applied sciences

MDPI

Article

Technique for Evaluating the Security of Relational Databases Based on the Enhanced Clements–Hoffman Model

Vitalii Yesin [1], Mikolaj Karpinski [2,*], Maryna Yesina [1,*], Vladyslav Vilihura [1] and Stanislaw A. Rajba [2]

[1] Department of Security of Information Systems and Technologies, Faculty of Computer Science, V. Karazin National University of Kharkiv, 61022 Kharkiv, Ukraine; v.i.yesin@karazin.ua (V.Y.); viligura93@gmail.com (V.V.)

[2] Department of Computer Science and Automatics, Faculty of Mechanical Engineering and Computer Science, University of Bielsko-Biala, 43-309 Bielsko-Biala, Poland; rajbas@ath.bielsko.pl

* Correspondence: mkarpinski@ath.bielsko.pl (M.K.); m.v.yesina@karazin.ua (M.Y.)

Abstract: Obtaining convincing evidence of database security, as the basic corporate resource, is extremely important. However, in order to verify the conclusions about the degree of security, it must be measured. To solve this challenge, the authors of the paper enhanced the Clements–Hoffman model, determined the integral security metric and, on this basis, developed a technique for evaluating the security of relational databases. The essence of improving the Clements–Hoffmann model is to expand it by including a set of object vulnerabilities. Vulnerability is considered as a separate objectively existing category. This makes it possible to evaluate both the likelihood of an unwanted incident and the database security as a whole more adequately. The technique for evaluating the main components of the security barriers and the database security as a whole, proposed by the authors, is based on the theory of fuzzy sets and risk. As an integral metric of database security, the reciprocal of the total residual risk is used, the constituent components of which are presented in the form of certain linguistic variables. In accordance with the developed technique, the authors presented the results of a quantitative evaluation of the effectiveness of the protection of databases built on the basis of the schema with the universal basis of relations and designed in accordance with the traditional technology of relational databases.

Keywords: security; security model; security measure; security evaluation; database

Citation: Yesin, V.; Karpinski, M.; Yesina, M.; Vilihura, V.; Rajba, S.A. Technique for Evaluating the Security of Relational Databases Based on the Enhanced Clements–Hoffman Model. *Appl. Sci.* **2021**, *11*, 11175. https://doi.org/10.3390/app112311175

Academic Editors: Gianluca Lax and Antonia Russo

Received: 10 October 2021
Accepted: 23 November 2021
Published: 25 November 2021

1. Introduction

The growth of Big Data and the vision of a data-driven world opens up many interesting opportunities, while simultaneously revealing many unresolved problems [1,2]. In particular, the new era of Big Data, which involved many researchers in the "data management game" and forced them to abandon the usual ways of designing, developing and implementing data management solutions, has exacerbated the problem of ensuring data security, since interest in the information circulating inside information systems (IS) has increased not only from legitimate users and owners, but also from attackers. For the latter, databases and data warehouses, as the most important information resources, are some of the most vulnerable and attractive elements of the IS. Security is one of the most important characteristics of the quality of the IS as a whole [3], and databases (DBs), as their main component, in particular. In this regard, the presence of an information protection system, as a complex of software, technical, cryptographic, organizational and other methods, means and measures that ensure the integrity, confidentiality, authenticity and availability of information in conditions of exposure to threats of a natural or artificial nature, has become an integral feature of any modern IS and databases. At the same time, in order to be able to verify the conclusions about the security level, it must be measured in some way.

By now there have been many major efforts to measure or evaluate security, including using the Trusted Computer System Evaluation Criteria (TCSEC) [4], Information Technology Security Evaluation Criteria (ITSEC) [5], the Systems Security Engineering Capability Maturity Model (SSE-CMM) [6], Common Criteria [7]. However, as stated by Jansen et al. [8], each attempt had only limited success. To measure the security of databases in [9], it was proposed to use such metrics as the metrics for losses that arise from security incidents, the database security control costs metric, and confidence metrics. However, specific mathematical expressions allowing to determine their quantitative value, as recommended by the performance measurement guide for information security [10], have not been given. It was also proposed to use a metric consisting of several levels to evaluate the security of databases [11,12]. A set of requirements that must be met by the system in order to achieve the corresponding level of security was listed for each level. However, this assessment was qualitative, although ranked. Neto et al. [13] proposed to evaluate the security of database configurations based on a survey of database administrators about the use of certain best practices in the system under study, followed by the definition of a security index. Developing this approach, the Oracle Corporation has developed a tool [14] to assess the security of its databases, which analyzes database configurations, users, their rights, security policies and determines where sensitive data are located in order to identify security risks and improve the state of database security. However, all of these decisions are usually based on intuition and are fragmented. In many cases, there is no integral metric to evaluate the security degree of the database as a whole.

In this connection, the objectives of our paper are:

(a) To present a technique for evaluating the security of relational databases, the security system of which is based on the provisions of the enhanced theoretical Clements–Hoffman model, and the degree of security is calculated on the basis of a determined integral quantitative metric. This metric is the reciprocal of the total residual risk associated with the possibility of implementing threat in relation to a database object when using security measures;

(b) To show the practical application of this technique for measuring the security of relational databases, including in order to identify a more secure one (in which solutions are used that provide a higher degree of database security).

The main contribution of the authors is the creation of a technique for evaluating the security of relational databases, based on the enhanced Clements–Hoffman model, which they obtained, and the integral metric of database security defined by them. The Clements–Hoffman model, traditionally considered the basis for the formal description of security systems, has been expanded to include a variety of object vulnerabilities. At the same time, vulnerability is considered as a separately objectively existing category. This makes it possible to evaluate the likelihood of an unwanted incident (threat realization) and the database security as a whole more adequately. As an integral metric of database security, the reciprocal of the total residual risk was determined, the constituent components of which characterize the strength of a certain security barrier and are presented in the form of certain linguistic variables. This made it possible to quantify the security of databases. In accordance with the evaluation technique developed by the authors and the formulated assumptions, a comparative analysis of their security was carried out on the example of relational databases created using various technologies. As analyzed databases, we researched databases designed according to the traditional technology of relational databases and built them based on the schema with the universal basis of relations (UBR) [15]. The expediency of researching a database with UBR is due to the fact that within the framework of its invariant schema, many original solutions have been implemented related to the protection of data and stored programs. This ensures that the data stored and processed in them is secure.

The rest of this paper is organized as follows: Section 2 presents related works from the literature; in Section 3, we give a formalized description of a full overlap security system (a covered security system) for databases. Section 4 presents the evaluation technique

of database security. Section 5 presents the results of a comparative assessment of the effectiveness of database security measures proposed within the framework of the database schema with UBR with the existing solutions implemented within the framework of traditional relational databases (RDB). Section 6 concludes this work.

2. Related Works

Information security metrics, as noted in the NIST document [8], are an important factor in making informed decisions on various aspects of security, from the design of architectures and security controls to effectiveness and efficiency security operations. Effectiveness is understood as a property of the assessment object, representing how well it provides security in the context of its actual or proposed operational use [5,6]. Security effectiveness means the confidence that the security-enforcing mechanisms of the system meet the stated security objectives (that is, they do nothing other than what they should do while satisfying expectations for resiliency) [8,16,17]. Security efficiency denotes assurance that adequate security quality has been achieved in the system under study, meeting the resource, time and cost constraints [16,17].

A systematic survey of system security metrics is given in [18]. To measure security at the system level, the authors propose a structure of security sub-metrics based on vulnerability metrics, defenses metrics, attack metrics, and situation metrics. Each of these sub-metrics has a hierarchical structure. This paper discusses open questions in the research domain of security metrics and proposes key factors for improving security metrics from a system security perspective.

Despite the abundance of models and recommendations used for evaluating information security performance, Bernik et al. [19], referring also to other authors [20–22], point to the lack of studies that could comprehensively measure or consider information security through the use of specific positioning indicators. They criticize the existing models for their narrow focus or impossibility to apply in practice. Therefore, they propose their own multilevel model for measuring information security performance, which belongs to the scope of qualitative assessment of organizations' systems.

Based on the argumentation theory, Yasasin et al. [23] derived and showed what requirements should be fulfilled by the security metrics of information technology (IT). Katt et al. [24] proposes a quantification method that aims to evaluate the security assurance of systems by measuring the level of confidence that mechanisms that meet security requirements are present and the vulnerabilities associated with potential security threats are absent. They use this method to evaluate the security level of some REST APIs. Sanders [25], noting much work done in the development of methods for quantitative security assessment, speaks of the need for multiple approaches, including formal methods, probabilistic methods, benchmarking and experimentation, classical risk assessment, threat and vulnerability assessment, as well as informal and semi-formal methods. At the same time, for the developed metrics and approaches to be useful, their usability must be thoughtful. Various aspects of database security are discussed in [11,12,26–31].

Obtaining sufficient and credible security evidence of the system under study is one of the main challenges in information security engineering and management is noted in [16]. System developers, project managers, and executive management need information about the security status of technical systems at various stages of the system lifecycle. This study proposes a new Security Metrics Objective Segments (SMOS) model to enable the design of security metrics taxonomies. The model can be integrated with risk-based security metrics development approaches.

The studies carried out and described in [17] revealed such factors contributing to a holistic perception of security effectiveness in software systems, as evidence of (a) direct security effectiveness, (b) quality of risk assessment, (c) security correctness and system quality. However, as noted in the paper, their practical application causes certain difficulties. For example, measuring security effectiveness directly is not easy, and in practice, it is only

partially possible. In this connection, further research is needed for definition of a rigorous methodology enabling systematic development of security effectiveness metrics.

Mishra et al. [32] analyze the impact of security policy, deterrence practices and system audit on the information security effectiveness. Fabian et al. [33] consider the conceptual framework for security engineering with an emphasis on elicitation and analysis of security requirements. This conceptual framework, as a guide for comparing different methods of developing security requirements, is used by the authors to compare and evaluate current approaches to developing security requirements, such as Common Criteria, Secure Tropos (Tropos is a software development methodology based on the paradigm of agent-oriented software development), Security Requirements Engineering Process (SREP), Multilateral Security Requirements Analysis (MSRA), as well as methods based on Unified Modeling Language (UML) and problem frames. Mapping the terminology of a particular method with a conceptual framework allows to assess the method scope and, therefore, its usefulness for a given purpose. This paper provides an example for comparing methods that can help practitioners and academics to choose the method that best suits their application area.

The fundamental monograph [34] and paper [35] discuss the concept of a covered security system, where at least one security measure exists for each identified penetration path. They also describe a formal model (known as the Clements–Hoffman model) that defines the protection domain, the threat domain, security measures and the relationship between them. The model systematizes the resistance, probability, and value measurement process. Resistance is taken to mean the degree to which a security technique succeeds in combating the set of threats against which it has been implemented. The measurement process is based on fuzzy set theory.

Various approaches to measuring security, which can be conditionally classified as cost, functional and based on risk analysis, with appropriate methods and metrics for evaluating the asset protection, are described in [36–41].

