Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 418535, 19 pages
http://dx.doi.org/10.1155/2015/418535

Research Article

On Security Management: Improving Energy Efficiency, Decreasing Negative Environmental Impact, and Reducing Financial Costs for Data Centers

Katarzyna Mazur,[1] Bogdan Ksiezopolski,[1,2] and Adam Wierzbicki[2]

[1] *Institute of Computer Science, Maria Curie-Sklodowska University, Plac M. Curie-Sklodowskiej 5, 20-031 Lublin, Poland*
[2] *Polish-Japanese Academy of Information Technology, Koszykowa 86, 02-008 Warsaw, Poland*

Correspondence should be addressed to Bogdan Ksiezopolski; bogdan.ksiezopolski@acm.org

Received 26 March 2015; Revised 19 May 2015; Accepted 26 May 2015

Academic Editor: Bao Rong Chang

Security management is one of the most significant issues in nowadays data centers. Selection of appropriate security mechanisms and effective energy consumption management together with caring for the environment enforces a profound analysis of the considered system. In this paper, we propose a specialized decision support system with a multilevel, comprehensive analysis scheme. As a result of the extensive use of mathematical methods and statistics, guidelines and indicators returned by the proposed approach facilitate the decision-making process and conserve decision-maker's time and attention. In the paper we utilized proposed multilevel analysis scheme to manage security-based data flow in the example data center. Determining the most secure, energy-efficient, environmental friendly security mechanisms, we implemented the role-based access control method in Quality of Protection Modeling Language (QoP-ML) and evaluated its performance in terms of mentioned factors.

1. Introduction

The challenges faced by companies working in nowadays complex IT environments pose the need for comprehensive and dynamic systems to cope with the security requirements [1–3]. Security planning cannot answer all questions: we must take a step further and discuss a model for security management. One of the possible approaches to deal with this problem is to use the decision support system that is capable of supporting decision-making activities. Good decisions are usually result of detailed and precise examination of input parameters and comprehensive analysis of all the available alternatives. In order to solve a given problem, one should formulate it by using standardized methodologies and collect all the information that are relevant to the final decision. As the action of decision-making is usually a repeatable process, it is reasonable to embrace a much more comprehensive view of decision-making. In general, decision cycle consists of four main, ordered phases: *problem definition* (a decision situation that may deal with some difficulty or with an opportunity), *model construction* (in which one needs to describe the real world problem using specialized tools, designed specifically for this purpose), *identification and evaluation* (where the possible solutions to the modeled problems are identified and evaluated), and finally the *recommendation* and *implementation* stages, in which potential solutions are examined, compared, and chosen.

Since nowadays IT environments are combination of physical, virtual, and cloud infrastructures, legacy and modern technology components, they pose a need for going deeper, that is, performing more detailed, more complex, and integrated types of analysis. To meet this requirement, we propose the foundations of our decision support system for complex IT environments.

By definition, decision support system (DSS) is a conceptual framework which supports decision-making, usually by modeling problems and employing quantitative models for solution analysis. DSS is an interactive, programmed, computer-based system which helps decision makers to use available assets and resources, identify and solve problems, complete decision process tasks, and make decisions. Being a multilevel, model-driven approach, our DSS lets one describe, examine, and analyse complex IT environments. Additionally, thanks to the extensive use of abstract models, it helps determine differences between distinct options. Such an approach is particularly helpful in the case of security-based data flow management, where one is capable of modeling miscellaneous scenarios (which differ in utilized security mechanisms) and assessing their quality. Managing interactions that occur across the complex secure systems characterized by the high level of dynamics, determining proper role definitions and permissions assignments, quantification of the authorization level of users allowed to perform various actions (based on the scope of the assigned role), is also crucial from the role-based access control point of view.

The main contributions of this paper are summarized as follows.

(1) We introduced the foundations of the comprehensive decision support system, together with presenting and describing its essential part, the multilevel analysis scheme utilized for the evaluation of complex secure systems.

(2) We extended previous studies on system's performance to a broader context: we discussed and examined new types of analyses, the financial and carbon dioxide emissions analyses (both being components of our new, utilized analysis scheme).

(3) We implemented proposed analysis techniques as modules utilized in Automated Quality of Protection Analysis (AQoPA) tool (which can be downloaded from the web page of the QoP-ML Project [4]); such solution gives the possibility to compare available security approaches in terms of time, energy, quality of protection, finance, and environmental impact.

(4) We illustrated the introduced approach by presenting the case study of the example data center with role-based access control implemented, since it has been widely adopted in secure IT environments and has been studied in many different contexts over the years (e.g., the economic [5] or security related studies [6]).

We organized the rest of this paper in the following way: in Section 2 we present the related work. Section 3 contains a brief overview of the Quality of Protection Modeling Language which we used in our research. Moving on to the next point, we demonstrate the foundations of the proposed decision support system and bring closer the multilevel analysis scheme. In Section 5, we introduce the CO_2 emission analysis phase, which tries to answer questions about the minimization of the ecological footprint of IT products and services. Moreover, we discuss the financial aspect of the data center performance and formalize its evaluation. To demonstrate the use of the proposed approach, in Section 6, we present the case study of the role-based access control as the example of the security-based data flow management. We prepare a RBAC model in Quality of Protection Modeling Language and assess its quality in terms of time, energy, quality of protection, finance, and environmental impact. Further, we discuss the results and finally summarize our work in Section 7.

2. Related Work

In the literature ([16, 17]) one can find miscellaneous types of decision support systems. Among them one can enumerate model-driven DSS [18], data-driven DSS, communication-driven DSS, document-driven DSS, and knowledge-driven DSS [19]. The model-driven decision support system approach is the most complex from the existing decision support system types. It deals with statistical, financial optimization or simulation models and uses input parameters and data provided by users to assist stakeholders in the decision-making process. In data-driven DSS, the data itself is the most significant part of the considered approach. Having easy access to a large amount of accurate, well-organized data sets stored in databases or data warehouses, the system seeks for specific information and reports retrieved data to users. With the rapid growth of the interconnected network environments, it became possible to utilize available network and communications technologies for the need of communication-driven decision support systems. Tools like groupware, video conferencing, and computer-based bulletin boards facilitate decision relevant collaboration and communication. Document-driven DSS is the most commonly used type of decision support system. It is focused on managing, retrieving, and manipulating unstructured information in a variety of electronic formats. To suggest possible solutions to a given problem, knowledge-based DSS makes use of expert systems and artificial intelligence. Simulating reasoning, explaining the logic behind its conclusion, knowledge-based DSS assists a decision-maker in an area where the specialized knowledge is required. The major goal of the security-driven data flow management is to simplify authorization management and review.

