АҚПАРАТТЫҚ-КОММУНИКАЦИЯЛЫҚ ТЕХНОЛОГИЯЛАР
ИНФОРМАЦИОННО-КОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ
INFORMATION AND COMMUNICATION TECHNOLOGIES

АҚПАРАТТЫҚ ҚАУІПСІЗДІК. ДЕРЕКТЕРДІ ҚОРҒАУ
ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ. ЗАЩИТА ИНФОРМАЦИИ
INFORMATION SECURITY. INFORMATION PROTECTION

DOI 10.51885/1561-4212_2024_3_165
IRSTI 81.93.29

G.E. Mukasheva[1], A.J. Karipzhanova[1], Stanio Kolev[2], Zh.Zh. Akhmetova[3], G.A. Uskenbayeva[3]
[1]EI «Alikhan Bokeikhan University», Semey, Kazakhstan
*E-mail: gulzira_7777@mail.ru**
E-mail: kamilakz2001@mail.ru
[2]Technical university of Sofia, Bulgaria
E-mail: skolev@tu-sofia.bg
[3]L.N. Gumilyov ENU, Astana, Kazakhstan
E-mail: zaigura@mail.ru
E-mail: gulzhum_01@mail.ru

ENSURING THE PROTECTION OF DATABASES FROM THREATS

МӘЛІМЕТТЕР ҚОРЫН ҚАУІПТЕРДЕН ҚОРҒАУДЫ ҚАМТАМАСЫЗ ЕТУ

ОБЕСПЕЧЕНИЕ ЗАЩИТЫ БАЗ ДАННЫХ ОТ УГРОЗ

Abstract. *The article is devoted to the substantiation of one of the directions of scientific research of the authors – the problems of database security. These studies are aimed at developing and testing new solutions, in particular, providing error correction and partial losses using multidimensional parity algorithms when splitting files. A review of the state of methods and algorithms for the functioning of distributed databases reveals one of the main aspects of the DD security problem - the reliability of information storage. With the increase in physical storage capacity to secure backup storage and the consequent increase in cost and power consumption, a security technology that uses multivariate parity algorithms that are robust to partial loss of storage locations, allowing multiple disks to fail even simultaneously, is an alternative security option.*

Keywords: *databases, security, distributed storage, splitting, replication, multidimensional parity.*

Аңдатпа*. Мақала авторлардың ғылыми зерттеулерінің бір бағыты – мәліметтер қорын қорғау мәселесін негіздеуге арналған. Бұл зерттеулер жаңа шешімдерді әзірлеуге және сынауға бағытталған, атап айтқанда, файлдарды бөлу үшін көп өлшемді нақты алгоритмдерін пайдалана отырып, қателер мен ішінара жоғалтуларды түзетуді қамтамасыз етеді. Бөлінген мәліметтер қорының жұмыс істеу әдістері мен алгоритмдерінің жағдайын шолу ТМҚ қауіпсіздік мәселесінің негізгі аспектілерінің бірі – ақпаратты сақтаудың сенімділігін көрсетеді. Резервтік сақтаумен қауіпсіздікті қамтамасыз ету үшін қоймалардың физикалық көлемін ұлғайту және тиісінше олардың құны мен энергия шығынын ұлғайту жағдайында сақтау орындарының ішінара жоғалуына төзімді көп өлшемді нақты алгоритмдерді қолдана отырып, бірнеше дискілердің істен шығуына мүмкіндік беретін қауіпсіздік технологиясы, тіпті бір уақытта да, қауіпсіздікті қамтамасыз етудің балама нұсқасы болып табылады.*

Түйін сөздер: *мәліметтер қоры, қауіпсіздік, таратылған сақтау орыны, бөліну, репликация, көпөлшемді нақтылық.*

Аннотация. *Статья посвящена обоснованию одного из направлений научных исследований авторов – проблемы безопасности баз данных. Эти исследования нацелены на разработку и*

апробацию новых решений, в частности, обеспечение исправления ошибок и частичных потерь с использованием алгоритмов многомерной четности при расщеплении файлов. Обзор состояния методов и алгоритмов функционирования распределенных баз данных выявляет один из главных аспектов проблемы безопасности РБД – надежность хранения информации. В условиях увеличения физического объема хранилищ для обеспечения безопасности за счет резервного хранения и соответственно, повышения их стоимости и энергопотребления, в качестве альтернативного варианта безопасности выступает технология защиты с использованием алгоритмов многомерной четности, устойчивых к частичной потере мест хранения, допускающих выход из строя нескольких дисков, даже одновременно.

Ключевые слова: *базы данных, безопасность, распределенное хранение, расщепление, репликация, многомерная четность.*

Introduction. The concept of a distributed database (DDB) has remained fundamental and unchanged for many years: a DDB is understood as a collection of logically related databases distributed in a specific network (Dejt, 2019). At the same time, the RDB includes parts of various information databases, which are located on network nodes and are controlled, in fact, by arbitrary operating control systems with their own software (Yocy, Val'duries, 2021).