The basis for holding any works in the information security area, including the assessment of the protection effectiveness, are International Standards, including ISO/IEC 15408 [42], ISO/IEC 27001 [43], ISO/IEC 27004 [44]. Thus, the International Standard ISO/IEC 15408 defines a common set of requirements for the security functionality of information technology products that can be implemented in the form of hardware, firmware or software, and for the assurance measures applied to these IT products during a security evaluation. It also defines a common approach (model) to assessing security, taking into account threats, vulnerabilities, assets, and risks of harm and the choice of countermeasures. ISO/IEC 15408 is applicable to risks arising from human activities (malicious or otherwise) and to risks arising from non-human activities. It is flexible enough, enabling a range of evaluation methods to be applied to a range of security properties of a range of IT products. Therefore, users of the standard are advised to be careful that this flexibility is not misused. For example, using standards in conjunction with unsuitable evaluation methods, inappropriate security properties, or inappropriate IT products can lead to meaningless evaluation results.

The International Standard ISO/IEC 27004 provides guidelines to assist organizations to evaluate the information security (InfoSec) performance and the effectiveness of an information security management system (ISMS) in order to fulfill the requirements set out in ISO/IEC 27001. It establishes monitoring and measurement of information security performance, monitoring and measurement of the ISMS effectiveness, including its processes and controls, analysis and evaluation of monitoring and measurement results.

Thus, from the experience gained to date, it can be concluded that security measurement is a tough problem that should not be underestimated. Therefore, for its solution today various approaches are proposed, including those mentioned above. In addition, since, in the general case, the formulation of the problem of ensuring information security can vary widely, and the effectiveness of the functioning of the information protection system depends on many factors and is evaluated by a set of metrics that are in complex

interrelationships, then the variety of the methods of evaluating the protection effectiveness is natural. These approaches and methods are mostly based on intuition, are empirical and fragmented, and the authors of this paper wanted to find some scientific-methodological, general approach to solving this problem. Therefore, having analyzed and summarized various, including the above-mentioned approaches and achievements in the domain of evaluating the security of information systems, the authors concluded that it is advisable to use the Clements–Hoffman model. This model is based on the theory of graphs, fuzzy sets, and probabilities. It is traditionally considered the basis for the formal description of protection systems.

Below, based on this model, after its certain enhancement, a technique for evaluating relational databases is proposed.

3. Enhanced Clements–Hoffman Model for Databases

So, let us take as a basis the Clements–Hoffman model in the form of a 5-tuple:

$$S = \{O, T, W, V, B\} \quad (1)$$

where O is the set of protected objects; T is the set of security threats; V is the set of vulnerabilities representing paths of implementing threats T in relation to objects O, determined by a subset of the Cartesian product $V = T \times O$; B is the set of barriers representing the points at which protection is required, defined by a subset of the Cartesian product $B = V \times W = T \times O \times W$.

At first, let us clarify some of these elements in relation to databases:

– $T = \{t_i\}$, $i = 1..I$ is the set of database security threats. According to studies [11,26,28,31,45–47], the main largest and most important threats (types of threats) to database security (to a greater extent they are associated with anthropogenic sources of threats—people or groups of persons, as a result of whose actions or inaction, the security of the considered system has been violated) are:
  - ✓ Excessive and unused privileges. For definiteness, let us designate this type of threat as t_1;
  - ✓ Privilege abuse—t_2;
  - ✓ Input injection—t_3;
  - ✓ Malware—t_4;
  - ✓ Wweak audit trail—t_5;
  - ✓ Storage media exposure—t_6;
  - ✓ Exploitation of vulnerabilities and misconfigured databases—t_7;
  - ✓ Unmanaged sensitive data—t_8;
  - ✓ Inference—t_9;
  - ✓ Denial of service—t_{10};
  - ✓ Limited security expertise and education—t_{11}.
– $O = \{o_j\}$, $j = 1..J$ is the set of protected database objects. Considering that database systems are information products with a dual nature (that is, consisting of two components (assets): DBMS software, independent of their scope, structure, semantic content of the accumulated and processed data, and the actual stored data), as well as the possible harmful effects on the corresponding assets, it is advisable to ensure the security of both components. For relational databases, as the most widespread (this thesis is confirmed by the results of DB-Engines and Popularity of Programming Language (PYPL) ratings [48,49], as well as reports of experts from the world-famous company Gartner, Inc. [50,51]), taking into account the possibility of various degrees of detail of these components, the following objects of protection can be distinguished [11,52]:
  - ✓ The entire database—o_1;
  - ✓ Tables—o_2;
  - ✓ Views—o_3;
  - ✓ Tuples (rows) of tables—o_4;

✓ Separate fields (attribute values) of rows—o_5;
✓ Triggers—o_6;
✓ Persistent stored modules—o_7 and some others.

– $W = \{w_k\}$, $k = 1..K$ is the set of security measures (also referred to in the literature [53–57] as controls), which include any process, policy, device, established practice, or other action which modifies risk [57]).

The elements of all the sets listed above are among themselves in certain relationships, at that the relationship between threats and objects is not a "one-to-one" relationship. Threat $t_i \in T$ can spread to any number of objects O, and object $o_j \in O$ can be vulnerable to more than one threat T.

Now we note one feature of the presented Clements–Hoffman model (Equation (1)). Hoffman and Clements [35], introducing the concept of vulnerability, formally represent it as a mapping of $T \times O$ onto a set of ordered pairs $v_r = (t_i, o_j)$, and not a separately objectively existing category—*vulnerability* (weakness asset or control that can be exploited by one or more threats [57]). Threats exist separately from asset weaknesses. Vulnerability in itself does not cause damage it is only a condition or set of conditions that allows a threat to harm assets. When a threat is realized, one or more vulnerabilities of an asset can be used [58]. At that, one type of vulnerability can lead to many various security threats. Therefore, it is advisable to consider threats and vulnerabilities as a whole. Therefore, it is advisable to consider threats and vulnerabilities in the complex. Only together, they can cause an unwanted incident that can harm the system (assets). Furthermore, in this case, it is necessary to correctly define threats, vulnerabilities and the relationship between them.

In this regard, we will extend the above model with full overlap to a 6-tuple by including a set of vulnerabilities (weakness) of objects (Γ):

$$S' = \{O, T, \Gamma, W, V, B\} \quad (2)$$

where the main components of tuple (2) basically correspond to the components of tuple (1). The distinctive features are shown below.

After the corresponding clarification of the model, the set V will be the set of ordered triples $v_r = (t_i, \gamma_\psi, o_j)$, $\psi = 1..\Psi$, where $\gamma_\psi \in \Gamma$ is the vulnerability (its type) used by the threat $t_i \in T$ aimed at violating the security of the object $o_j \in O$. The set of barriers will be accordingly defined as: $B = V \times W = T \times \Gamma \times O \times W = \{b_l = (t_i, \gamma_\psi, o_j, w_k),\ l = 1..L\}$. Furthermore, the condition for ensuring full security will take the following form: $\forall(v_r)$, $\exists(b_l = (t_i, \gamma_\psi, o_j, w_k)) \in B$. This condition means that for each triple (t_i, γ_ψ, o_j) from the set V, a barrier $b_l \in B$ is created, which makes it impossible to implement an undesirable incident (implementation of the $t_i \in T$ threat using vulnerability $\gamma_\psi \in \Gamma$) in relation to the protected object $o_j \in O$.

In order to have a clear idea of what types of vulnerabilities are most important for databases, the authors of the paper, based on the analysis of existing taxonomies of vulnerabilities, determined a list of the main common weaknesses. It was based on the specification from the Common Weakness Enumeration (CWE) more precisely the classification of the abstract representation of the Research Concepts CWE [59] used by academic researchers, vulnerability analysts, and assessment tool vendors. Taking into account, the specifics of the aspects under consideration, due to the characteristic features of security inherent for databases and DBMS, their list included the following are the main weaknesses of a sufficiently high level of abstraction:

(1) *Improper privilege management*: incorrect assignment of privileges, elevation (escalation) of privileges, performing operations with excessive privileges;
(2) *Improper authorization*: incorrect assignment of permissions for a critical resource, missing authorization, incorrect authorization, exposure of sensitive information through metadata, exposure of sensitive information through data queries. The authorization check is not performed or incorrectly performed when an actor attempts to access a resource or perform an action;

(3) *Improper authentication*: weak password, outdated password, authentication bypass, incorrect implementation of the authentication algorithm, insufficient session expiration, use of a password hash instead of a password for authentication, etc.;
(4) *Uncontrolled resource consumption*: the allocation of a limited resource is not properly controlled, thereby enabling an actor to influence the amount of resources consumed, which ultimately leads to their depletion;
(5) *Cleartext storage of sensitive information*;
(6) *Inadequate encryption strength*;
(7) *Improper scrubbing of sensitive data from decommissioned device*: scrubbing may be missing, insufficient, or incorrect;
(8) *Use of a broken or risky cryptographic algorithm*: use of a non-standard cryptographic primitive with no proven strength;
(9) *Use of insufficiently random values*;
(10) *Insufficient verification of data authenticity*: download of code without integrity check, improper validation of integrity check value, improper verification (no verification) of the cryptographic signature;
(11) *Improper input validation*: improper validation of syntactic correctness of input data, improper validation of specified type of input data, improper validation of consistency within input, improper validation of unsafe equivalence in input. The input data are either not validated, or are incorrectly validated—without assurance that their use will not lead in the future to incorrect and unsafe data processing;
(12) *Use of prohibited code*: functions, libraries, or third party components are used that has been explicitly prohibited, whether by the developer or the customer;
(13) *Embedded malicious code*: Trojan horse, trapdoor, time bomb, logic bomb, spyware, etc.;
(14) *Violation of secure design principles*: unnecessary complexity in the protection mechanism (a more complex mechanism is used than necessary); reliance on a single factor in a security decision; insufficient compartmentalization—functionality or processes that require different privilege levels, rights or permissions are not sufficiently separated; access check is not provided on a protected resource every time the resource is accessed by an entity; insufficient psychological acceptability (the difficulty and inconvenience of using the protection mechanism often encourages non-malicious users to disable or bypass it accidentally or deliberately); reliance on security through the obscurity (a defense mechanism is used, the strength of which heavily depends on its obscurity); imperfection of the mechanism for maintaining data integrity;
(15) *Incorrect provision of specified functionality*: the code does not function according to its published specifications, potentially leading to incorrect usage;
(16) *Hidden functionality*: there is functionality that is not documented, not part of the specification, and not accessible through an interface or command sequence. Hidden functionality can take many forms, including, for example, such as intentionally malicious code;
(17) *Incomplete documentation*: there are no descriptions of all relevant elements of the product, such as its usage, structure, interfaces, design, implementation, configuration, operation, etc., which naturally complicates maintenance, indirectly affecting security due to lack of awareness, making it difficult to find and/or fixing vulnerabilities or taking a lot of time, which can also simplify the introduction of vulnerabilities;
(18) *Configuration error*: non-compliance with safety requirements during the installation and configuration of the database. Administrative, auxiliary, educational accounts are installed, which are registered in the database by default without proper analysis and changing of default passwords, no limitations on the length and complexity of passwords are set, unused accounts are not blocked, critical updates are not installed, the event audit system is improperly configured, etc.

For definiteness, we denote them, respectively, as $\gamma_1, \ldots, \gamma_{18}$.

For a better representation (understanding) of the relationship between the main elements of the security system under consideration, Figure 1 in the form of a class diagram

in the Unified Modeling Language (UML) notation shows these high-level security concepts and their relationship. The relationships between the security system elements under consideration are many-to-many relationships, subdivided into so-called associations (represented by straight lines), and dependencies (represented by dashed lines).