Having the ability of modeling various access control requirements and facilitating security administration process, RBAC became the object of the study of many researchers. In the literature [20, 21], one can find plenty of RBAC implementations. Preparing RBAC models in SecureUML [22] and UMLsec [23] authors usually focus on their economic or security aspects, omitting the influence of distinct authorization levels on system's efficiency. However, role-based access control has an undeniable impact on performance and should be implemented carefully in order to provide the required level of security together with energy efficiency and assurance of the security trade-offs. To address this issue, many modeling languages and tools have been proposed. Among them one can enumerate UMLsec and SecureUML presented by the researchers in [20]. Using the mentioned approaches, one is able to model and verify secure systems, either preexisting or those under construction. Nevertheless, introduced solutions focus on developing secure infrastructure or determining system's efficiency, rather than examining security and performance concerns at the same time. The traditional approach assumes that implementation of the strongest security mechanisms makes the system as secure as possible. Unfortunately, such reasoning can lead

Table 1: Established characteristics of QoP models.

	QA	E	Con	EE	H	Com	PE	FE	EA
Agarwal and Wang [7]	√	—	—	—	√	√	√	—	—
Ksiezopolski and Kotulski [8]	√	√	—	—	√	√	—	—	—
LeMay et al. [9]	—	√	√	—	—	—	—	—	—
Lindskog [10]	√	—	√	—	—	—	√	—	—
Luo et al. [11]	√	—	—	—	√	√	√	—	—
Ong et al. [12]	√	—	—	—	—	—	—	—	—
Petriu et al. [13]	—	√	√	—	—	√	√	—	—
Schneck and Schwan [14]	√	—	√	—	—	—	√	—	—
Sun and Kumar [15]	√	—	—	—	—	—	—	—	—
QoP-ML	√	√	√	√	√	√	√	√	√

to the overestimation of security measures, which causes an unreasonable increase in the system load [24, 25]. System's performance is especially important in environments with limited resources, including wireless sensor networks and mobile devices. Another example where such analysis should be performed is the cloud architecture. The latest research indicates three main barriers for using cloud computing: security, performance, and availability [26]. Unluckily, when the strongest security mechanisms are used, the system performance decreases, influencing its availability. This tendency is particularly noticeable in complex and distributed systems. The latest results show [24, 25] that in many cases the best way is to determine the required level of protection and then adjust security measures. (Among the means to meet these challenges one can indicate the security metrics [27].) Such approach is achieved by the means of the quality of protection models, where security measures are evaluated according to their influence on system's security.

Introduction of QoP term allows us to concentrate on security requirements in analysed system. In the literature, the security trade-offs are based on quality of protection (QoP) models. These models were created for different purposes and have miscellaneous features and limitations. In order to compare available security modelling approaches in terms of quality of protection, we prepared our set of qualities. Furthermore, we investigated different methodologies available in the literature and assessed them taking into account selected attributes. Proposed set of qualities (Table 1) along with their explanations and author's comments are presented below.

Approaches summarized in Table 1 can be characterized by the following main attributes.

(i) Quantitative assessment (QA) refers to the quantitative assessment of the estimated quality of protection of the system.

(ii) Executability (E) specifies the possibility of the implementation of an automated tool able to perform the QoP evaluation.

(iii) Consistency (Con) is the ability to model the system maintaining its states and communication steps consistency.

(iv) Performance evaluation (PE) gives the possibility of performance evaluation of the analysed system.

(v) Energy evaluation (EE) allows for the evaluation of the of energy efficiency of the analysed system.

(vi) Holistic approach gives the possibility of the evaluation of all security attributes.

(vii) Completeness is the possibility of the representation of all security mechanisms. This attribute is provided for all models.

(viii) Financial evaluation (FE) refers to the possibility of the financial assessment of the analysed system.

(ix) Ecological assessment (EA) answers questions about the environmental impact of the utilized mechanisms.

One can notice that only QoP-ML can be used for finding a trade-off between security (QA) and performance (PE) including energy efficiency evaluation (EE) of the system which is modeled in a formal way with communication steps consistency (Con). By means of QoP-ML, one can evaluate all security attributes (H) and abstract all security mechanisms which protect the system (C). Additionally, the QoP-ML approach is supported by the tool (E) required for the analysis of complex systems. It is also worth mentioning that only the QoP-ML allows for the financial assessment of the considered system (FE), giving the possibility of the evaluation of its environmental impact (EA).

According to the author's knowledge, Quality of Protection Modeling Language (QoP-ML), introduced in [28], is the only existing modeling language which satisfies all these requirements simultaneously. It allows for balancing security against the system efficiency, performing multilevel analysis, and extending the possibility of describing the state of the environment in detail. Quality of Protection Modeling Language permits determining the required quality of protection (QoP) and adjusting some of the security measures to these requirements, together with ensuring efficient system performance. This type of profound analysis can be accomplished by the help of the Automated Quality of Protection Analysis tool [4], which allows for the evaluation of the impact of every single operation defined in the prepared security model in terms of the overall system security. Additionally, in previous works, there were proposed and examined approaches which were successful also in assessing time, energy, and quality of protection of the analysed IT environments.