The threat hierarchy for distributed databases includes many different sources of danger. These threats can come from both internal and external sources and may be aimed at the confidentiality, integrity or availability of data. Here are some of the main threats:

1. External threats:

– Physical threats – these can be theft, vandalism or physical access to data.

– Attacks on network communications - such as eavesdropping, interception or denial of service.

– Threats from intruders - it can be phishing, social engineering or other forms of fraud.

2. Internal threats:

– Software bugs – these may be vulnerabilities in the code or security issues.

– Human factor – this may include errors by users, administrators or developers.

– Incorrect system configuration - for example, weak passwords, insufficient authentication or lack of access control.

The key security problem of distributed databases is to ensure their protection from these threats. This requires a comprehensive approach to security, which includes measures to prevent, detect and respond to threats.

Materials and methods of research.

An assessment of the current state of RDB can be made on the basis of traditional principles known as Data rules, put forward back in 1987. There are twelve of them:

1) local autonomy;
2) independence of nodes;
3) continuous operations;
4) transparency of location;
5) transparent fragmentation;
6) transparent replication;
7) processing distributed requests;
8) processing of distributed transactions;
9) independence from equipment;
10) independence from operating systems;
11) network transparency;
12) independence from databases (Dejt, 2019).

According to these rules, the function must obey the apodictic principle: for any user, a distributed system must look like a regular, undistributed one. The point is that users do not feel the difference between distributed and centralized databases.

A review of methods and algorithms for constructions in the Data rules format reveals the following key problems.

1) The requirement of local autonomy means that all nodes in a distributed system must be autonomous, and data regulation on each node of the system is carried out to a limited extent. That is, the database on any node remains an integral element of the distributed system, but acts as a local database and is also managed locally. Problems in such a constellation can only arise when selecting a (local) database management system (DBMS) for a distributed database (RDBMS), which must ensure management of the RDB and transparency of its use for all clients (Dejt, 2019). The best option is a federated RDBMS, based on the technology of "federated" access to various databases (Adéle, 2015). The federated option is reliable, but sometimes associated with implementation difficulties. Problems can arise not only due to differences in supported data models, but also, for example, due to simple coincidence of field names. Modern software solutions make it possible today to overcome difficulties, but relatively low performance due to time restrictions on accessing data in a centralized database, as well as the constant growth in the volume of information, keep the desire to build an ideal RDBMS relevant.

2) The requirement for node independence is due to the fact that the RDB should not have a single node without which the system cannot function. This means, among other things, the absence of a supporting central node. Difficulties in implementing the concept of node independence may be associated with access to node data, as in the case of the requirement of local autonomy. The problem is solved in the global schema format, based on which users can build distributed queries and update data. In this case, the federated DBMS works only with the common data schema, since all local DBMSs have their own data schemas and provide data access to all users by their own means.

3) The requirement for continuous operation is associated with the need to ensure continuous access to data within a single database, regardless of the location of nodes, and the performance of local operations even for administrative needs (Dejt, 2019). Continuous work of users is carried out thanks to applications - local, which do not require access to data located on other nodes, and global, which require such access (global applications) (Atamanov, 2017). Difficulties in continuous operation generally arise from problems of shared access to common data. There are two approaches to solving problems: establishing locks and managing transaction concurrency. Often both approaches are used together. But interleaving operations can lead to incorrect results, which will compromise the integrity and consistency of the database. Therefore, in the first approach, a locking mechanism is used to organize multi-user access to the DBMS. Establishing locks and managing transaction concurrency does not solve all the problems of continuous operation. Distributed deadlocks are always possible when transactions occur on a system that attempt to access locked data on another system.

4) The requirement of location independence means that, firstly, the user must access the database from any of the nodes, and secondly, absolute clarity (transparency) of the placement of all data is guaranteed. Difficulties may be related to optimization problems when access is performed over the network and it is not clear how the network transmission delay should be taken into account when optimizing a request, how to take network parameters into account, etc. These problems are solved in the current mode.

5) The fragmentation independence rule means that the user must access data regardless of the method of fragmentation, which is one of the two main ways to organize a distributed storage database (the other is replication). Both methods are considered the main means of ensuring the security and integrity of information in the RDB (Mamonov, 2018).

6) The rule of independence from replication (replication) means that the user's work with the RDB should not depend on replication procedures. With full replication, synchronized

copies of the same database are placed on all computers. Data security in such a system will be the highest. The disadvantage is the complexity of replica synchronization during data updates and the fact that between updates the database copies may differ from each other (Kurmanbaev, Syrgabekov & Zadauly, 2017). Moreover, with the growth of information volumes, problems with the physical placement of growing hardware complexes and their energy supply with growing financial costs come to the fore.

7) The allocation of a separate rule for processing distributed queries means that the RDBMS must smoothly and error-free support the processing of queries that reference data located on more than one node. This is one of the main tasks of a distributed database management system (Karipzhanova, Sagindykov & Dimitrov, 2019). The ability to execute a distributed query is now supported by almost all server DBMSs.