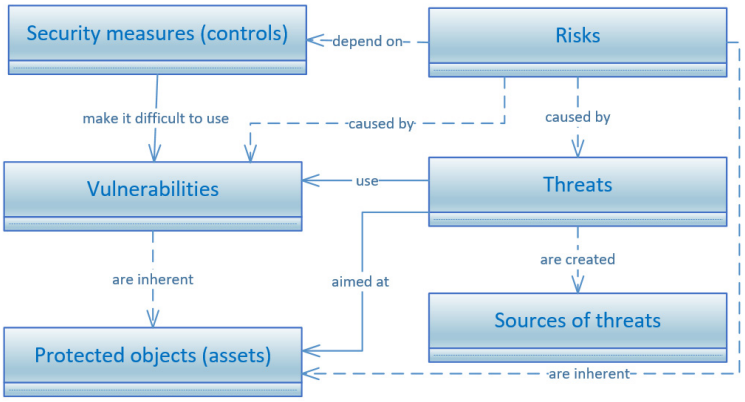

Figure 1. Security concepts and their relationship.

Ideally, each protection mechanism (security controls, security measures) should exclude an appropriate path of implementing the threat. In practice, however, these mechanisms provide only a limited amount of resistance to security threats. For example, passwords have a finite length; ciphers have different cryptographic strengths; different frequency of synchronization points between the database and the transaction log leads to all kinds of, sometimes unacceptable, recovery times in case of failures; dependence of security on the relevance and timeliness of installed updates, configuration parameters, etc.

The authors of the model [34,35] believe that for some quantitative evaluation of the security level of objects, it is necessary and possible to measure the degree of system security. As an appropriate structure for expressing such measures, they propose a linguistic variable that assumes values, which are words rather than numbers. To do this, they redefine security barriers B, each of which ($b_l \in B$) is represented as a composite linguistic variable, the components of which are linguistic variables: P_l is the probability of threat occurrence; L_l is the amount of damage (loss) in case of successful implementation of the threat in relation to the protected object; R_l is security measure resistance (the degree of security measure resistance w_k, characterized by the probability of overcoming it). At that, it is noted that these components are evaluated in the context of the specific barrier ($b_l = (t_i, o_j, w_k)$) that they form. The indices of the P_l, L_l, R_l linguistic variables are the same as the barrier index, and not the same as those of the $b_l = (t_i, o_j, w_k)$ barrier components in the basic security system—threats, objects, and security measures (controls). Clements and Hoffman [34,35] state that the resistance value determines the degree of increase or decrease in the overall system security, and an informal combination of the probability and the loss value gives the importance (weight) of the barrier in the overall rating (evaluation). In general, these values determine the contribution of the barrier to the overall system security. However, they do not say anything about specific methods of obtaining (evaluating) them.

Therefore, after analyzing the various approaches set out in relevant sources [60–62] the residual risk Rr has been selected as such an indicator (metric). The risk remaining after risk treatment (residual risk [57]) is associated with the possibility of implementing threat $t_i \in T$ in relation to the DB object $o_j \in O$ when using security measures (controls) $w_k \in W$. Naturally, that a quantitative approach to risk evaluation is preferable to a qualitative one, since it offers a more tangible value of the situation [63]. The residual risk value characterizing the strength of the barrier $b_l \in B$ can be determined as follows [60,61]:

$$Rr_l = P_l L_l (1 - R_l) \quad (3)$$

At the same time, let us clarify that the probability P_l is understood as the probability of an undesirable incident (threat realization), as the product of the probability of the threat occurrence P_{t_i} (the so-called motivational component of the threat realization probability [64]) and the probability of successful exploitation of the vulnerability P_{γ_ψ}: $P_l = P_{t_i} \cdot P_{\gamma_\psi}$ [62]. Furthermore, the amount of damage (loss) L_l in relation to the protected object should be considered from the standpoint of the successful implementation of the threat t_i exploiting the vulnerability γ_ψ.

Residual risk is essentially a measure of insecurity asset. Then, the value of the database security can be determined by calculating the reciprocal of the total residual risk [60,61]:

$$S = \sum_{\forall b_l \in B} \frac{1}{P_l L_l (1 - R_l)} \tag{4}$$

where $P_l, L_l \in (0,1)$, $R_l \in [0,1)$.

If there are no barriers b_l in the system that block certain paths of implementing threats in relation to the objects, the degree of security measure resistance R_l is taken to be zero. From the formal point of view, this can be represented by introducing the so-called null security measure (protection means with a zero degree of providing security) w_o added to the set W. Each unprotected object is assigned such a protection means. Thus, for $\forall (t_i, \gamma_\psi, o_j) \in V$, for which $(\forall k \in K)\ (t_i, \gamma_\psi, o_j, w_k) \notin B$, barrier $(t_i, \gamma_\psi, o_j, w_o)$ is added to the B.

- Thus, the Clements–Hoffman model was extended to a 6-tuple by including a set of vulnerabilities of objects, as a separate objectively existing category. This allows you to evaluate the probability of an unwanted incident and the security of the database as a whole more adequately. In addition, as a result of enhancing the Clements–Hoffman model, taking into account the dual nature of the relational database system and varying degrees of detail of its components, the following were determined: the main objects of protection;
- The list of the main common weaknesses (as some types of vulnerabilities);
- The main significant threats to the security of databases;
- Integral metric of database security (as the reciprocal of the total residual risk).

4. Evaluation Technique of Database Security

It is easy to see that with known values of the probability of an undesirable incident (threat realization) $P_l = P_{t_i} \cdot P_{\gamma_\psi}$, the amount of damage (loss) L_l (with the successful implementation of the threat in relation to the protected object), the degree of corresponding security measure resistance R_l, it is possible to evaluate the database security using Equation (4). However, obtaining accurate P_{t_i}, P_{γ_ψ}, L_l, and R_l values is not an easy task. This is often not possible in practice [58]. In addition, to paraphrase Zadeh [65], as system complexity increases, analytical precision decreases [35]. Therefore, as a rule, in such cases it is advisable to resort to numerical estimates in a certain range of values, especially since each quantitative range can be associated with a certain qualitative scale, with which under certain conditions it is much easier to work. A linguistic variable can serve as a suitable structure for expressing such values, as noted above. For these reasons, first of all, in accordance with the introduced changes in the model, we will redefine the security B barriers, each of which ($b_l \in B$) will be represented as a composite linguistic variable, the components of which are linguistic variables:

- The probability of threat occurrence (P_t);
- The probability of exploiting the vulnerability (P_γ);
- The amount of damage (L) in case of successful implementation of the threat in relation to the protected object;
- The degree of security measure resistance (R), characterized by the probability of overcoming it.

At that, again, we note that these components are assessed in the context of the specific barrier that they form (the $P_l = f(P_{t_i}, P_{\gamma_\psi})$, L_l, R_l indices are the same as the barrier index, and not the same as those of the $b_l = (t_i, \gamma_\psi, o_j, w_k)$ barrier components in the basic security system—threats, vulnerabilities, objects, and security measures).

We begin formalizing the corresponding components with the probability of a threat occurrence P_t. At the same time, we note that in practice, to calculate the risk, it is often not the mathematical probability that is used, but the approximate frequency of its implementation over a certain period. To avoid confusion, the standards deliberately use the concept of *likelihood* instead of the mathematical term *probability*. In what follows, we will use exactly this term.

In view of the above, the likelihood of a threat occurrence P_t can be represented as a linguistic variable:

$$\langle name, T, X, G, M \rangle \tag{5}$$

where *name* is the name of the linguistic variable (in our case, this is the likelihood of a threat occurrence P_t); T is a set of values of a linguistic variable (term-set), which are the names of fuzzy variables (α_ε, where $\varepsilon = 1, 2, \ldots$ ($\varepsilon \in \mathbb{N}^*_{<n}$), n is the maximum number of fuzzy variables), the definition domain of each of which is the set X—a universal set or universe (in this case, these are the numerical values of the probability of threat occurrence P_t); G is some syntactic procedure that allows you to operate with the elements of the term-set T, in particular, generate new terms (values); M is a semantic procedure that makes it possible to transform each new value of a linguistic variable, obtained using the procedure G, into a fuzzy variable, that is, to form a corresponding fuzzy set. In the considered case, we can restrict ourselves to the assumption of the trivial nature of G and M, that is, no logical connectives and modifiers will be used.

An analysis of various relevant sources on the problems of information risk management [53,58,66,67] showed that to evaluate P_t it is enough to enter three verbal gradations with the corresponding approximate quantitative estimates, without which any qualitative scale is meaningless:

- Low likelihood (L). This threat is unlikely to occur. There are no incidents, statistics, motives that would indicate that this can happen. The expected frequency of the threat does not exceed 1 time in 5 years;
- Moderate likelihood (M). There are prerequisites for the emergence of a threat (there have been incidents in the past), there are statistics or other information indicating the possibility of a given threat, the attacker has the motivation to realize appropriate actions. The expected frequency of occurrence of this threat is approximately once a year;
- High likelihood (H). There are objective prerequisites for the emergence of a threat. There are incidents, statistics, or other information indicating that the threat is most likely to realize, the attacker has motives to take appropriate action. The expected frequency of occurrence of a threat is on average once every four months or more often.

This three-level scale, as noted by some experts [53,58,66,67], is usually sufficient for an initial high-level assessment. This is explained by the fact that estimates of the expected frequency of occurrence of a threat from level to level on a qualitative scale differ significantly, so it is unlikely that competent experts would be greatly mistaken in their estimates. Nevertheless, in the future, the authors plan to expand the number of levels by adding several intermediate ones.

On the other hand, the value of the frequency estimate can be converted into the numerical equivalent of the probability of the threat occurrence, corresponding to a certain range of values. The results of the analysis of relevant sources [66,68,69] suggest that, in numerical terms, the likelihood of such a threat at the appropriate level may be in the corresponding range:

- For level L—$P_t = [0, 0.2]$;
- For level M—$P_t = [0.2, 0.6]$;

– For level H—$P_t = [0.6, 1]$.

Then, using the well-known qualitative scales used in assessing information security risks [53,58,66,67], in particular, a three-level qualitative scale, we define the names of fuzzy variables—a set of values of a term-set T: T = {"low likelihood", "moderate likelihood", "high likelihood"} = {"L", "M", "H"}, that is α_1 = "L", α_2 = "M", α_3 = "H".

As you know, when we are talking about a fuzzy variable α, we always mean some fuzzy set $A = \{\mu_A(x)/x\}$, which determines its possible values, where $\mu_A(x)$ is the membership function ($\mu_A(x) \in [0,1]$; $\mu_A(x) : X \to [0,1]$), which indicates the grade of membership of an element x in the fuzzy set A.

The most widespread in the construction of membership functions of fuzzy sets are direct and indirect methods [70,71]. In view of the fact that $x \in X$ can be measured on a quantitative scale, we will use the direct method, when an expert or a group of experts sets for each $x \in X$ the value of the membership function $\mu_A(x)$. The theory of fuzzy sets when using direct methods for constructing the membership function does not require its absolutely precise assignment [70]. Very often, it is enough to fix only the most characteristic values and the view of the function $\mu_A(x)$.