3. QoP-ML Overview

In the article [28] the Quality of Protection Modeling Language was introduced. Proposed solution provides the modeling language for making abstraction of cryptographic protocols that puts emphasis on the details concerning the quality of protection. The intended use of QoP-ML is to represent the series of steps, which are described as a cryptographic protocol. The QoP-ML introduced the multilevel protocol analysis that extends the possibility of defining the state of the cryptographic protocol. Since approaches presented in the literature usually speak for an example of a model-driven security, in the light of the available development methodologies, QoP-ML excellently fits in a design known as a Model-Driven Engineering. The Model-Driven Engineering (simply known as MDE) is meant to focus on the creation and utilization of the abstract representations of the knowledge that govern a particular domain, rather than on the computing, algorithmic, or implementation concepts. Model-Driven Engineering approach is a broader concept than Model-Driven Architecture (MDA) or Model-Driven Security (MDS). MDE adds multiple modeling dimensions and the notion of a software engineering process. The various dimensions and their intersections together with a domain-specific language (DSL) form a powerful framework capable of describing engineering and maintenance processes by defining the order in which models should be produced and how they are transformed into each other. Serving as a domain-specific language, QoP-ML is capable of expressing security models in a formalized, consistent, and logical manner.

As is apparent from the above description, QoP-ML is a flexible, powerful approach to model complex IT environments. Therefore, we utilized it to prepare our case study and evaluate the quality of chosen security mechanisms using its supporting, automatic framework. In the following sections we present all the significant components of the language we utilized to create model for our scenario.

3.1. General Information. The structures used in the QoP-ML represent high level of abstraction which allows concentrating on the quality of protection analysis. The QoP-ML consists of processes, functions, message channels, variables, and QoP metrics. Processes are global objects grouped into the main process, which represents the single computer (host). The process specifies behavior, functions represent a single operation or a group of operations, and channels outline the environment in which the process is executed. The QoP metrics define the influence of functions and channels on the quality of protection. In the article [28] the syntax, semantics, and algorithms of the QoP-ML are presented in detail.

3.2. Data Types. In the QoP-ML, an infinite set of variables is used for describing communication channels, processes, and functions. The variables are used to store information about the system or specific process. The QoP-ML is an abstract modeling language, so there are no special data types, sizes, or value ranges. The variables do not have to be declared before they are used. They are automatically declared when used for the first time. The scope of the variables declared inside the high hierarchy process (`host`) is global for all processes defined inside `host`.

3.3. Functions. The system behavior is changed by the functions, which modify the states of the variables and pass the objects by communication channels. During the function definition, one has to set the arguments of this function which describe two types of factors. The functional parameters, which are written in round brackets, are necessary for the execution of the function and the additional parameters which are written in square brackets and have an influence on the system's quality of protection. The names of the arguments are unrestricted.

3.4. Equation Rules. Equation rules play an important role in the quality of protection protocol analysis. Equation rules for a specific protocol consist of a set of equations asserting the equality of function calls. For instance, the decryption of the encrypted data with the same key is equal to the encrypted data.

3.5. Process Types. The processes are the main objects in the QoP-ML. The elements which describe the system behavior (functions, message passing) are grouped into processes. In the real system, the processes are executed and maintained by a single computer. In the QoP-ML the sets of processes are grouped into the higher hierarchy process named `host`. All of the variables used in the high hierarchy process (`host`) have a global scope for all processes which are grouped by the host. Normally, the variables used in the `host` process cannot be applied to the other high hierarchy process. This operation is possible only when the variable is sent by the communication channel.

3.6. Message Passing. The communication between processes is modeled by means of channels. Any type of data can be passed through the channels. The channels must be declared before the data is passed through. The data can be sent or received by the channels. The channels pass the message in the FIFO order. When the channels are declared with the nonzero buffer size, the communication is asynchronous. The buffer size equal to zero stands for the synchronous communication. In synchronous communication, the sender transmits the data through the synchronous channel only if the receiver listens to this channel. When the size of the buffer channel equals at least 1, then the message can be sent through this channel even if no one is listening to this channel. This message will be transmitted to the receiver when the listening process in this channel is executed.

3.7. Security Metrics. The system behavior, which is formally described by the cryptographic protocol, can be modeled by the proposed QoP-ML. One of the main aims of this language is to abstract the quality of protection of a particular version of the analysed cryptographic protocol. In the QoP-ML, the influence of the system protection is represented by the means of functions. During the function declaration, the quality of protection parameters is defined and details about this function are described. These factors do not influence the flow of the protocol, but they are crucial for the quality of protection analysis. During that analysis, the function's quality of protection (QoP) parameters are combined with the next structure of QoP-ML, the security metrics. In this structure, one can abstract the functions' time performance, their influence on the security attributes required for the cryptographic protocol, or other important factors during the QoP analysis.

4. Model-Based Decision Support System

In this section we introduce the foundations of our approach to the creation of a model-based decision support system. To perform a detailed, profound analysis of secure systems, focusing on every aspect of the examined environment, many different components need to be taken into consideration. To facilitate this complex process in interrelated IT systems, we designed, implemented, and utilized the model-based decision support system. Introduced DSS helps in the evaluation of consequences of given decisions and may advise what decision would be the best for achieving given set of goals. Being a well-organized, consistent solution, the proposed approach allows for a precise, specific analysis of the studied systems, making the analysis process multilevel (regarding time, energy, financial costs, environmental impact, and quality of protection). In following sections all stages of the proposed DSS are presented.

Step 1 (problem definition). The formulation of the problem may be defined as the process of acquiring and organizing knowledge in any situation on which the decision-maker intends to act. In this stage potential problems are identified and described. To protect the confidentiality, integrity, and/or availability of the considered system, security objectives need to be determined as early in this phase as possible. This can be done by following best practice recommendations on information security management defined in ISO/IEC 27002 standard [29]. Through the standardized approach to security objectives definition, we can assume that they reflect the high-level goals of the system in providing an appropriate secure environment for users of the system.

Step 2 (model construction). To organize and examine existing alternatives accurately, the modeling process should be appointed. The quality of the decision relies upon the correctness and accuracy of the modeling solution. Model creation aims at creating model of the chosen IT environment with the use of the Quality of Protection Modeling Language. QoP-ML, as a dedicated, specialized solution, uses qualitative models that help highlight important connections in real world systems and processes and recommend the solution to a given problem.