8) Distributed transaction processing is an integral function of an RDBMS, which must support the execution of a transaction as a recovery unit. Problems that can arise when executing transactions in parallel include lost update results, uncommitted dependencies, and incompatible analysis. Loss of update results occurs when, for example, several transactions write data to one tuple, the last update is committed, the rest are lost. Methods of solution, as well as the causes of deadlock situations and methods for resolving them are known.

9) The rule of independence from the type of equipment is due to the fact that the RDBMS must be able to function on equipment with various, almost any, computing platforms. This is one of the most successfully addressed areas of RDB development.

10) The requirement of independence from the operating system implies that the RDBMS must be ready to operate under the control of various operating systems. Thanks to modern solutions, RDBs operating at the network level do not depend on the operating systems installed on the network, since they operate on network traffic exchanged by all network nodes.

11) Network transparency practically means the independence of the RDB from the network architecture. The RDBMS must function on networks with different architectures and media types. In fact, this category of requirements concerns the essence of the distributed database itself, since access to RDB data is always implemented over the network.

12) The requirement of independence from databases actually means independence from the type of DBMS. In this case, the RDBMS must interact with local DBMSs with different (heterogeneous) data types. The solution to this problem leads to the fact that DBMSs from different manufacturers successfully coexist in RDBs.

This brief review of the state of the methods and algorithms for the functioning of RDB reveals the validity and consistency of the distributed database paradigm, which implies providing flexible forms of servicing a large number of remote users and working with large flows of information in conditions of spatial and structural isolation. However, some fundamental shortcomings of the RDB appear to this day, although they are solved promptly in accordance with the autonomy of the tasks. Nevertheless, the promise of RDBS as a subtype of distributed computing systems involved in data processing is clear. Two advantages are crucial:

1) high power of the distributed system in solving common tasks;

2) possibility of autonomous operation of separate elements of the system.

It is still believed that the advantages include economic benefits, as well as increased reliability, data availability, and productivity (Dejt, 2019; Yocy, Val'duries, 2021). However, the increasing complexity of security threats that are constantly growing forces us to reconsider this assessment, since the current security paradigm is based on backup storage, requiring continuous replication using an ever-increasing number of storage devices. This inevitably leads to an increase in the cost of operating the RDB, a decrease in the reliability and availability of

data. Therefore, one of the main directions of development and operation of RDB today is a qualitative improvement in the security of distributed storage.

Currently, there are several types of relatively reliable data storage systems (DSS). They differ primarily in the way storage devices are connected to the server.

Traditionally, the storage system is connected directly to the server. This technology is called Server Attached Storage (SAS) or Direct Attached Storage (DAS).

In addition, there are two more common technologies that involve connecting storage systems to a network (Network Attached Storage, NAS) or creating a special dedicated network that combines storage systems with application servers (Storage area Network, SAN) (Figure 1).

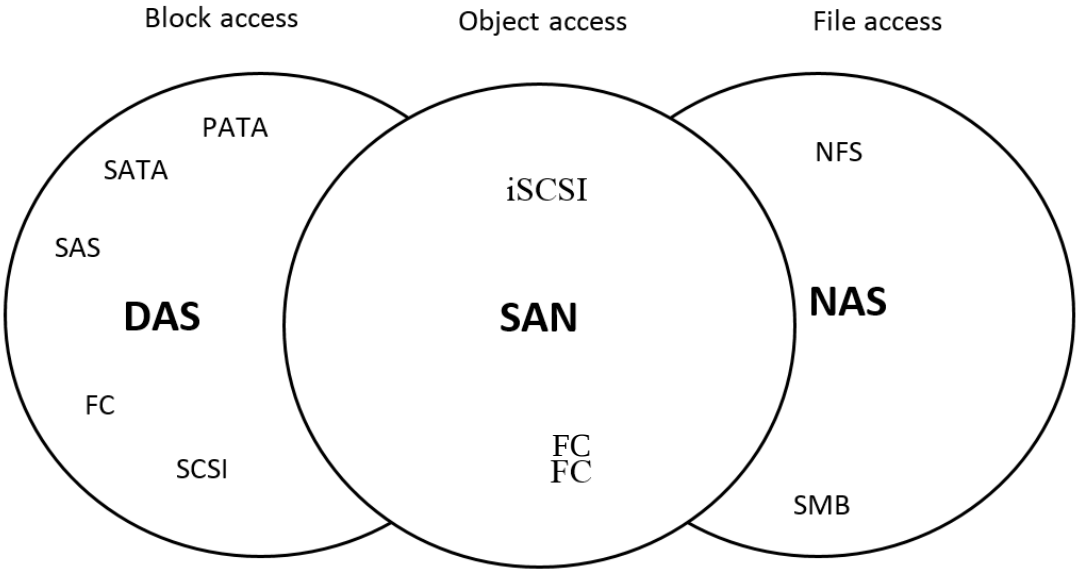

Figure 1. Levels of data storage systems

DAS storage systems that connect directly to the server's high-speed interface have been developed to expand the storage capacity of existing servers (Omel'yanenko, Papinashvili, 2017). One or more devices are connected to a common server using one or more interfaces, which may include SCSI, RAID, Fiber Channel and others. Today, the DAS architecture occupies a leading position in the market and is used in most storage systems. Disadvantages of the model: insufficient manageability; growing costs of data storage; overloading of the local network when processing large amounts of data; physical limitations on the total capacity of connected storage devices imposed by the bus architecture (SCSI, FC and others); dependence of storage systems on the performance of the server to which they are connected. linked (Syrgabekov, Zadauly & Kurmanbaev, 2017).