Based on the analysis of the main membership functions used to represent such properties of fuzzy sets, which are characterized by the uncertainty of types, such as: "small value", "negligible value"; "located in the range", "approximately equal"; "large value", "significant value", for the considered fuzzy variables "L", "M", "H" trapezoidal, linear Z- and linear S-shaped functions were selected. Each of these functions can be represented as follows:

– Linear Z-shaped membership function of a fuzzy set $A_{\mathrm{L}} = \{\mu_{\mathrm{L}}(x)/x\}$, corresponding to a fuzzy variable "L" for a linguistic variable P_t:

$$\mu_{\mathrm{L}}(x; a, b) = \begin{cases} 1, & x \le a, \\ \frac{b-x}{b-a}, & a < x < b, \\ 0, & b \le x, \end{cases} \tag{6}$$

where a, b are numeric parameters ($a \le b$);

– Trapezoidal membership function of a fuzzy set $A_{\mathrm{M}} = \{\mu_{\mathrm{M}}(x)/x\}$ corresponding to a fuzzy variable "M" for a linguistic variable P_t:

$$\mu_{\mathrm{M}}(x; a, b, c, d) = \begin{cases} 0, & x \le a, \\ \frac{x-a}{b-a}, & a \le x \le b, \\ 1, & b \le x \le c, \\ \frac{d-x}{d-c}, & c \le x \le d, \\ 0, & d \le x, \end{cases} \tag{7}$$

where a, b, c, d are numeric parameters ($a \le b \le c \le d$);

– Linear S-shaped membership function of a fuzzy set $A_{\mathrm{H}} = \{\mu_{\mathrm{H}}(x)/x\}$ corresponding to a fuzzy variable "H" for a linguistic variable P_t:

$$\mu_{\mathrm{H}}(x; c, d) = \begin{cases} 0, & x \le c, \\ \frac{x-c}{d-c}, & c < x < d, \\ 1, & d \le x, \end{cases} \tag{8}$$

where c, d are numeric parameters ($c \le d$).

Figure 2 shows all three graphs of the membership functions of fuzzy variables used to determine the linguistic variable—the likelihood of a threat occurrence P_t.

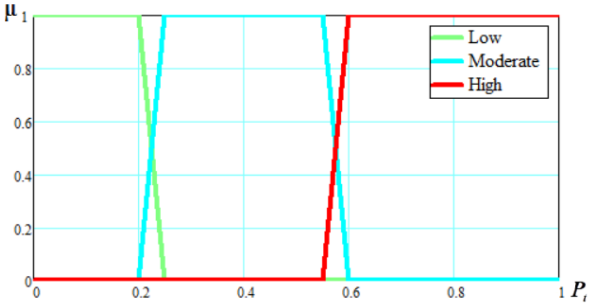

Figure 2. Graphs of the membership function of fuzzy sets A_{L}, A_{M}, A_{H}.

The expert based on a priori knowledge assigns linguistic values, which are the names of fuzzy variables, for each the likelihood of a threat occurrence P_{t_i}, as a component of the corresponding specific barrier b_l. In this case, these values can be represented verbally as "low likelihood", "moderate likelihood", "high likelihood" (or "L", "M", "H"). At that, since each such value is associated with the corresponding membership function with the corresponding approximate quantitative estimates, then, in principle, for each threat $t_i \in T$, it is possible to determine with a limited degree of accuracy the numerical value of this likelihood P_{t_i}, for example, as the *modal value* of a fuzzy set. If the core of a fuzzy set A (is the crisp subset of the domain X consisting of all elements of A with a membership grade equal to one [72]: $C(A) = core(A) = \{x : \mu_A(x) = 1, x \in X\}$) contains more than one element, then for such a set the modal value is calculated as the mean value of the core.

Further, using the above approach, we represent in the form of the corresponding linguistic variable—the likelihood of exploiting the vulnerability—P_γ (the likelihood that in the event of implementing threat in relation to an asset, this threat will be successfully implemented using this vulnerability). To estimate P_γ, we introduce three verbal gradations with the corresponding approximate quantitative estimates:

- High (H). The vulnerability is easy to exploit and there is weak protection or no protection at all. The likelihood of exploiting a vulnerability (the likelihood of successful implementation of a threat due to a given vulnerability) is in the range [0.7, 1];
- Moderate (M). The vulnerability can be exploited, but there is some protection. The likelihood of exploiting a vulnerability is in the range [0.3, 0.7];
- Low (L). The vulnerability is difficult to exploit and there is good protection. The likelihood of exploiting a vulnerability is in the range [0, 0.3].

As with threats, this three-tier scale may be sufficient for an initial high-level evaluation of the vulnerability. In the future, for a more detailed evaluation, the authors also plan to expand it.

Then, using the introduced designations, we define the names of fuzzy variables (β_ε, where $\varepsilon \in \mathbb{N}^*_{<n}$) is the set of values of the term-set T_γ for the linguistic variable P_γ: T_γ = {"high vulnerability", "moderate vulnerability", "low vulnerability"} = {"H", "M", "L"}, that is, β_1 = "H", β_2 = "M", β_3 = "L". The definition domain of each of the fuzzy variables is a set of numerical values ($X \in [0, 1]$) of the likelihood of exploiting the vulnerability. In the case under consideration, we also restrict ourselves to the assumption that G_γ and M_γ are trivial (without logical connectives and modifiers).

Based on the analysis of the main membership functions, similar to the above, for the considered fuzzy variables β_1 = "B", β_2 = "C", β_3 = "H", trapezoidal, linear Z- and linear S-shaped functions were selected.

Figure 3 shows graphs of these membership functions ($\mu^v_{\mathrm{L}}(x)$, $\mu^v_{\mathrm{M}}(x)$, $\mu^v_{\mathrm{H}}(x)$) used to determine the linguistic variable—the likelihood of exploiting the vulnerability P_γ.

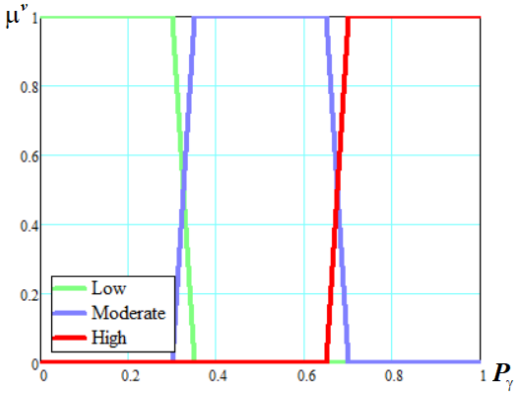

Figure 3. Graphs of the membership function of fuzzy sets $A^v_{\text{L}} = \{\mu^v_{\text{L}}(x)/x\}$, $A^v_{\text{M}} = \{\mu^v_{\text{M}}(x)/x\}$, $A^v_{\text{H}} = \{\mu^v_{\text{H}}(x)/x\}$.

The expert based on a priori knowledge assigns linguistic values, which are the names of fuzzy variables, for each likelihood of exploiting the vulnerability P_γ, as components of the corresponding barrier b_l, thanks to which it becomes possible to implement the corresponding threat t_i. These meanings are presented verbally as "L", "M", "H". Since each such value is associated with the corresponding membership function with the corresponding approximate quantitative estimates, then for each vulnerability γ_ψ, it is possible to calculate with a limited degree of accuracy the numerical value of this likelihood P_{γ_ψ}, for example, as the modal value of the corresponding fuzzy set.

By analogy, you can determine the degree of resistance of the security measures, characterized by the likelihood of overcoming them ($P^{ov}_l = 1 - R_l$). The corresponding levels of control (degrees of resistance) can be determined as follows:

- H is the high degree of security measure (mechanism) resistance (high level of control). It is unlikely that such a mechanism will be overcome. The likelihood of overcoming (bypassing) such a mechanism is in the range $P^{ov}_l \in [0,\ 0.4]$.
- M is the moderate degree of security measure resistance. This measure provides some protection, but it is possible to overcome it, spending some effort. The likelihood of overcoming the corresponding security measure is in the range [0.4, 0.8].
- L is the low degree of security measure resistance. This measure is quite easy to overcome. The likelihood of overcoming the corresponding security measure is in the range [0.8, 1].

Then, using this scale, we define the names of fuzzy variables (δ_ε, where $\varepsilon \in \mathbb{N}^*_{<n}$) is the set of values of the term-set T_R for the linguistic variable R: T_R = {"high degree of resistance", "moderate degree of resistance", "low degree of resistance"} = {"H", "M", "L"}, that is, δ_1 = "H", δ_2 = "M", δ_3 = "L". The definition domain of each of the fuzzy variables is a set of numerical values ($X \in [0, 1]$) of the likelihood of overcoming security measures. In the case under consideration, we also restrict ourselves to the assumption that G_R and M_R are trivial.

Similar to the above approach, for the considered fuzzy variables δ_1 = "B", δ_2 = "C", δ_3 = "H" (with which the corresponding fuzzy sets are associated, defining their possible values: $A^{ov}_{\text{H}} = \{\mu^{ov}_{\text{H}}(x)/x\}$, $A^{ov}_{\text{C}} = \{\mu^{ov}_{\text{C}}(x)/x\}$, $A^{ov}_{\text{B}} = \{\mu^{ov}_{\text{B}}(x)/x\}$) were selected trapezoidal, linear Z- and linear S-figurative membership functions ($\mu^{ov}_{\text{H}}(x)$, $\mu^{ov}_{\text{C}}(x)$, $\mu^{ov}_{\text{B}}(x)$). Figure 4 shows three graphs of the membership functions of fuzzy variables used to determine the linguistic variable R ($R = 1 - P^{ov}$; in some sources [53] P^{ov} is called reverse of the control strength).

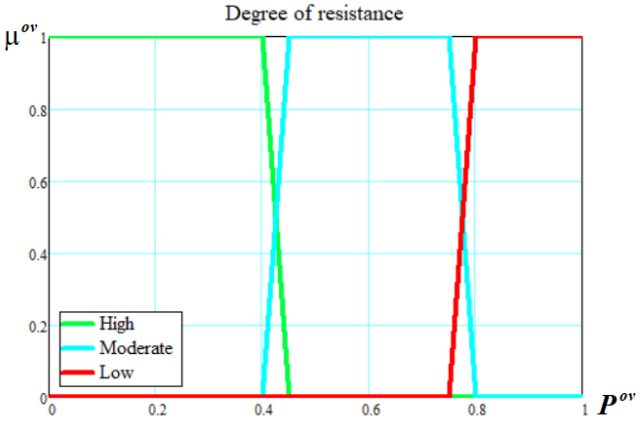

Figure 4. Graphs of the membership function of fuzzy sets A^{ov}_{H}, A^{ov}_{M}, A^{ov}_{L}.

An expert, on the basis of a priori knowledge of security measures used that complicate the exploitation of the corresponding vulnerability γ_ψ, due to which it becomes possible to implement the corresponding threat t_i, assigns the linguistic values ("H", "M", "L") for each R_l as components of the corresponding barrier b_l. In view of the fact that each such value is associated with the corresponding membership function with the corresponding approximate quantitative estimates, then for each security measure $w_k \in W$ of barrier b_l, it is possible to determine the numerical value of both P^{ov}_l and $R_l = 1 - P^{ov}_l$. Again, as the modal value of the corresponding fuzzy set.

The damage caused as a result of security incidents is associated with the target function of the system—one of the relevant indicators, such as lost profit, loss of competitive advantages, deterioration of the organization's reputation, damage to the interests of a third party, financial losses associated with the restoration of resources, etc. For different organizations, the importance of each of them can have significantly different meanings.