Security Metrics Definition. To be able to make a good decision, one needs to have relevant and accurate information, which help in choosing the best solution from existing alternatives. Right decisions impose a requirement for input parameters to be as solid and adequate as possible. When the model is ready, there comes the time when its input data should be gathered. Obtaining robust, repeatable security metrics, one should rely on the results gathered by Crypto Metrics Tool, since the framework was designed to use statistics to ensure most reliable measurements. Robust method, proposed by the authors and utilized further in our case study, involves characterizing a system based upon its statistical parameters. The approach examined and discussed in [30] guarantees that obtained results are accurate and free of random errors. The proposed tool yields the results in a form that is appropriate for the Automated Quality of Protection Analysis tool. Crypto Metrics Tool can be downloaded from the webpage of the QoP-ML project [4].

Scenarios Definition. Scenarios represent different versions of the evaluated model. They are also known as versions, in such one can assess the quality of protection of the modeled environment, using miscellaneous security mechanisms and applying them to the same model.

Data Flow Management. In any e-business solution, it is integral to manage every piece of data. To provide high quality services and drive successful business, company must have complete, accurate, combined data available in a timely manner. It is relevant to think about data flow as it pertains to every functional requirement, what kind of data with what kind of system within what time frame, in order to maintain efficiency and control throughout every process. To meet the given requirements, we proposed adding a new stage to the analysis process, the data flow management. Introduced phase uses a data flow management to help reduce the security risks, preventing security vulnerabilities at the same time. The main goal of this stage is to indicate most suitable mechanisms that optimize the data flow and make data available as quickly and consistently as possible.

Step 3 (identification and evaluation). This stage consists of multimeasure cross-validation and assessment of the modeled alternatives. Since QoP-ML provides the possibility of multilevel analysis, in this paper, we decided to extend this phase and divide it into five connected substeps. Time analysis, energy analysis, and QoP analysis [31] were proposed in previous works and implemented in Automated Quality of Protection Analysis tool. In this paper, we aim to introduce additional types of analysis: the finance estimation and carbon dioxide emissions evaluation.

Below we outline the underlying concepts of each of the proposed analysis phases to utilize them further in the paper. As shown in Figure 1, the time analysis is the analysis on which three remaining analyses are based. Therefore, before we go any further in introducing our new proposals, let us focus on time, energy, and quality of protection analyses.

The Time Analysis. The aim of the time analysis is to estimate the time, which was taken by all the security operations performed during the execution of the cryptographic protocol. The time analysis helps to determine mechanisms which are the most time efficient between the proposed security operations: it can be also useful when determining the total number of users that can be handled by the server within a given time interval.

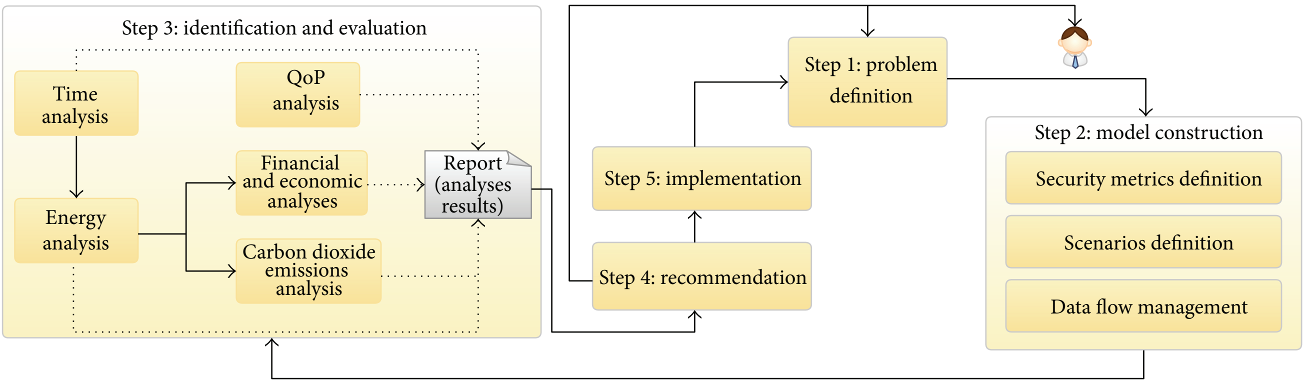

Step 3: identification and evaluation
Time analysis
QoP analysis
Energy analysis
Financial and economic analyses
Carbon dioxide emissions analysis
Report (analyses results)
Step 1: problem definition
Step 5: implementation
Step 4: recommendation
Step 2: model construction
Security metrics definition
Scenarios definition
Data flow management

Figure 1: Utilized decision support system.

The Energy Usage Analysis. Besides the time analysis, we proposed including the power consumption to the analysis process, in order to evaluate the energy consumption of the modeled system. To obtain the total amount of the energy consumed by the security operations, the time analysis module must be included in the performance analysis process, since it tracks the time of operations and communication steps. The energy consumption is calculated as the sum of energy consumed by operations that use CPU. (The details can be found in [32].)

The QoP Analysis. Another crucial aspect of the multilevel analysis is the assessment of the security quality. Such analysis is based on the evaluation of the impact of security mechanisms on system performance. Estimation of quality of protection (QoP) is a challenging task: utilized approach should be flexible enough to allow evaluating quality of protection of different versions of cryptographic protocols (security policies) in an economic manner. In [31] the authors introduced the framework which permits assessing the quality of protection of previously predefined security mechanisms and for not directly defined security configurations. Proposed solution, being an automated approach, lets one define all possible scenarios for all IT processes, which can be very complex and in many cases not feasible with other existing frameworks.

New Type of Analyses: The Financial and Economic Analyses. The reason for the introduction of the new stage to the utilized analysis scheme is the crucial role of finances in IT. As data center is one of the most financially concentrated assets of any organization, there is a high demand for a standardized method for measuring the total cash outlay of the physical infrastructure of data centers. Designing IT budgets, identifying IT budgets items, developing appropriate pricing strategies, and implementing and operating financial management enable companies to get more purchasing power out of their budgets and preserve cash for operational issues. One of the essential parts of the enterprise economic policy is the effective management of the utilized power, which is as well covered in our pleasantly new analysis scheme. In large IT environments, consisting of tens, if not hundreds, of thousands of working machines, electricity is the factor which generates one of the biggest expenditures.