Network Attached Storage (NAS) is a high-performance network architecture with a special hardware platform in which the hardware and software performs the function of a file server (Grigor'ev, Pluzhnikov, 2011). The main goal pursued by developers is to simplify file sharing compared to the DAS architecture. Physically, NAS - devices are equipment that is connected directly to a local network (most often Ethernet). The advantages of NAS architecture are cost reduction by placing data within one logical structure and simplicity (relative to DAS) of storage management. NAS is a transition architecture between DAS and SAN.

A Storage Area Network (SAN) is an additional dedicated data network that connects one or more application servers to one or more storage systems (Omel'yanenko, Papinashvili, 2017).

The basis of SAN is the ability to connect any of the servers to any data storage device. RAID arrays, libraries based on tape, magneto-optical and other types of devices, as well as simple disk arrays without RAID capabilities can be used as storage devices. SAN systems are used to work with large databases. A significant disadvantage of SAN is its high cost.

The systems considered today cannot guarantee 100% reliability of data storage, and this is one of the main aspects of the problem of RDB security, the most pressing direction for today in the development and functioning of methods and algorithms for creating RDBs (Karipzhanova, Sagindykov, 2019). The problem of RDB reliability is becoming more and more relevant with the sharp increase in the volume of stored information. The term Big Data characterizes a trend that becomes apparent after becoming familiar with numerous cases of unauthorized penetration of databases with large-scale leaks (see, for example, (Korneev, 2019).

The main factor influencing the reliability of data storage is the hardware reliability of storage devices, which is determined only by the existing level of production and is indicated in the equipment specification. Any fact of device failure is a deterministic phenomenon. But incomplete information about the processes occurring in it and the environment leads to the probabilistic nature of failures, i.e., failure can be caused by various reasons and have a different nature and nature. If we take into account that the time of occurrence of a failure is a random value, then the probability of this event can be estimated using the methods of mathematical statistics and the theory of random processes.

A random variable can be characterized by a distribution law in the form of a failure probability distribution function:

$$P(t) = P(T < t), \tag{1}$$

where T is the mean time between failures.
The density distribution of failure probability will be:

$$p(t) = \frac{dP(t)}{dt} \tag{2}$$

The distribution of a random positive variable is called exponential if its probability density function has the form:

$$p(t) = \lambda e^{-\lambda t}, \quad t \geq 0 \tag{3}$$

Exponential distribution is used when considering sudden failures, when the phenomena of wear and aging are so weakly expressed that they can be neglected. In hard drives, mechanical wear factors are practically reduced to zero and are determined only by the reliability of electronic components.

In the first approximation, the time to failure of electronic components obeys an exponential distribution. In the case of intensive operation under heavy load, the effects of slow degradation may manifest themselves. A more accurate model of the failure probability distribution in this case will be the parametric Weibull distribution, which describes the slow degradation of the functional parameter y over time t. Parametric reliability characterizes the ability of an electronic device to maintain the level of the functional parameter $y(t)$ within the established standards from a to b for a given time T. The level of parametric reliability is determined by the probability of failure within a given period of operating time T:

$$P(T) = P(a \leq y(t) \leq b,\ t \leq T) \tag{4}$$

In general, the Weibull distribution for a random variable x is given by the density $fx(x)$

(Borovikov, 2011):

$$f_x(x) = \begin{cases} \frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} e^{-\left(\frac{x}{\lambda}\right)^k}, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (5)$$

If we take x as the mean time between failures, we find that the failure rate is proportional to time:

$k < 1$, the failure rate decreases over time;

$k = 1$, the failure rate does not change over time;

$k > 1$, the failure rate increases with time.

Here the coefficient k is the Weibull modulus.

Parametric distribution of the conditional Weibull reliability function:

If we take x as the mean time between failures, we find that the failure rate is proportional to time:

$$R(t|T) = \frac{R(T+t)}{R(T)} = \frac{e^{-\left(\frac{T+t}{\lambda}\right)^k}}{e^{-\left(\frac{T}{\lambda}\right)^k}} \quad (6)$$

or

$$R(t|T) = e^{-\left[\left(\frac{T+t}{\lambda}\right)^k - \left(\frac{T}{\lambda}\right)^k\right]} \quad (7)$$

R(t|T) – shows the probability that the object will work for another t time, provided that it has already worked for T time.