From an economic point of view, damage to assets is conveniently expressed in terms of financial losses. However, in practice, obtaining accurate quantitative values of damage is often difficult or even impossible [62]. Nevertheless, most of the losses that cannot be described quantitatively can be represented numerically by using an empirical scale of the damage level—a qualitative scale of measurement, divided into areas (ranks) corresponding to different degrees of satisfaction of the requirements under consideration, for example, on a five-point scale: from 1 to 5. Each of these levels (ranks) can be associated with the value of the term set T_L (T_L = {"Very low", "Low", "Medium", "High", "Very high"} = {"VL", "L", "M", "H", "VH"}) linguistic variable—the amount of damage L. The definition domain of each of the fuzzy variables is the set of numerical values of the damage value/damage level (in points)—$X \in (0, 6)$. In the case under consideration, we also restrict ourselves to the assumption that G_L and M_L are trivial.

For the considered fuzzy variables ρ_1 = "VH", ρ_2 = "H", ρ_3 = "M", ρ_4 = "L", ρ_5 = "VL" (with which the corresponding fuzzy sets are associated, defining their possible values: $A^L_{\mathrm{VH}} = \{\mu^L_{\mathrm{VH}}(x)/x\}$, $A^L_{\mathrm{H}} = \{\mu^L_{\mathrm{H}}(x)/x\}$, $A^L_{\mathrm{M}} = \{\mu^L_{\mathrm{M}}(x)/x\}$, $A^L_{\mathrm{L}} = \{\mu^L_{\mathrm{L}}(x)/x\}$, $A^L_{\mathrm{VL}} = \{\mu^L_{\mathrm{VL}}(x)/x\}$), triangular, linear Z- and linear S-shaped membership functions ($\mu^L_{\mathrm{VH}}(x)$, $\mu^L_{\mathrm{H}}(x)$, $\mu^L_{\mathrm{M}}(x)$, $\mu^L_{\mathrm{L}}(x)$, $\mu^L_{\mathrm{VL}}(x)$) were selected:

$$\mu^L_{\mathrm{VL}}(x; a, b) = \begin{cases} 1, & x \le a, \\ \frac{b-x}{b-a}, & a < x < b, \\ 0, & b \le x; \end{cases} \tag{9}$$

$$\mu^L_{\mathrm{H}}(x; a, b, c, d), \mu^L_{\mathrm{M}}(x; a, b, c, d), \mu^L_{\mathrm{L}}(x; a, b, c, d) = \begin{cases} 0, & x \le a, \\ \frac{x-a}{b-a}, & a \le x \le b, \\ \frac{c-x}{c-b}, & b \le x \le c, \\ 0, & c \le x; \end{cases} \tag{10}$$

$$\mu^{L}_{\mathrm{VH}}(x;c,d) = \begin{cases} 0, & x \le c, \\ \frac{x-c}{d-c}, & c < x < d, \\ 1, & d \le x. \end{cases} \tag{11}$$

Figure 5 shows the graphs of the membership functions of fuzzy variables used to determine the linguistic variable—the amount of damage L.

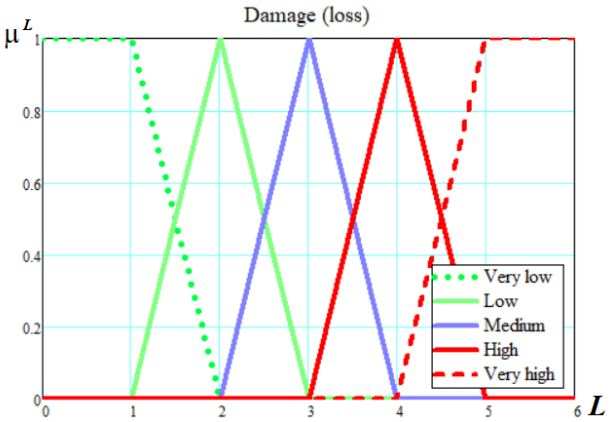

Figure 5. Graphs of the membership function of fuzzy sets A^{L}_{VH}, A^{L}_{H}, A^{L}_{M}, A^{L}_{L}, A^{L}_{VL}.

Table 1 presents an assessment of damage on a five-point scale and its semantic characteristic.

Table 1. The assessment of damage and its semantic characteristic.

Level	T_L	*Semantic Characteristic*
1	Very low	Loss can be ignored.
2	Low	The damage can be easily eliminated; the costs of eliminating the consequences of the threat implementation are low.
3	Medium	Eliminating consequences of the threat implementation is not associated with large costs.
4	High	Eliminating consequences of the threat implementation is associated with significant financial losses.
5	Very high	The organization ceases to exist.

In order for the assessment of the value of assets to make economic sense, it is advisable to correlate the qualitative scale of assessing the damage with the amount of direct financial losses. However, establishing such a correspondence requires additional research in each specific case and depends on many factors for the systems under consideration. Possible independent scales (examples) for assessing direct financial losses and their relative values ($r^{rel}_{fl} = z_{fl}/z^{per}_{fl}$, where z_{fl} is direct financial losses; z^{per}_{fl} is permissible direct financial losses) are shown in Table 2. At that, it should be understood that, depending on the tasks solved by the organization, the area, the nature and scale of its activities, the form of ownership, the value of assets, the severity of the consequences of violating their security and a number of other factors, they may be other.

Table 2. Financial damage assessment scales.

Level	T_L	*Range* z_{fl}	*Range* r^{rel}_{fl}
1	Very low	<100 $	≤0.1
2	Low	(100–1000) $	(0.1, 0.3]
3	Medium	(1000–10,000) $	(0.3, 0.6]
4	High	(10,000–100,000) $	(0.6, 0.9]
5	Very high	>100,000 $	>0.9

Thus, when developing an evaluation technique of database security, the authors, based on a generalization of experts' recommendations, determined the number of levels for the linguistic variables under consideration with their corresponding ranges, as well as the membership functions and a variant of determining the numerical value for the corresponding likelihood or damage.

Having the appropriate data using Equation (4), it is possible to determine the security value of the analyzed database.

It should be noted that the proposed technique, in contrast to some known, is characterized by a certain flexibility. This is manifested in the ability to adapt to new conditions of functioning and to take into account the emerging new actual threats, vulnerabilities, security measures that can be combined into some general groups. Including there is the possibility of choosing the number of levels of the corresponding linguistic variables. At that, the use of the introduced integral security metric makes it possible to evaluate the security value of the investigated RDB quantitatively.

5. Quantifying Database Security

In this section, the authors tried to show, using examples of relational databases developed using various technologies, the ease of use and potential of the proposed technique with explainable and non-contradictory results of evaluating their security that confirm its sufficiency.

Before proceeding to assessing the security of relational databases built using various technologies and comparing their security, we note some important aspects and assumptions.

1. As the studied databases, we consider databases designed based on the schema with the universal basis of relations and according to the traditional technology of relational databases.
2. In the DB with UBR, which can be used as an ordinary DB, a data warehouse of various subject domains (SDs) or a configuration DB of the dataspace management environment [73–75], various security measures are implemented [76–80]. These measures are based on the provisions of the theory of relational databases [8,30,81], formal access control models [82,83] and ensuring data integrity [84], the potential of the modern blockchain model [85,86], row-level security (RLS) technology [87], SQL capabilities [45]. Separate elements of these solutions can be used to protect databases and data warehouses with various models (relational, NoSQL, NewSQL [12,39,82,88–91]). However, in this case, for traditional RDBs, which are investigated below, these measures were not implemented.
3. It is believed that the likelihoods: P_t is the likelihood of occurrence of the corresponding threats $(t_1, \ldots, t_{11})$ and P_γ is the likelihood of exploitation the corresponding vulnerabilities $(\gamma_1, \ldots, \gamma_{18})$ in relation to specific protected objects $(o_j \in O, j = \overline{1,7})$ are the same for the compared databases.
4. Evaluation of the residual risk for the compared databases is carried out for the case of a "Low" amount of damage (damage level-2; Tables 1 and 2) with a relative value of possible financial losses amounting to 0.2 ($L^{\mathrm{UBR_{quant}}} = L^{\mathrm{RDB_{quant}}} = 0.2$, where $L^{\mathrm{UBR_{quant}}}$, $L^{\mathrm{RDB_{quant}}}$ are the numerical values (relative) values of damage L for a database with UBR and traditional database, respectively).
5. As security measures/controls $(w_k \in W)$, some generalized solutions are used associated with a certain process, policy, device, established practice and other actions aimed at modifying the risk, namely:
   - w_1—means that allow to identify and remove incorrectly assigned privileges. Such, for example, as: audit tools, utilities, scripts used by the database administrator (DBA) for aggregating user rights into a single repository, collecting information about users, their roles and behavior, as well as data privacy, identifying users who have too many privileges and users who do not use their rights, viewing and approving/rejecting the individual rights of users, tracking

all actions to access the database, real-time alerts and blocking, detecting unusual access activity, etc.;

- w_2—tools provided by the DBMS and special developed means in the DB schema with UBR (means that ensure the maintenance of a special *log-table of the modified data*, the formation of data for a *special table of users* and some others [76]), allowing to identify and eliminate incorrectly assigned privileges;
- w_3—tools provided by the DBMS and special developed means in the DB schema with UBR (means providing the formation of data from a special *table of the access privilege distribution to the data of other users* and some others [76]), allowing to identify and eliminate incorrectly assigned privileges;
- w_4—tools provided by the DBMS and special developed means in the DB schema with UBR (means providing the data formation from a special *table of restrictions on access rights to a specific data element* and some others [76]), allowing to identify and eliminate incorrectly assigned privileges;
- w_5—means that allow to identify and eliminate excessive privileges; detect vulnerabilities, missing patches from vendors; inactive accounts, modify default passwords; properly configure the event auditing system, including tracking unusual user access activity, etc. Timely installation of patches or the use of virtual patches to protect the database;
- w_6—means that allow detecting unusual user access activity and complicating the leakage of confidential data from database tables (including the use of means for masking data provided by the DBMS and proposed in [79]; the usage of means of restricting access rights to a specific data element [76] implemented in the DB with UBR);
- w_7—means to detect unusual user access activity and complicate code disclosure of confidential persistent modules (including the use of means for masking data provided by the DBMS and proposed in [77]);
- w_8—means that allow to identify and eliminate incorrectly assigned privileges, detect vulnerabilities, inappropriate session duration, improper implementation of the algorithm, authentication protocol, settings. Timely installation of critical updates or the use of virtual patches to protect the database from attempts to exploit vulnerabilities until a full-fledged and permanent patch is deployed;
- w_9—means that allow controlling resource consumption (for example, through the profile mechanism—a named set of resource restrictions that can be used by the user);
- w_{10}—means that allow controlling the integrity of the trigger code and persistent stored modules, including those based on the potential of the modern blockchain model proposed in [78] and implemented in a DB with UBR;
- w_{11}—using parameterized queries, stored procedures, least privileges; escaping user input; converting data types to the type that was assumed by the logic of the program, checking the data entered by the user for compliance with the allowed character sequences;
- w_{12}—maintenance of the list of "prohibited" functions, procedures, the usage of which should be avoided;
- $w_{13}-$ —anti-virus software;
- w_{14}—means providing support for data integrity (both built into the DBMS and specially developed in the DB schema with UBR [76,80]), as well as implementing security models based on discretionary and role-based policies;
- w_{15}—means that implement security models based on: discretionary, mandatory, role-based, attribute policy, including those specific to a database with UBR [76];
- w_{16}—special documented diagnostic functions capable of identifying the causes of defects caused by the incorrect formation of primary keys, entering incorrect data, inadmissible entry, deletion, modification of data, unauthorized access to data, unauthorized changes to the database schema with UBR and its objects

(including using the capabilities of blockchain technology [78]); special triggers that can be used to intercept and log operations performed in the database; DBMS audit tools;

- w_{17}—audit means built into the DBMS, including specially developed means in the DB schema with UBR (means that ensure the maintenance of a special log-table of the modified data);
- w_{18}—masking data of tables based on the approach described in [79];
- w_{19}—masking of stored objects using the means provided by the DBMS, as well as based on the approach described in [77];
- w_{20}—using transparent data encryption (TDE) and cryptographically strong primitives built into the DBMS as well as national encryption standards (for example, the symmetric block cipher "Kalyna" from the national standard of Ukraine DSTU 7624: 2014);
- w_{21}—timely installation of critical updates, monitoring of the cryptographic strength of the used implementations of encryption algorithms and randomness of numbers generated by pseudo-random number generators (PRNG) that meet the specified requirements;
- w_{22}—database administrator tools built into the DBMS, as well as specially developed scripts that simplify the work of the DBA;
- w_{23}—detailed documentation on the DBMS, DB with a description of all its corresponding elements, their use, including all the main components of the DB schema with UBR;
- w_{24}—audit, blocking a response if the number of requests is incorrect.