New Type of Analyses: The Carbon Dioxide Emissions Analysis. The recent growth of data centers requires more energy-efficient server design. However, these days saving some money on energy bills is one thing, but reducing the CO_2 is a much more admirable goal. Since data centers are a large fraction of the IT, there is a high demand for lowering emissions of CO_2 (and, in turn, bills) by reducing power consumption. Besides servers, data centers also use storage and network components, which as well produce huge amounts of carbon dioxide. Estimating the amount of the CO_2 emissions, it is important to note that its total quantity depends on miscellaneous factors, such as the size of the data center (number of all its working components), server load (which translates into the utilized kilowatt-hours), and the type of resource utilized to generate electricity. Hence, machines, which consume a great deal of power, cause a negative environmental impact. As stated before, reduction of the energy usage from green computing techniques translates into lower emissions of carbon dioxide, stemming from a reduction in the resources used in power plants.

Steps 4 and 5 (recommendation and implementation). After the choice phase all alternatives are searched and evaluated and one of them is chosen as recommended solution. The chosen decision is to be carried out. Only if the recommended solution is successfully implemented, the problem is considered solved.

5. The Financial, Economic, and Carbon Dioxide Emissions Analyses

In this section we present and briefly describe our two new modules utilized in the proposed decision support system. Introduced equations are used to calculate the financial aspect of the data center maintenance. Further in this section we discuss the environmental impact of the data center management. By means of the proposed method and corresponding formula, we present a simple approach for estimating the rough amount of the carbon dioxide released to the atmosphere.

5.1. The Financial Analysis.

Calculating total operating cost of a data center, one needs to take into account both *fixed* and *variable* costs, which are affected by complex and interrelated factors. In this paper, performing the finance analysis, we took into consideration only costs of power delivery and cooling infrastructure utilization, as they refer to the *variable* expenditures. Both power delivery and cooling costs respond directly and proportionately to changes in input parameters (such as utilized security mechanisms, scenarios, and metrics). Any change in any of the input parameters can have the influence on those expenditures. When it comes to the *fixed* costs, (such as hardware amortization expenditures, hardware and software purchase costs, and personnel salaries), we simply omitted them in our analysis, since they are independent, remain more or less unchanged irrespective of the input parameters, remain constant throughout the relevant range, and are usually considered sunk for the relevant range (not relevant to output decisions).

Below we introduce general formulas used for calculating total energy and cooling costs.

5.1.1. Cost of Power Delivery.

The design of the electrical power system must ensure that adequate, high quality power is provided to each machine at all times. One should pay special attention to the relationship between the CPU utilization and energy consumed by other components of a working server. Since the amount of the power consumed by a single machine consists of the energy usage of all its elements, it is obvious that a decrease in the energy consumed by the CPU will result in a lower energy consumption in total. Thus, total cost of both power delivery and utilization can be summarized as follows:

$$\varsigma_{\text{power}} = \kappa_{\text{busy+idle}} \cdot \sigma \cdot \chi \cdot \rho, \tag{1}$$

where $\kappa_{\text{busy+idle}}$ is the total amount of the utilized kilowatt-hours by the server, σ is the cost of a one kWh, χ is the total amount of hours when the server was busy, and ρ is the total amount of days when the server was busy.

However, total power consumption of a single server is a sum of the power utilized by all its working components. In such case, $\kappa_{\text{busy+idle}} = \kappa_{\text{CPU}} + \kappa_{\text{RAM}} + \kappa_{\text{HDD}} + \cdots$. Therefore, to assess the real cost of a single working machine, the approximate amount of the energy utilized by all its elements should be estimated.

In general, electricity provided by the electric company and consumed by its customers is measured in kilowatt-hours. Being aware of the price of one kWh and knowing that CPU worked χ hours through ρ days, utilizing κ kilowatt-hours, it is fairly straightforward to calculate the total financial cost of its work, using the formula analogous to (1). Before we start further evaluation of the energy consumed by the CPU, we need to make some assumptions about its utilization. Let us introduce the simplified CPU utilization formula:

$$U = \frac{R}{C}, \tag{2}$$

where U is the CPU utilization, expressed in percentage, R defines our requirements, the actual busy time of the CPU (seconds), and C stands for the CPU capacity, the total time spent on analysis (seconds).

Usually, the CPU utilization is measured in percentage. In the formula introduced above, R refers to the time we require from the CPU to perform an action. This time is also known as the *busy* time. CPU capacity can be expressed as the sum of the *busy* and *idle* time (i.e., the *total* time available for the CPU). Going simple, one can say that over a 1-minute interval, the CPU can provide a maximum of 60 of its seconds (power). The CPU *capacity* can then be understood as *busy time* + *idle time* (the time which was used plus the one which was left over). Using the above simplifications, when going multicore, CPU capacity should be multiplied by the number of the CPU cores ($C = C{\cdot}\text{cores}$). In context of served requests, presented equation (2) can be further detailed as follows:

$$\text{load}\ [\%] = \frac{\text{time}_{\text{session}} \cdot \text{users}}{\text{time}_{\text{total}}}, \tag{3}$$

where $\text{time}_{\text{session}}$ refers to the time the single request took (seconds), users stands for the number of incoming user connections to be managed, and $\text{time}_{\text{total}}$ is expressed as $\text{time}_{\text{session}} \cdot \text{users} + \text{time}_{\text{idle}}$ and represents the total time taken by all the handled connections together with the one which was left over.

5.1.2. Cost of Cooling Infrastructure Utilization.

Providing sufficient cooling is essential to ensure reliable running of servers, routers, switches, and other key data center pieces of equipment. As the cooling infrastructure absorbs energy to fulfill its function, the cost of cooling needs to be included in the total cost of the server maintenance. To keep servers operational, cooling a server consumes defined amount of watts for every watt that powers it, depending on cooling efficiency. Same as in the case of power delivery, to keep server rooms temperature within the listed tolerances, there is a requirement for additional, back-up cooling solutions. Back-up chillers generate additional costs, which must be taken into account as well. To obtain an approximate amount of the power consumed by the cooling, one can use the equipment heat dissipation specifications, most often expressed in British Thermal Units (BTUs), generally available either in the system users guide or on the manufacturers website. Specifications state how many BTUs are generated in each hour by the individual machine. Therefore, the formula for calculating the cooling cost to keep the equipment in normal operating conditions is given as follows:

$$\varsigma_{\text{cooling}} = \text{BTU}_{\text{cooling}} \cdot \sigma \cdot \chi \cdot \rho, \tag{4}$$

where $\text{BTU}_{\text{cooling}}$ is the amount of the BTUs generated by the cooling system (per server), σ is the cost of a one kWh, χ is the total amount of hours when the server was busy, and ρ is the total amount of days when the server was busy.