The technical specifications for hard drives used in storage devices usually indicate the following reliability parameters:

1) MTBF (Mean time between failures). MTBF, measured in hours, means the conditional average number of hours of disk operation before the first failure. Typical MTBF values for Desktop-class disks range from 0.5 to 1.0 million hours, and for Enterprise they can reach up to 2 million (Chen, Zhao, 2012).

2) AFR (annual failure rate). This is a more useful parameter than MTBF, which is the annual failure rate measured as a percentage. Typical value ≈1%, which is equivalent to MTBF = 0.88 million hours before first failure. For example, Western Digital claims an AFR during the warranty period of less than 0.8%, which equates to an MTBF < 1,095,000 hours.

3) UER (unrecoverable error rate) – frequency of occurrence of an unrecoverable error. This is a technological parameter that depends on the quality of the magnetic coating. Such a faulty bit is also called a «rotten bit» (bit rot). The typical value for Desktop disks at the moment is ~10-14, i.e., one erroneous bit can appear on a stream of 1014 bits, and for Enterprise class disks 10-15-10-16

The obvious way to increase the reliability of data storage in the RDB database by increasing the reliability of hard disks seems to be the most simple and acceptable method. However, it is connected with the level of development of hard disk production technology, at which there is a certain limit, above which it is impossible to rise in principle. It is necessary to increase the reliability of elements so that the probability of failure-free operation of the system corresponds to the specified requirements. Practical realization of such highly reliable elements is not always possible. As the growth rate of information volumes in RDBS increases, the problems of hardware reliability of individual storage devices are added to the problems of reliability

assurance in multi-component systems - the effects related to storage scaling.

To increase reliability, special technologies are used to achieve high storage reliability. First of all, these are hardware backup and data replication technologies. In the process of recording, data is simultaneously replicated (copied) to several storage locations. Replication is characterized by the replication coefficient R, which is equal to the number of written copies. The most common replication variant is, for example, mirroring - parallel writing of data to two locations simultaneously, with replication coefficient R=2.0. In «Big data» technologies, when arrays of storage locations are used, coefficients less than R=3.0 are almost never used. The default replication factor is 3.0, for example, in Apache Hadoop and OpenStack Swift object storage systems, Red Hat Ceph distributed file system, etc. In practice, this parameter is adjusted depending on the reliability requirements and can reach R=16.0-32.0 (Karipzhanova, Sagindykov, 2018).

Let us estimate the probability of failure of redundant systems. If the probability of failure of a single disk is equal to *Pdev*, then the probability of failure *Parr* of an array of n disks is equal to the joint probability of failure of n disks, i.e., all disks must fail at the same time:

$$P_{arr} = P_{dev,1} \cdot P_{dev,2} \cdot \ldots \cdot P_{dev,n} = \prod_{i=1}^{n} P_{dev,i} \quad (8)$$

If identical disks are used, then for the replication coefficient *R*=*n* we get:

$$P_{arr} = P_{dev}^{n} \quad (9)$$

When using, for example, disks with AFR=1.0%, with a probability of disk failure within a year Pdev=0.01, mirroring will reduce the probability of failure to Parr=10-4 (0.01%), and with a replication factor of 3, 0 the probability of information loss in such a storage will decrease to Parr=10-6 (0.0001%).

Redundancy is a very effective way to increase reliability, but at the same time it is too wasteful. As the replication coefficient increases, redundancy increases by the same factor. If we need to write *D* bits of data, then when writing to a storage with a replication coefficient of R=n, *Darr* bits will actually be written, which is *n* times more:

$$D_{arr} = D_{dev,1} + D_{dev,2} + \cdots + D_{dev,n} = \sum_{i=1}^{n} D_{dev,i} \quad (10)$$

$$D_{arr} = nD_{dev} \quad (11)$$

When storing and transmitting data, errors and/or loss of information inevitably occur. To increase reliability, it is necessary to ensure data integrity control, error correction and loss recovery. Several strategies are possible in case of errors or failures:

a) detection of data errors and automatic request for retransmission of damaged blocks;

b) discarding damaged blocks;

c) error correction.

And here correction codes are used, which serve to detect and, if possible, correct errors that occur during the transmission and storage of information. In the correction codes, when writing data to storage locations or during transmission, specially prepared redundant information is added. Redundant data is used to detect and restore missing or damaged information. Establishing the existence of an error does not always mean that it can be corrected

(Karipzhanova, Sagindykov, 2018). Any error-correcting code can also detect errors, but not the other way around. In practice, block codes are mainly used – this is a method of encoding when information is processed in fixed blocks. The primary mathematical framework commonly employed to analyze information storage revolves around the model of channels with erasures. In this model, emphasis is placed on errors such as erasures, which primarily occur due to failures in storage devices, rather than on the distortion of the information itself. Utilizing this model facilitates the development of more effective codes aimed at enhancing storage reliability. This is because identifying the position of an erased channel is significantly simpler than determining the location of a distorted symbol, especially when compared to conventional error-correcting codes.