In accordance with the above technique and the accepted assumptions, let us estimate the potential value of the database security with the universal basis of relations and compare it with the security of traditional relational databases. For this purpose, on the basis of the above-defined list of main objects, threats, vulnerabilities, available security measures, summarizing the experience of operating and building protection systems for relational databases and databases with UBR, we determine the values of the corresponding components of security barriers ($P_l = f(P_{t_i}, P_{\gamma_\psi})$, L_l, R_l). For this, we will correlate them with the quadruple corresponding most significant (from the point of view of the issues under consideration) elements of barrier $b_l = (t_i, \gamma_\psi, o_j, w_k)$ in the basic security system. Figure 6 shows a fragment of a database security system model in the form of a directed graph.

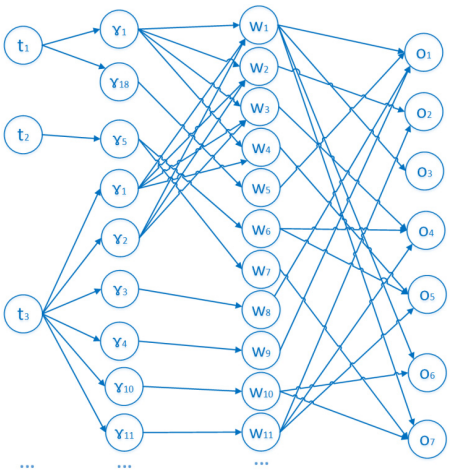

Figure 6. Fragment of the database security system model in the form of a graph.

Table 3 shows a fragment of the evaluation results of the main components of security barriers and resistance (strength) of each of them.

Table 3. Fragment of the evaluation results of the main components of security barriers.

Barrier No.	Threat (t)	$\frac{P_t^{\text{verbal}}}{P_t^{\text{quant}}}$	Vulnerability (γ)	$\frac{P_\gamma^{\text{verbal}}}{P_\gamma^{\text{quant}}}$	Security Measure (w)	$\frac{R^{\text{UBR}_{\text{verbal}}}}{R^{\text{UBR}_{\text{quant}}}}$	$\frac{R^{\text{RDB}_{\text{verbal}}}}{R^{\text{RDB}_{\text{quant}}}}$	Object (o)	Rr^{UBR}	Rr^{RDB}
1	t_1	"M"/0.4	γ_1	"M"/0.5	w_1	"H"/0.8	"H"/0.8	o_1	0.008	0.008
2	t_1	"M"/0.4	γ_1	"M"/0.5	w_2	"H"/0.85	"H"/0.8	o_2	0.006	0.008
3	t_1	"M"/0.4	γ_1	"M"/0.5	w_3	"H"/0.85	"H"/0.8	o_4	0.006	0.008
4	t_1	"L"/0.1	γ_1	"M"/0.5	w_4	"H"/0.8	"L"/0	o_5	0.002	0.01
5	t_1	"M"/0.4	γ_1	"M"/0.5	w_1	"H"/0.8	"H"/0.8	o_3	0.008	0.008
6	t_1	"M"/0.4	γ_1	"M"/0.5	w_1	"H"/0.8	"H"/0.8	o_6	0.008	0.008
7	t_1	"M"/0.4	γ_1	"M"/0.5	w_1	"H"/0.8	"H"/0.8	o_7	0.008	0.008
8	t_1	"M"/0.4	γ_{18}	"M"/0.5	w_5	"H"/0.8	"H"/0.8	o_1	0.008	0.008
9	t_2	"M"/0.4	γ_5	"H"/0.85	w_6	"H"/0.8	"M"/0.6	o_4	0.0136	0.0272
10	t_2	"M"/0.4	γ_5	"H"/0.85	w_6	"H"/0.8	"M"/0.6	o_5	0.0136	0.0272
11	t_2	"M"/0.4	γ_5	"H"/0.85	w_7	"H"/0.8	"M"/0.6	o_7	0.0136	0.0272
12	t_3	"M"/0.4	γ_1	"M"/0.5	w_1	"H"/0.8	"H"/0.8	o_1	0.008	0.008
13	t_3	"M"/0.4	γ_1	"M"/0.5	w_2	"H"/0.85	"H"/0.8	o_2	0.006	0.008
14	t_3	"M"/0.4	γ_1	"M"/0.5	w_3	"H"/0.85	"H"/0.8	o_4	0.006	0.008
15	t_3	"L"/0.1	γ_1	"M"/0.5	w_4	"H"/0.8	"L"/0	o_5	0.002	0.01
16	t_3	"M"/0.4	γ_2	"M"/0.5	w_1	"H"/0.8	"H"/0.8	o_1	0.008	0.008
17	t_3	"M"/0.4	γ_2	"M"/0.5	w_2	"H"/0.85	"H"/0.8	o_2	0.006	0.008
18	t_3	"M"/0.4	γ_2	"M"/0.5	w_3	"H"/0.85	"H"/0.8	o_4	0.006	0.008
19	t_3	"M"/0.4	γ_2	"M"/0.5	w_1	"H"/0.8	"H"/0.8	o_3	0.008	0.008
20	t_3	"M"/0.4	γ_2	"M"/0.5	w_1	"H"/0.8	"H"/0.8	o_6	0.008	0.008
21	t_3	"M"/0.4	γ_2	"M"/0.5	w_1	"H"/0.8	"H"/0.8	o_7	0.008	0.008
22	t_3	"M"/0.4	γ_3	"M"/0.5	w_8	"H"/0.8	"H"/0.8	o_1	0.008	0.008
23	t_3	"M"/0.4	γ_4	"M"/0.5	w_9	"M"/0.4	"M"/0.4	o_1	0.024	0.024
24	t_3	"M"/0.4	γ_{10}	"M"/0.5	w_{10}	"H"/0.9	"M"/0.4	o_6	0.004	0.024
25	t_3	"M"/0.4	γ_{10}	"M"/0.5	w_{10}	"H"/0.9	"M"/0.4	o_7	0.004	0.024
26	t_3	"M"/0.4	γ_{11}	"M"/0.5	w_{11}	"H"/0.8	"H"/0.8	o_2	0.008	0.008

Where P_t^{verbal} is the verbal value of the linguistic variable—the likelihood of a threat occurrence P_t; P_t^{quant} is the numerical value of the likelihood P_t; P_γ^{verbal} is the verbal value of the linguistic variable—the likelihood of exploiting the vulnerability P_γ; P_γ^{quant} is the numerical value of the likelihood P_γ; $R^{\text{UBR}_{\text{verbal}}}$ is the verbal value of the linguistic variable—the degree of security measure resistance R ($R = 1 - P^{ov}$) for the database with UBR; $R^{\text{UBR}_{\text{quant}}}$ is the numerical value of the degree of security measure resistance R for the DB with UBR; $R^{\text{RDB}_{\text{verbal}}}$ is the verbal value of the linguistic variable—the degree of security measure resistance R for the traditional database; $R^{\text{RDB}_{\text{quant}}}$ is the numerical value of the degree of security measure resistance R for the traditional database;Rr^{UBR} is the numerical value of the residual risk value for the DB with UBR; Rr^{RDB} is the numerical value of the residual risk value for the traditional database.

In accordance with the obtained results of assessments of the main components of security barriers and residual risk values (Figure 7), under the given assumptions, in accordance with Equation (2) the values of the security quantities of traditional databases and DB with UBR were calculated. All obtained values are presented in Figure 8 in the form of a corresponding diagram.

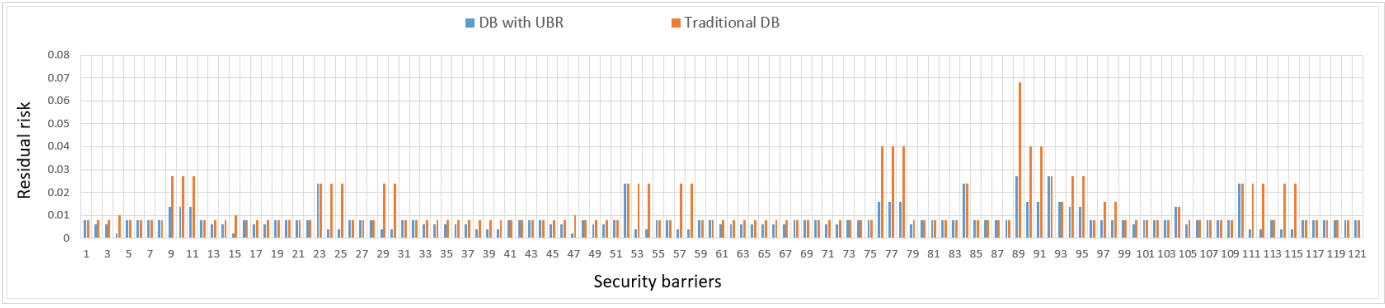

Figure 7. Diagram of residual risk values of the compared databases.

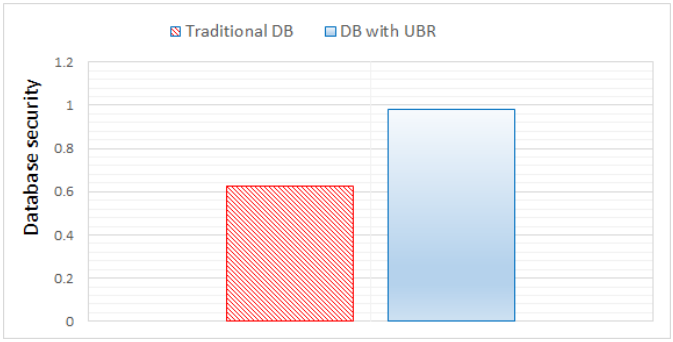

Figure 8. Diagram of security values of compared databases.

Based on the results obtained, a general conclusion was made about the greater effectiveness of the solutions proposed within the framework of the database schema with UBR in comparison with the existing solutions implemented within the framework of traditional relational databases. Taking into account the obtained quantitative assessment, the usage of the proposed solutions will increase the effectiveness of protection (as the reciprocal of the total residual risk) of databases built on the basis of the schema with the universal basis of relations by more than 1.5 times.