Knowing that one watt is equal to 3.412 BTU/hour (e.g., using 100 watts of power generates 341.2 BTU per hour), the above formula becomes analogous to (1).

5.1.3. Total Cost. Key elements in data center budgets are the power delivery system, the networking equipment, and the cooling infrastructure. Besides the above most-crucial factors, there exist additional costs associated with data center operation, such as personnel and software expenses. Therefore, the real operating cost of the data center can be expressed as

$$\varsigma_{\text{total}} = \varsigma_{\text{power}} + \varsigma_{\text{cooling}}. \tag{5}$$

As mentioned before, in our case we simply exclude *fixed* costs from our analysis, since they do not vary with changes in input parameters.

5.2. The Carbon Dioxide Emissions Analysis. Using simple formula one can easily estimate the annual environmental impact (ς_{CO_2}) for the considered CPU (and, analogously, for the single server, server room, and, of course, the whole data center):

$$\varsigma_{\text{CO}_2} = \kappa \cdot \chi \cdot \rho \cdot \delta, \tag{6}$$

where κ is the total amount of the utilized kilowatt-hours, χ is the total amount of hours when the server was busy, ρ is the total amount of days when the server was busy, and δ defines the amount of pounds of CO_2 per kWh.

The amount of carbon dioxide (CO_2) produced per kilowatt-hour (kWh) depends on the type of the fuel utilized to generate electricity. One can calculate the amount of carbon dioxide produced per kilowatt-hour for specific fuels and specific types of generators by multiplying the CO_2 emissions factor for the fuel (in pounds of CO_2 per million BTU) by the heat rate of a generator (in BTU per kWh generated) and dividing the result by 1 000 000.

6. Case Study: Security-Based Data Flow Management in Data Center

In the paper we utilized the case study approach to obtain profound, in-depth understanding of the introduced, complex, multilevel analysis method in its potential implementation context, which is, in our case, the security-based data flow management, described in detail in latter sections. To demonstrate the use of the presented system and its analysis scheme, we proposed using the role-based access control approach, prepared a scenario, and assessed its quality. We made use of QoP-ML framework and created by its means the role-based access control model (since it is an excellent example of the data flow management), to examine the quality of chosen security mechanisms in terms of time, energy, quality of protection, finance, and environmental impact. Below we use the model-based decision support system proposed in Figure 1 and briefly describe its steps in following subsections.

6.1. Example DSS Implementation

Step 1 (problem definition). The decision-making process begins when one identifies the real problem. In our case, we managed to formulate questions about the fastest, the most energy-saving, the most secure, the cheapest, or the most green security mechanism among available solutions, on the example of the role-based access control model.

Addressing the security objectives issue, we provided a clear definition of what the system should do to counter and manage threats to its security. Our goal was to examine how to achieve a balance between the performance and security. We managed to perform such analysis by striving to accomplish example security objectives, created on the basis of guidelines from ISO/IEC 27002:2013, using miscellaneous security mechanisms. Table 2 contains established security objectives together with their example realization in our model.

Step 2 (model construction). Before we adopt QoP-ML approach to our needs, utilize analysis scheme, and finally perform the actual estimation of the daily, weekly, monthly, and annual server impact in terms of money and environmental effects, let us give some assumptions about the potential implementation environment (Figure 2). Consider a call center company located in Nevada, USA, managing a typical IT environment of 42U server racks (520 physical servers in total, 13 physical servers per rack). Suppose that the example enterprise uses electricity provided by the electric company, which measures energy consumption in kilowatt-hours. Examined enterprises have many departments with miscellaneous responsibilities, and thus distinct permissions and rights to the company's assets. Given a specified load capacity, servers handle enterprise's traffic continuously for 24 hours. In the example enterprise, we have 1 000 CSRs' workstations (which automatically translates into the number of the users assigned the first role), 10 security managers (employees having role's 3 permissions), and 20 system operators, people with a second level of authorization. Each server in the example call center is equipped with the Intel Xeon X5675 processor, being able to handle the required number of employees' connections, regardless of the assigned RBAC role. To simplify, in our analysis, we assumed that on average single Dell PowerEdge R710 server consumes about 0.3 kWs per hour on performing its daily routine tasks and added the amount of power consumed by the CPU in each role.

After a brief introduction of the potential implementation environment, let us now move on to the example utilization of our novel analysis scheme, where we, step by step, examine the system in detail.

We proposed preparing the role-based access control in the Quality of Protection Modeling Language to improve security management, thereby enhancing the security itself, and analyse security-based data flow management impact on the whole system performance in the context of time and energy consumption, quality of protection, cost savings, and environmental impact.

Table 2: Security objectives established according to the guidelines provided by ISO/IEC 27002:2013.

Security objective	ISO/IEC 27002:2013 section	Realization
(1) Security management and access control (SO1)	5.1.1b: assignment of general and specific responsibilities for information security management to defined roles 6.1.1c: authorization levels should be defined and documented 9.1.2c: authorization procedures for determining who is allowed to access which networks and networked services 9.1.2d: the means used to access networks and network services [should be defined] 9.1.2e: user authentication requirements for accessing various network services 9.2.3a: the privileged access rights associated with each system or process and the users to whom they need to be allocated should be identified	Data flow management: RBAC role assignment
(2) Confidentiality, authenticity, or integrity protection (SO2)	10.1.1a: the management approach towards the use of cryptographic controls across the organization 10.1.1b: the required level of protection should be identified taking into account the type, strength, and quality of the encryption algorithm required 10.1.1c: the use of encryption for protection of information transported across communication lines	Communication channel protection (TLS)
(3) System capacity management (SO3)	12.1.3: the use of resources should be monitored and tuned and projections made of future capacity requirements to ensure the required system performance	Monitoring and analysis of server resources with distinct security mechanisms applied: the multilevel analysis process
(4) Assurance of correct and secure operation of information processing and handling facilities (SO4)	12.1.1b: procedures should specify the operational instructions for processing and handling of information both automated and manual	Secure access to FTP, web, . . ., and servers
(5) Maintenance of the integrity and availability of information (SO5)	12.3.1a: accurate and complete records of the back-up copies and documented restoration procedures should be produced 12.3.1c: the back-ups should be stored in a remote location, at a sufficient distance to escape any damage from a disaster at the main site	Information back-up: secure access to data storage
(6) Protection of information involved in electronic messaging (SO6)	13.2.3a: protecting messages from unauthorized access, modification, or denial of service 13.2.3f: stronger levels of authentication controlling access from publicly accessible networks	Secure access to e-mail server