In the realm of storage technologies, two types of erasure codes are predominantly used:

1. Cyclic Redundancy Check (CRC) represents the simplest form of encoding that does not necessitate intricate calculations. CRC functions by generating a checksum, computed as the parity of bits within the received block. This checksum enables the detection and correction of single errors.

2. Reed-Solomon codes, on the other hand, are more intricate and computationally intensive. These codes, categorized as cyclic codes, are capable of error correction within data blocks. Instead of bits, the elements of the code vector are blocks comprising groups of bits, with the commonly utilized codes operating with bytes. Reed-Solomon codes represent a specific instance of BCH codes.

The use of erasure codes improves data storage reliability with significantly less redundancy than replication, thereby reducing overhead.

In addition, it is worth mentioning the RAID reliability enhancement technology. Initially, developers introduced different levels of RAID specification, which have become de facto standards: RAID1 – mirrored disk array; RAID2 – reserved for arrays using Hamming code; RAID3 and RAID4 – interleaved disk arrays with dedicated parity disks; RAID5 - interleaved disk array without dedicated parity disk. The reliability technology used in RAID uses parity-based error-correcting CRC codes. Although the somewhat more complex Hamming code algorithm was reserved for RAID2, it has never found practical application (Karipzhanova, Sagindykov, 2018).

Let's take a look at RAID technology from the perspective of providing reliable storage for large amounts of data using the most popular RAID5 as an example. Initially, everything works smoothly, and any faulty sectors that occur do not result in data loss because they are immediately compensated for by parity data. If one of the disks fails, RAID loses its error recovery capability, requiring the failed disk to be replaced with a serviceable one. To recover lost data from a failed disk, you must run the recovery procedure.

If the probability of failure of a single disk is *Pdev*, then the probability of failure of one of the disks in an array of *n* disks increases in proportion to the sum of all probabilities:

$$P = P_{dev,1} + P_{dev,2} + \cdots + P_{dev,n} \tag{12}$$

The probability of a disk failure in an array is *n* times higher than the probability of a single disk failure. those. in RAID, the more disks there are, the higher the probability of a single disk failure:

$$P = nP_{dev} \tag{13}$$

Previously, when 1 TB of data was considered a fairly large amount, RAID used low-

capacity disks. In server systems where performance is the main thing, high-speed disks with a rotation speed of 10-15 thousand revolutions per minute and a maximum capacity of 11 GB were used. But today the processing and storage of huge volumes of information is required, so the capacity of disks reaches many terabytes. In such conditions, hardware reliability factors begin to take their toll, which can no longer be neglected.

Let's consider the impact on the reliability of such a parameter as UER. Let there be a RAID5 consisting of 6 disks of 600 GB each with UER=10-15. After one of the disks fails, the process of recovering the lost data begins. The total amount of data read during Rebuild: 5 disks of 600GB is equal to 0,24.1014 bits. Probability of an unrecoverable error: *PUER* = (0,24**.**1014)**.**10-15 = 0,024. You need to read stripes from five disks, calculate checksums and write them to the sixth disk, which was replaced with a new, working one. During the Rebuild process, there is a possibility of errors in the form of a "damaged" bit ("rotten bit" - bit rot), which cannot be corrected. For the configuration under consideration, on an array of 6 600 GB Enterprise class disks, the probability of impossibility of data recovery is 2.4%. This means that if we use RAID5 with large disks, we may have an unacceptably high probability of data loss.

The larger the disk size and the greater the number of disks in the array, the greater the likelihood of data loss. For example, for an array of 16 2TB disks, if one of them fails, the total amount of data read during Rebuild (15 2TB disks) is 2.40.1014 bits. Probability of an unrecoverable error: *PUER* = (2,40**.**1014)**.**10-15=0,24. Those. the probability of a recovery error already reaches 24%.

Modern storage systems use disks with capacities of up to 10 TB, and the process of increasing volumes will continue. The use of standard RAID controllers in storage systems is unacceptable. In this regard, new multi-disk storage technologies are being developed using more complex error correction algorithms. Consequently, an increase in the physical volume of storage facilities, an increase in their cost and energy consumption is inevitable.

This large-scale problem determines the choice of the main direction of our research, namely: studying the possibilities of the RDB security principle with data splitting and the use of multidimensional parity algorithms that are resistant to partial losses of storage locations. In this system, the security of the RDB is determined not by protecting information in its pure form, but by ensuring its reliability. Security in this case means: we do not lose information.

As indicated in the general overview above, in order to save information, it is common today to make numerous copies (replication, backup storage). But the more information is stored, the more disks are required and the less reliable the storage. Plus, disks do not work on their own, but are located in computers and are served by additional electronics, and electronic devices have their own degree of reliability. And when information is stored on multiple devices, it is even less reliable (Feltynowski, 2023).

Results and discussion.

Results and discussion (Results and discussion). The rationale for our chosen direction of research is that the proposed security system is designed specifically for distributed storage, which is carried out on many disks. During normal storage, information has to be copied several times, because storing it on several disks in itself is no longer reliable. The technology being researched allows multiple drives to fail, even simultaneously.

