - 2013 IEEE International Conference on High Performance Computing and Communications, HPCC 2013 and 2013 IEEE International Conference on Embedded and Ubiquitous Computing, EUC 2013*, 2014.

M. Kvet, Š. Toth, and E. Krsak, “Concept of temporal data retrieval: Undefined value management,” *Concurr. Comput. Pract. Exp.*, vol. 32, p. e5399, Jun. 2019.

A. Zahid, R. Masood, and M. A. Shibli, “Security of sharded NoSQL databases: A comparative analysis,” in *Conference Proceedings - 2014 Conference on Information Assurance and Cyber Security, CIACS 2014*, 2014.

[D. B. Rawat and A. Alshaikhi, “Leveraging Distributed Blockchain-based Scheme for Wireless Network Virtualization with Security and QoS Constraints,” in *2018 International Conference on Computing, Networking and Communications, ICNC 2018*, 2018.

M. Kvet, “Data Distribution in Ad-hoc Transport Network,” in *2019 International Conference on Information and Digital Technologies (IDT)*, 2019, pp. 275–282.

K. Sharma and V. Attar, “Generalized Big Data Test Framework for ETL migration,” in *International Conference on Computing, Analytics and Security Trends, CAST 2016*, 2017.