In complex secure environments, aside from enhanced security and reduced administration costs, system performance is another important artifact that needs to be carefully evaluated. Modeling a complicated enterprise infrastructure and applying the role-based access control is a challenging task. Instead of introducing a complex infrastructure abstraction and considering all the possible operations, we managed to model a simple RBAC usage in an example business situation (i.e., we examined communication steps that occur in a single client accessing a single server) and evaluated its performance with the use of the Automated Quality of Protection Analysis tool. Due to the page limitation, complete RBAC model cannot be presented in the paper. However, it can be downloaded (along with the Automated Quality of Protection Analysis tool) from the web page of the QoP-ML project [4].

Modeling the RBAC approach, we defined QoP-MLs functions, equations, channels, processes, subprocesses, and hosts. Let us now briefly discuss utilized QoP-ML structures we prepared to create the role-based access control model. To create the role-based access control in the Quality of Protection Modeling Language, we prepared a security model consisting of two communicating hosts: a client and a server. In addition, we prepared three asynchronous communication channels to facilitate the information exchange process. On the client's site, we modeled the main process being responsible for establishing secure connection with the server and a subprocess capable of generating different types of network traffic based on the role received from the server. Server abstracted in Quality of Protection Modeling Language is much alike the client; it also has a main process which sets up the communication parameters, but, opposite to the client, it contains three subprocesses, thereby being able to manage clients with miscellaneous levels of authorization. Modeling the RBAC, we defined QoP-MLs functions, equations, channels, processes, and hosts. In this section we present and discuss only the essential QoP-ML elements we prepared to create the security model.

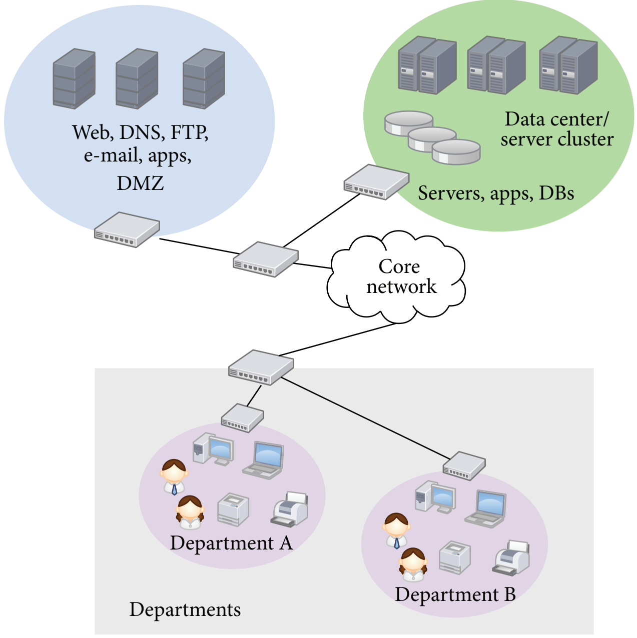


FIGURE 2: Example enterprise network architecture.

Functions defined in Quality of Protection Modeling Language refer to the roles specified for the example enterprise. Declared operations represent 3 roles: *role 1*, *role 2*, and *role 3*. Along with *functions* we declared some *equational rules*.

Since one needs communication between running processes, it is necessary to define QoP-MLs channels. Defined channels are used to exchange the TLS traffic, actual data traffic, and the assigned RBAC role.

In our approach, subprocesses express operations that may be performed by users with different RBAC roles assigned. Client and server processes are used to model the TLS handshake operations. After defining processes and subprocesses, one can group them into `host` structures, named *client* and *server*, which express the communicating sites in the RBAC model. An example client's subprocess connects to the FTP server, while server's subprocess handles the request, according to the assigned role. Notice that the server handles distinct roles differently.

Security Metrics Definition. It is worth noticing that QoP-ML provides the ability to determine modeled system performance on machines with different hardware specifications. To examine the hardware impact on access control management, one can use QoP-ML's feature, the security metrics, described earlier in this paper. We collected data required by the *Security Metrics Definition* stage using the Crypto Metrics Tool and performing the metrics generation process on a server equipped with the Intel Xeon X5675 processor. Using CMTool we gathered security metrics for all the security mechanisms proposed in our case study (Box 1). We took into consideration just one of the available base measures: the CPU time. In our case, the CPU time indicates the amount of time for which a central processing unit was used for processing, the execution of the security operations (such as AES encryption/decryption, SHA-1 hashing, and others available). Obtained, free of random errors values (along with their attributes) are gathered in Table 3.

TLS version 1
(security level = *low*)
RC4 + MD5
TLS version 2
(security level = *medium*)
AES/CBC 128 + SHA-1
TLS version 3
(security level = *high*)
AES/CBC 256 + SHA-512

BOX 1: TLS protocol versions together with corresponding cryptographic algorithms.

Remaining characteristics needed for the analysis (such as the CPU voltage) were taken from the official documentation of the processor which was utilized for the metrics generation [33]. Since cryptographic operations utilized in the defined scenario are considered power-consuming, in our analysis, when evaluating energy utilized in the *busy* state of CPU, we decided to choose its maximum available voltage. Assessing the power consumed in the *idle* state of the CPU, one should assume the smallest available voltage value. All the utilized values are gathered in Table 4.