Conclusions.

Accordingly, our research includes: justification of the principle of distributed information storage with data splitting using multidimensional parity algorithms that are resistant to partial losses of storage locations; theoretical justification for the efficiency of distributed storage with multidimensional parity control; development of appropriate software; verification (testing) of a distributed information storage system with data splitting using multidimensional parity algorithms.

Conflict of Interest. The authors declare that there are no conflicts of interest.

Acknowledgments. The authors thank their mentor and teacher, Erbol Asylkhanovich Kurmanbayev, for his support in scientific research.

References

Adéle Da Veiga, Nico Martins, «Information security culture and information protection culture: A validated assessment instrument», Computer Law & Security Review, 31, 2. – P. 243-256, 2015, https://doi.org/10.1016/j.clsr.2015.01.005

Bouras, C., Kokkinos, V., & Tseliou, G. (2013). Methodology for Public Administrators for selecting between open source and proprietary software. Telematics and Informatics, 30(2), 100-110. https://doi.org/10.1016/j.tele.2012.03.001

D.Chen and H.Zhao, «Data Security and Privacy Protection Issues in Cloud Computing», 2012 International Conference on Computer Science and Electronics Engineering, Hangzhou, China, 2012, pp. 647-651, doi: 10.1109/ICCSEE.2012.193.

Feltynowski, M. (2023). Technological challenges associated with land-use policies in Polish cities and towns. Acta Scientiarum Polonorum Administration Locorum, 22(1), 33-43. https://doi.org/10.31648/aspal.8090

Karipzhanova A., Sagindykov K., Dimitrov K., «Justification of the method and algorithm of multidimensional parity control in distributed databases of information systems», Proc. X National Conference with International Participation «Electronica 2019», May 16-17, 2019, Sofia, Bulgaria. – https://ieeexplore.ieee.org/xpl/conhome/8816819/proceeding. – DOI: 10.1109/ELECTRONICA. 2019.8825600.

Khashirova T.Y., Mamuchiev I.I., Mamuchieva M.I., Ozhiganova M.I., Kostyukov A.D. and Shumeiko I., «Assessment of Information Security in Integrated Systems», 2021 International Conference on Quality Management, Transport and Information Security, Information Technologies (IT&QM&IS), Yaroslavl, Russian Federation, 2021. – Pp. 201-205, https://doi:10.1109/ITQMIS53292.2021.9642824

Kurmanbaev E.A., Syrgabekov I. N., Zadauly E. Karipzhanova A.Zh., Urazbaeva K.T. «Information Security System on the Basis of the Distributed Storage with Splitting of Data», International Journal of Applied Engineering Research. – 2017. – Vol. 12. – № 8. – Pp. 1703-1711.

Menard, Philip, Bott, Gregory J., Crossler, Robert E., « User Motivations in Protecting Information Security: Protection Motivation Theory Versus Self-Determination Theory», Journal of Management Information Systems, 2017, https://doi.org/10.1080/07421222.2017.1394083

Stanislav Mamonov, Raquel Benbunan-Fich, «The impact of information security threat awareness on privacy-protective behaviors», Computers in Human Behavior, 83. – Pp. 32-44, 2018, https://doi.org/10.1016/j.chb.2018.01.028

Phesto P. Namayala, Tabu S. Kondo and Leonard J. Mselle, «The Factors Affecting User Experience Maturity in Free and Open-Source Software Community: An Empirical Study», Taylor & Francis, 1 – 17, 2023, SN - 1044-7318 https://doi:10.1080/10447318.2023.2262270

Von Solms, Rossouw, Van Niekerk, Johan, «From information security to cyber security», Computers & Security, https://doi.org/10.1016/j.cose.2013.04.004

WD Black PC HD Series Specification Sheet, Western Digital [Электронный ресурс] / 26.11.2019. – Режим доступа: https://www.wdc.com/content/dam/wdc/website/downloadable_assets/eng/spec_data_sheet/2879-771434.pdf

WD Gold Enterprise-class Hard Drives Specification Sheet // Western Digital [Электронный ресурс] 26.11.2019. – Режим доступа: https://www.wdc.com/content/dam/wdc/website/downloadable_assets/eng/spec_data_sheet/2879-800074.pdf

Атаманов Ю.С. «Проблемы распределённых СУБД», Молодой ученый. – 2017. – № 15 (149). – С. 5 // Atamanov Yu.S., «Problemy raspredelyonnyh SUBD», Molodoj uchenyj. – 2017. – № 15 (149). – S. 5

Боровиков С.М., «Использование распределения Вейбула для прогнозирования параметрической надежности изделий электронной техники», Доклады БГУИР. – 2011. – № 4. – С. 31-37 // Borovikov S.M., «Ispol'zovanie raspredeleniya Vejbula dlya prognozirovaniya parametricheskoj nadezhnosti izdelij elektronnoj tekhniki», Doklady BGUIR. – 2011. – № 4. – S. 31-37.