An analysis of various countermeasures aimed at ensuring security shows that many problems with the protection of data stored in a database often arise not due to a lack of research, the presence of theoretically developed models, methods, but due to insufficient security in the corresponding specific database implementation or applications working with it. In this sense, DBs with UBR have an advantage, since they are not designed from scratch every time and are not subject to significant modification during reengineering, including in terms of their security. The schema of such databases invariant to various SDs has already been developed, including special measures to ensure security (in the form of appropriate methods, implemented objects). This schema can be installed on the platform of some relational DBMS. When expanding the data set of the simulated SDs in a DB with UBR, unlike traditional relational databases, new basic relations, attributes, keys and other schema objects, including those ensuring its security, are not created, but a new record is simply added to one of the existing basic schema relations. This makes it possible, when reengineering databases built based on this schema, to simplify the process of their adaptation to dynamic changes in subject domains.

The results obtained indicate the objectivity of the developed technique. It is natural that if a database with UBR contains original solutions aimed at improving security, but traditional relational databases do not have them, then the resulting gain in improving the protection effectiveness is predictable.

In the future, it is planned to compare the proposed technique with other approaches.

6. Conclusions

Having analyzed and summarized various approaches and achievements in the field of assessing the security of information systems, the authors of the paper have developed a

technique for evaluating the security of relational databases. The proposed technique is the result of a comprehensive combination of the enhanced Clements–Hoffman model, defined integral security metric, the provisions of the theory of fuzzy sets and risk. The Clements–Hoffman model has been extended to a 6-tuple (sextuple). The expansion was carried out by supplementing the model with a set of vulnerabilities (weaknesses) of objects, as a separate objectively existing category. This made it possible to evaluate both the likelihood of an unwanted incident and the database security as a whole more adequately. In addition, in the process of developing the enhanced model, some of its significant components were concretized. Namely:

- Identified the main significant threats to the security of databases;
- The main protected objects are determined taking into account the dual nature of the relational database system and the various degrees of detail of its components.

As an integral metric of database security, the reciprocal of the total residual risk was determined, which is essentially an insecurity measure of an asset. This made it possible to quantify the security of databases. The constituent components that determine the residual risk and characterize the strength of a certain security barrier are presented in the form of certain linguistic variables.

The proposed technique, in contrast to a number of known ones, is based on the time-tested provisions of the theories of probability, fuzzy sets, and risk, allowing at the same time to quite simply, comprehensively and quantitatively evaluate the security of RDBs. The explainable, non-contradictory results of evaluating the security of relational databases designed using various technologies with various security measures presented in the paper indicate the objectivity of the developed technique. If the database with UBR contains original solutions aimed at improving security, but traditional relational databases do not have them, then the gain in improving the protection effectiveness is natural. At that, the very value of the obtained advantage is also explainable and plausible. This is all very important. First of all, from the point of view of the possibility and expediency of practical application of the developed technique in the future for evaluating and comparing the security of various RDBs. Due to its flexibility, the proposed technique can also be used to evaluate the security of databases with various data models.

Author Contributions: Conceptualization, V.Y.; methodology, V.Y. and V.V.; software, V.Y., V.V. and S.A.R.; formal analysis, M.K.; investigation, V.Y., M.Y., V.V. and S.A.R.; writing—original draft preparation, V.Y., M.Y. and S.A.R.; writing—review and editing, V.Y., M.K. and M.Y.; funding acquisition, M.K. All authors have read and agreed to the published version of the manuscript.

Funding: The research work reported in this paper was, in part, supported by the University of Bielsko-Biala, Poland, under program no. K18/1b/UPBJ/2019-2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not available.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Abadi, D.; Agrawal, R.; Ailamaki, A.; Balazinska, M.; Bernstein, P.A.; Carey, M.J.; Chaudhuri, S.; Dean, J.; Doan, A.; Franklin, M.J.; et al. The Beckman Report on Database Research. *ACM SIGMOD Rec.* **2014**, *43*, 61–70. [CrossRef]
2. Abadi, D.; Ailamaki, A.; Andersen, D.; Bailis, P.; Balazinska, M.; Bernstein, P.; Boncz, P.; Chaudhuri, S.; Cheung, A.; Doan, A.; et al. The Seattle Report on Database Research. *ACM SIGMOD Rec.* **2020**, *48*, 44–53. [CrossRef]
3. ISO/IEC 25010:2011 Systems and Software Engineering. Systems and Software Quality Requirements and Evaluation (SQuaRE). System and Software Quality Models. Available online: https://www.iso.org/standard/35733.html/ (accessed on 21 September 2021).
4. Latham, D.C. *Department of Defense Trusted Computer System Evaluation Criteria*; Department of Defense: Arlington, VA, USA, 1986. Available online: http://csrc.nist.gov/publications/history/dod85.pdf (accessed on 21 September 2021).

5. Commission of the European Communities. Information Technology Security Evaluation Criteria (ITSEC): Provisional Evaluation Criteria. Document COM(90) 314, Version 1.2. Available online: https://www.ssi.gouv.fr/uploads/2015/01/ITSEC-uk.pdf (accessed on 21 September 2021).
6. ISO/IEC 21827:2008 Information Technology. Security Techniques. Systems Security Engineering. Capability Maturity Model® (SSE-CMM®). Available online: https://www.iso.org/obp/ui/#iso:std:iso-iec:21827:ed-2:v1:en (accessed on 21 September 2021).
7. Common Criteria for Information Technology Security Evaluation Part 1: Introduction and General Model. Version 3.1 Revision 5 CCMB-2017-04-001. Available online: https://www.commoncriteriaportal.org/files/ccfiles/CCPART1V3.1R5.pdf (accessed on 21 September 2021).
8. Jansen, W.; Gallagher, P.D. NISTIR 7564. Directions in Security Metrics Research. Available online: https://nvlpubs.nist.gov/nistpubs/legacy/ir/nistir7564.pdf (accessed on 21 September 2021).
9. Juma, J.; Makupi, D. Understanding Database Security Metrics: A Review. *Mara Int. J. Sci. Res. Publ.* **2017**, *1*, 40–48.
10. NIST Special Publication 800-55 Revision 1. 2008. Available online: https://csrc.nist.gov/publications/detail/sp/800-55/rev-1/final (accessed on 21 September 2021).
11. Sandhu, R.S.; Jajodia, S. Data and Database Security and Controls. In *Handbook of Information Security Management*; Auerbach Publishers: Boca Raton, FL, USA, 1993; pp. 481–499.
12. Date, C.J. *An Introduction to Database Systems*, 8th ed.; Pearson Education Inc.: New York, NY, USA, 2004.
13. Neto, A.A.; Vieira, M.; Madeira, H. An appraisal to assess the security of database configurations. In Proceedings of the Second International Conference on Dependability, Athens, Greece, 18–23 June 2009; pp. 73–80. [CrossRef]
14. Oracle. Database Security Assessment Tool User Guide. Available online: https://docs.oracle.com/en/database/oracle/security-assessment-tool/2.2.2/satug/index.html#UGSAT-GUID-C7E917BB-EDAC-4123-900A-D4F2E561BFE9 (accessed on 21 September 2021).
15. Yesin, V.I.; Karpinski, M.P.; Yesina, M.V.; Vilihura, V.V. Formalized Representation for the Data Model with the Universal Basis of Relations. *Int. J. Comput.* **2019**, *18*, 453–460. [CrossRef]
16. Savola, R.M. A Security Metrics Taxonomization Model for Software-Intensive Systems. *J. Inf. Process. Syst.* **2009**, *5*, 197–206. [CrossRef]
17. Savola, R.M. Towards Measurement of Security Effectiveness Enabling Factors in Software Intensive Systems. *Lect. Notes Softw. Eng.* **2014**, *2*, 104–109. [CrossRef]
18. Pendleton, M.; Garcia-Lebron, R.; Cho, J.-H.; Xu, S. A Survey on Systems Security Metrics. *ACM Comput. Surv.* **2017**, *49*, 1–35. [CrossRef]
19. Bernik, I.; Prislan, K. Measuring Information Security Performance with 10 by 10 Model for Holistic State Evaluation. *PLoS ONE* **2016**, *11*, e0163050. [CrossRef]
20. Kong, H.-K.; Kim, T.-S.; Kim, J. An analysis on effects of information security investments: A BSC perspective. *J. Intell. Manuf.* **2010**, *23*, 941–953. [CrossRef]
21. Jacobs, M.A. Complexity: Toward an empirical measure. *Technovation* **2013**, *33*, 111–118. [CrossRef]
22. Savola, R.M. Quality of security metrics and measurements. *Comput. Secur.* **2013**, *37*, 78–90. [CrossRef]
23. Yasasin, E.; Schryen, G. Requirements for IT Security Metrics—An Argumentation Theory Based Approach. In *European Conference on Information Systems—ECIS*; Completed Research Paper; Paper 208; ECIS: Münster, Germany, 2015.
24. Katt, B.; Prasher, N. Quantitative security assurance metrics: REST API case studies. In Proceedings of the 12th European Conference on Software Architecture: Companion Proceedings, Madrid, Spain, 24–28 September 2018; pp. 1–7.
25. Sanders, W.H. Quantitative Security Metrics: Unattainable Holy Grail or a Vital Breakthrough within Our Reach? *IEEE Secur. Priv. Mag.* **2014**, *12*, 67–69. [CrossRef]
26. Sarmah, S. Database Security—Threats & Prevention. *Int. J. Comput. Trends Technol. (IJCTT)* **2019**, *67*, 46–50.
27. Awadallah, R.; Samsudin, A. Using Blockchain in Cloud Computing to Enhance Relational Database Security. *IEEE Access* **2021**, *9*, 137353–137366. [CrossRef]
28. Pfleeger, C.P.; Pfleeger, S.L.; Margulies, J. *Security in Computing*, 5th ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2015.
29. Mousa, A.; Karabatak, M.; Mustafa, T. Database security threats and challenges. In Proceedings of the 8th International Symposium on Digital Forensics and Security (ISDFS), Beirut, Lebanon, 1–2 June 2020; pp. 1–5.
30. Connolly, T.M.; Begg, C.E. *Database Systems: A Practical Approach to Design, Implementation, and Management*; Pearson Education Limited: London, UK, 2015.
31. Kulkarni, S.; Urolagin, S. Review of attacks on databases and database security techniques. *Int. J. Emerg. Technol. Adv. Eng.* **2012**, *2*, 2250–2459.
32. Mishra, S.; Morris, R.; Chasalow, L. Information security effectiveness: A research framework. *Issues Inf. Syst.* **2011**, *12*, 246–255.
33. Fabian, B.; Gürses, S.; Heisel, M.; Santen, T.; Schmidt, H. A comparison of security requirements engineering methods. *Requir. Eng.* **2009**, *15*, 7–40. [CrossRef]
34. Hoffman, L.J. *Modern Methods for Computer Security and Privacy*; Prentice-Hall, Inc.: Englewood Cliffs, NJ, USA, 1977.
35. Hoffman, L.J.; Clements, D. *Fuzzy Computer Security Metrics: A Preliminary Report*; Electronics Research Laboratory, College of Engineering University of California: Berkeley, CA, USA, 1977. Available online: https://www2.eecs.berkeley.edu/Pubs/TechRpts/1977/ERL-m-77-6.pdf (accessed on 21 September 2021).