Григорьев Ю.А., Плужников В.Л., «Анализ времени обработки запросов к хранилищу данных в

параллельной системе баз данных», Информатика и системы управления. – 2011. – № 2. – С. 94-106 // Grigor'ev Yu.A., Pluzhnikov V.L., «Analiz vremeni obrabotki zaprosov k hranilishchu dannyh v parallel'noj sisteme baz dannyh», Informatika i sistemy upravleniya. – 2011. – № 2. – S. 94-106.

Дейт К.Дж., «Введение в системы баз данных», 8-е изд. – СПб.: Диалектика, – 2019. – С.1328 // Dejt K.Dzh., «Vvedenie v sistemy baz dannyh», 8-e izd. – SPb.: Dialektika, – 2019. – S. 1328.

Ёсу М.Т., Вальдуриес П. «Принципы организации распределенных баз данных», пер. с англ. Слинкина А.А. – М.: ДМК Пресс, – 2021. – С. 672 // Yocy M.T., Val'duries P. «Principy organizacii raspredelennyh baz dannyh», per. s angl. Slinkina A.A. – M.: DMK Press, – 2021. – S. 672.

Карипжанова А.Ж., Сагиндыков К.М., «Система безопасности распределенных баз данных в облаке с использованием технологии многомерной четности», Вестник Государственного университета имени Шакарима. Технические, биологические науки. – 2019. – № 1(85). – С. 194-200 // Karipzhanova A.Zh., Sagindykov K.M., «Sistema bezopasnosti raspredelennyh baz dannyh v oblake s ispol'zovaniem tekhnologii mnogomernoj chetnosti», Vestnik Gosudarstvennogo universiteta imeni Shakarima. Tekhnicheskie, biologicheskie nauki. – 2019. – № 1(85). – S. 194-200.

Карипжанова А.Ж., Сагиндыков К.М., «Способы повышения надежности хранения информации в базах данных», Вестник Казахского Гуманитарно-Юридического Инновационного Университета, 3 (39), 2018 год, – С. 264-269 // Karipzhanova A.Zh., Sagindykov K.M., «Sposoby povysheniya nadezhnosti hraneniya informacii v bazah dannyh», Vestnik Kazahskogo Gumanitarno-Yuridicheskogo Innovacionnogo Universiteta, 3 (39), 2018 god. – Pp. 264-269.

Корнеев В., «Крупные атаки хакеров в 2001-2016 годах: хронология», ТАСС-Досье [Электронный ресурс], 26.11.2019. – Режим доступа: https://tass.ru/info/2619230 // Korneev V., «Krupnye ataki hakerov v 2001-2016 godah: hronologiya», TASS-Dos'e [Elektronnyj resurs], 26.11.2019. – Rezhim dostupa: https://tass.ru/info/2619230

Омельяненко М.В., Папинашвили В.Г., «Решение проблем параллельной обработки транзакций и выход из тупиковых ситуаций в базах данных», Молодой ученый. – 2017. – № 9 (143). – С. 31-34 // Omel'yanenko M.V., Papinashvili V.G., «Reshenie problem parallel'noj obrabotki tranzakcij i vyhod iz tupikovyh situacij v bazah dannyh», Molodoj uchenyj. – 2017. – № 9 (143). – S. 31-34.

Распределение Вейбулла [Электронный ресурс], 26.11.2019. – Режим доступа: https://ru.wikipedia.org/wiki/Распределение_Вейбулла // Raspredelenie Vejbulla [Elektronnyj resurs], 26.11.2019. – Rezhim dostupa: https://ru.wikipedia.org/wiki/Raspredelenie_Vejbulla

Сыргабеков И., Задаулы Е., Курманбаев Е., «Защита информационных баз по методу распределенного хранения», Доклады Национальной академии наук Республики Казахстан. – №5. – 2017. – С. 141-153 // Syrgabekov I., Zadauly E., Kurmanbaev E., «Zashchita informacionnyh baz po metodu raspredelennogo hraneniya», Doklady Nacional'noj akademii nauk Respubliki Kazahstan. – №5. – 2017. – S. 141-153.

Information about authors

Mukasheva Gulzira Yersainovna – Master, Alikhan Bokeikhan University, Semey, Kazakhstan, E-mail: gulzira_7777@mail.ru

Karipzhanova Ardak Jumagazievna – doctor of PhD, Alikhan Bokeikhan University, Semey, E-mail: kamilakz2001@mail.ru

Stanio V. Kolev – doctor of Technical Sciences, Technical University of Sofia, Sofia, Bulgaria, E-mail: skolev@tu-sofia.bg

Akhmetova Zhanar Zhumanovna – doctor of PhD, a.a.Professor, L.N.Gumilyov Eurasian National University, Astana, Kazakhstan, E-mail: zaigura@mail.ru

Uskenbayeva Gulzhan Amangazievna – doctor of PhD, a.a.Professor, L.N.Gumilyov Eurasian National University, Astana, Kazakhstan, E-mail: gulzhum_01@mail.ru