36. Anishchanka, U.V.; Krishtophic, A.M. Methods of evaluating the effectiveness of protecting the assets in information technology objects. *Informatika* **2004**, *3*, 95–105.
37. Maslova, N.A. Methods for assessing the effectiveness of information systems protection systems. *Artif. Intell.* **2008**, *4*, 253–264.
38. Domarev, V.V. *Information Technology Security. Systems Approach*; OOO «TID «DS»: Kyiv, Ukraine, 2004.
39. Hoffmann, R.; Kiedrowicz, M.; Stanik, J. Evaluation of information safety as an element of improving the organization's safety management. In Proceedings of the 20th International Conference on Circuits, Systems, Communications and Computers (CSCC 2016), MATEC Web of Conferences, Corfu Island, Greece, 14–17 July 2016; Volume 76, p. 04011. [CrossRef]
40. Kiedrowicz, M.; Stanik, J. Method for assessing efficiency of the information security management system. In Proceedings of the 22nd International Conference on Circuits, Systems, Communications and Computers (CSCC 2018), MATEC Web of Conferences, Majorca, Spain, 14–17 July 2018; Volume 210, p. 04011. [CrossRef]
41. Lee, M.-C. Information Security Risk Analysis Methods and Research Trends: AHP and Fuzzy Comprehensive Method. *Int. J. Comput. Sci. Inf. Technol.* **2014**, *6*, 29–45. [CrossRef]
42. ISO/IEC 15408-1:2009. Information Technology. Security Techniques. Evaluation Criteria for IT Security. Part 1: Introduction and General Model. Available online: https://www.iso.org/standard/50341.html (accessed on 21 September 2021).
43. ISO/IEC 27001:2013. Information Technology. Security Techniques. Information Security Management Systems. Requirements. Available online: https://www.iso.org/standard/54534.html (accessed on 21 September 2021).
44. ISO/IEC 27004:2016. Information Technology. Security Techniques. Information Security Management. Monitoring, Measurement, Analysis and Evaluation. Available online: https://www.iso.org/standard/64120.html (accessed on 21 September 2021).
45. Rohilla, S.; Mittal, P.K. Database Security: Threads and Challenges. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* **2013**, *3*, 810–813.
46. Imperva Whitepaper. Top Ten Database Security Threats. 2015. Available online: https://informationsecurity.report/Resources/Whitepapers/e763d022-6ee4-4215-9efd-1896b0d9c381_wp_topten_database_threats%20imperva.pdf (accessed on 21 September 2021).
47. Imperva Whitepaper. Top 5 Database Security Threats. 2016. Available online: https://www.imperva.com/docs/gated/WP_Top_5_Database_Security_Threats.pdf (accessed on 21 September 2021).
48. DB-Engines Ranking. Available online: https://db-engines.com/en/ranking (accessed on 21 September 2021).
49. TOPDB Top Database Index. Available online: https://pypl.github.io/DB.html (accessed on 21 September 2021).
50. Adrian, M.; Feinberg, D.; Heudecker, N. Gartner Magic Quadrant for Operational Database Management Systems. ID G00376881. Available online: https://www.gartner.com/en/documents/3975492/magic-quadrant-for-operational-database-management-syste (accessed on 21 September 2021).
51. Adrian, M.; Feinberg, D.; Greenwald, R.; Ronthal, A.; Cook, H. Critical Capabilities for Cloud Database Management Systems for Operational Use Cases. ID G00468197. Available online: https://www.oracle.com/explore/adw-ocom/gartner-cloud-database-management/?source=:ow:o:p:mt:::RC_WWMK200720P00100:Gartnerdatabase&intcmp=:ow:o:p:mt:::RC_WWMK200720P00100:Gartnerdatabase&lb-mode=overlay (accessed on 21 September 2021).
52. Groff, J.; Weinberg, P.; Oppel, A. *SQL: The Complete Reference*, 3rd ed.; McGraw-Hill, Inc.: New York, NY, USA, 2010.
53. Talabis, M.; Martin, J. *Information Security Risk Assessment Toolkit Practical Assessments through Data Collection and Data Analysis*; Syngress: Waltham, MA, USA, 2012.
54. Whitman, M.E.; Mattord, H.J. *Principles of Information Security*, 6th ed.; Cengage Learning: Boston, MA, USA, 2017.
55. *NIST Special Publication 800-53 Revision 5. Security and Privacy Controls for Information Systems and Organizations*; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2020. [CrossRef]
56. ISO/IEC 27002:2013 Information Technology. Security Techniques. Code of Practice for Information Security Controls. Available online: https://www.iso.org/standard/54533.html (accessed on 21 September 2021).
57. ISO/IEC 27000:2018 Information Technology. Security Techniques. Information Security Management Systems. Overview and Vocabulary. Available online: https://www.iso.org/standard/73906.html (accessed on 21 September 2021).
58. Astakhov, A.M. *The Art of Information Risk Management*; DMK Press: Moscow, Russia, 2010.
59. MITRE. CWE VIEW: Research Concepts. Available online: https://cwe.mitre.org/data/definitions/1000.html (accessed on 21 September 2021).
60. Astakhov, A. *Analysis of the Security of Corporate Systems*; Open System DBMS: Moscow, Russia, 2002; pp. 7–8. Available online: https://www.osp.ru/os/2002/07-08/181720 (accessed on 21 September 2021).
61. Averchenkov, V.I.; Rytov, M.Y.; Gainulin, T.R. Optimization of the choice of the composition of the means of engineering and technical information protection based on the Clements-Hoffman model. *Bull. Bryansk State Tech. Univ.* **2008**, *1*, 61–66.
62. Karpychev, V.Y. Economic analysis of normative and technical support of information security. *Econ. Anal. Theory Pract.* **2011**, *35*, 2–18.
63. Burtescu, E. Database security—Attacks and control methods. *J. Appl. Quant. Methods* **2009**, *4*, 449–454.
64. Arkhipov, A.E. Expert-analytical assessment of information risks and the efficiency level of the information protection system. *Radio Electron. Comput. Sci. Control* **2009**, *2*, 111–115.
65. Zadeh, L.A. The concept of a linguistic variable and its application to approximate reasoning—I. *Inf. Sci.* **1975**, *8*, 199–249. [CrossRef]
66. Petrenko, S.A.; Simonov, S.V. *Information Risk Management. Economically Justified Safety*; DMK Press: Moscow, Russia, 2004.

67. NIST Special Publication 800-30 Revision 1. Guide for Conducting Risk Assessments. Available online: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-30r1.pdf (accessed on 21 September 2021).
68. Kornienko, A.A.; Nikitin, A.B.; Diasamidze, S.V.; Kuz'menkova, E.Y. Simulation of computer attacks on distributed software. *Bull. St. Petersburg State Transp. Univ.* **2018**, *15*, 613–628.
69. FSTEC Russia. Methodology for Determining Current Threats to the Security of Personal Data during Their Processing in Personal Data Information Systems. Available online: https://fstec.ru/tekhnicheskaya-zashchita-informatsii/dokumenty/114-spetsialnye-normativnye-dokumenty/380-metodika-opredeleniya-aktualnykh-ugroz-bezopasnosti-personalnykh-dannykh-pri-ikh-obrabotke-v-informatsionnykh-sistemakh-personalnykh-dannykh-fstek-rossii-2008-god (accessed on 21 September 2021).
70. Leonenkov, A.V. *Fuzzy Modeling in MATLAB and Fuzzytech*; BHV Petersburg: Sankt-Petersburg, Russia, 2005.
71. Kruglov, V.V.; Dli, M.I.; Golunov, R.Y. *Fuzzy Logic and Artificial Neural Networks*; Fizmatlit: Moscow, Russia, 2001.
72. Piegat, A. *Fuzzy Modeling and Control*; Physica-Verlag: Heidelberg, Germany, 2001.
73. Yesin, V.I.; Vilihura, V.V. Method for Development of Databases Easily Adaptable to Variations in The Subject Domain. *Telecommun. Radio Eng.* **2019**, *78*, 595–605. [CrossRef]
74. Yesin, V.I.; Karpinski, M.; Yesina, M.V.; Vilihura, V.V.; Veselska, O.; Wieclaw, L. Approach to Managing Data From Diverse Sources. In Proceedings of the 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Metz, France, 18–21 September 2019; pp. 1–6. [CrossRef]
75. Franklin, M.; Halevy, A.; Maier, D. From databases to dataspaces: A new abstraction for information management. *ACM SIGMOD Rec.* **2005**, *34*, 27–33. [CrossRef]
76. Yesin, V.I.; Yesina, M.V.; Rassomakhin, S.G.; Karpinski, M. Ensuring Database Security with the Universal Basis of Relations. In *CISIM 2018: Computer Information Systems and Industrial Management*; Lecture Notes in Computer Science, 11127; Saeed, K., Homenda, W., Eds.; Springer: Cham, Switzerland, 2018; Chapter 42; pp. 510–522. [CrossRef]
77. Yesin, V.; Karpinski, M.; Yesina, M.; Vilihura, V.; Warwas, K. Hiding the Source Code of Stored Database Programs. *Information* **2020**, *11*, 576. [CrossRef]
78. Yesin, V.I.; Yesina, M.V.; Vilihura, V.V.; Yesin, V. Monitoring the integrity and authenticity of stored database objects. *Telecommun. Radio Eng.* **2020**, *79*, 1029–1054. [CrossRef]
79. Yesin, V.; Vilihura, V.; Yesin, V. Some approach to data masking as means to counteract the inference threat. *Radiotekhnika* **2019**, *3*, 113–130. [CrossRef]
80. Yesin, V.; Karpinski, M.; Yesina, M.; Vilihura, V.; Warwas, K. Ensuring Data Integrity in Databases with the Universal Basis of Relations. *Appl. Sci.* **2021**, *11*, 8781. [CrossRef]
81. Sadalage, P.J.; Fowler, M. *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*; Pearson Education: London, UK, 2013.
82. Harrison, M.A.; Ruzzo, W.L.; Ullman, J.D. Protection in operating systems. *Commun. ACM* **1976**, *19*, 461–471. [CrossRef]
83. Lipton, R.J.; Snyder, L. A Linear Time Algorithm for Deciding Subject Security. *J. ACM* **1977**, *24*, 455–464. [CrossRef]
84. Clark, D.D.; Wilson, D.R. A Comparison of Commercial and Military Computer Security Policies. In Proceedings of the IEEE Symposium on Research in Security and Privacy (SP'87), Oakland, CA, USA, 27–29 April 1987; IEEE Press: Oakland, CA, USA, 1987; pp. 184–193.
85. Bashir, I. *Mastering Blockchain: Distributed Ledger Technology, Decentralization, and Smart Contracts Explained*, 2nd ed.; Packt Publishing: Birmingham, UK, 2018.
86. Antonopoulos, A.M. *Mastering Bitcoin: Programming the Open Blockchain*, 2nd ed.; O'Reilly Media: Sebastopol, CA, USA, 2017.
87. Cotner, C.; Miller, R.L. International Business Machines Corporation. Row-Level Security in a Relational Database Management System. US Patent 8,478,713 B2, 16 January 2018. N 15/343,568.
88. Meier, A.; Kaufmann, M. *SQL & NoSQL Databases. Databases Models, Languages, Consistency Options and Architectures for Big Data Management*; Springer Fachmedien: Wiesbaden, Germany, 2019. [CrossRef]
89. Harrison, G. *Next Generation Databases: NoSQL, NewSQL, and Big Data*; Apress: Berkeley, CA, USA, 2015.
90. Pavlo, A.; Aslett, M. What's Really New with NewSQL? *ACM SIGMOD Rec.* **2016**, *45*, 45–55. [CrossRef]
91. Garcia-Molina, H.; Ullman, J.D.; Widom, J. *Database Systems: The Complete Book*, 2nd ed.; Pearson Prentice Hall: Upper Saddle River, NJ, USA, 2009.