Scenarios Definition. To emphasize and prove role's influence on data flow management, system performance, and adequate security objectives realization, we prepared and analysed a simple scenario. Prepared scenario refers to the example business situation and possible role assignment in the actual enterprise environment. Given the enterprise network infrastructure in Figure 2, consider having three roles: *role 1*, *role 2*, and *role 3* with corresponding security levels: *low*, *medium*, and *high*. The role's permissions are defined as follows.

(i) *Role 1*. Users assigned to this role have the access to the e-mail, FTP, web, and application servers with the communication channel protected by means of the TLS protocol in version 1; moreover, they are allowed to access data centers in the enterprise's DMZ.

(ii) *Role 2*. This represents users with greater responsibilities and thus stronger protection mechanisms of the actions they are allowed to perform. Being a member of the role 2, user is able to access e-mail, FTP, web, and application servers with a communication channel secured by the TLS version 2. Furthermore, members of the *role 2* are permitted to access servers *S*1 and *S*2 in the production DMZ.

(iii) *Role 3*. Members of the role 3 are the most authorized users among considered roles. They are allowed to access e-mail, FTP, web, and application servers with the communication channel protected by means of the TLS protocol in version 3, permitted to perform actions on production management resources and on general office assets, and process data available within

Table 3: Security metrics gathered by Crypto Metrics Tool for the purpose of our case study.

Security mechanism	Characteristics			
	Operation type	Operation mode	Key length	Base measure (CPU time)
AES	Encryption Decryption	CBC	128 bytes	0.0000043972 ms
AES	Encryption Decryption	CBC	256 bytes	0.0000060001 ms
RSA	Encryption Decryption	—	2048 bytes	0.0002539800 ms
RC4	Encryption Decryption	Stream	128 bytes	0.0000012571 ms
SHA-1	hmac	—	—	0.0000016390 ms
SHA-512	hmac	—	—	0.0000028952 ms
MD5	hmac	—	—	0.0000014137 ms

Table 4: CPU specific values used for our estimation.

Power consumption	$\text{Voltage}_{\min}$	$\text{Voltage}_{\max}$	$\text{Current}_{\text{busy}}$	$\text{Current}_{\text{idle}}$
95 W	0.75 V	1.35 V	W/V ≈ 70.37 A	W/V ≈ 126.67 A

DMZ data centers with the strongest communication protection mechanism.

Using described roles we managed to build different versions which comprise example implementation scenario, summarized in Table 5.

As it is clear from Table 5 and Box 1, different roles use distinct security mechanisms. *Role 1* makes use of two cryptographic algorithms, namely, RC4 and MD5. RC4 is the most widely used software stream cipher. The cipher is fairly simplistic when compared to competing algorithms of the same strength and boasts one of the fastest speeds of the same family of algorithms. It should be chosen when the performance is the main concern.

In TLSv2 we utilized AES in CBC mode with 128-bit key and SHA-1. AES is a symmetric-key algorithm which makes use of the same key for both decrypting and encrypting information. AES is not great in that its strength comes through excessive CPU effort but it is of considerably greater strength (both theoretically in real world attacks) than RC4.

Last but not least, the *most secure* role, *role 3*, utilizes for its functioning AES in CBC mode with 256-bit key and SHA-512 as a cryptographic hash function. If the performance is the main concern, one should choose AES-128, since AES-256 is not much more secure in practice (128-bit keys cannot be brute forced anywhere in the foreseeable future).

Data Flow Management. To examine the performance of miscellaneous roles in a real-life situation, we mapped the general rules proposed in the defined scenarios to the example segment of the enterprise network in a call center company. Considering the extended example, we can actually prove that the chosen role matters if it comes to the system performance and the energy usage. In our case study, users assigned to the given role have different responsibilities they perform. The role of the user is usually determined by her/his responsibilities within the company. Such extension clarifies our approach: the need for the RBAC and different security levels is undeniable. All network traffic in the enterprise needs to be secure, but users who manage the most valuable enterprise assets should use the most secure network connections, since any security vulnerability or weakness can compromise the integrity, availability, or confidentiality of the enterprise crucial resources and thus expose the enterprise to serious costs.

In that case, we mapped the third role to the category of users called *security manager*, since the mentioned role has the highest privilege in the system managing crucial assets and thus needs the highest possible security level to ensure required security. Users assigned the mentioned role have access rights that enable them to perform management level operations. Their permissions do not allow them to perform application related activities such as usage of the Customer Service's VoIP software. They perform the following, example operations: User Profile Maintenance, Task Access Control, System Security, and, the most important from the company's point of view, Communication System Maintenance (VoIP Software Maintenance). Call centers function efficiently when they have strong communications system consisting of voice over IP technology or VoIP. With a reliable VoIP software technology, a call center would be able to render outstanding communication service with the least risk of downtime occurrence.

The second role introduced in our scenario is equivalent to the permissions of the *system operator* in the example enterprise network. The activities of the users assigned the *system operator* role are as follows (but not limited to): File Transfer (FTP), Archival/Retrieval, and the rest. However, *system operators* are not allowed to perform any application or security manager related activities. Since *system operators* have access to the less significant enterprise assets, the security level of the role can be, respectively, lower.

Table 5: Scenario defined for our case study.

	Scenario		
	Role 1	*Role 2*	*Role 3*
Access type	E-mail, FTP, web, applications, data center servers	E-mail, FTP, web, applications, server S1 in DMZ, server S2 in DMZ	E-mail, FTP, web, applications, data center servers, DMZ production servers, General office
Data size (for each action separately)	E-mail: 8 MB FTP: 10 MB Web: 3 MB Applications: 4 MB Data center: 25 MB	E-mail: 10 MB FTP: 10 MB Web: 4 MB Applications: 6 MB Server S1: 5 MB Server S2: 15 MB	E-mail: 10 MB FTP: 5 MB Web: 4 MB Applications: 6 MB Data center: 15 MB DMZ production servers: 8 MB General office: 2 MB
Security mechanisms	TLSv1 (see Box 1)	TLSv2 (see Box 1)	TLSv3 (see Box 1)
Security level	*Low*	*Medium*	*High*
Security objectives	1, 2, 3, 4, 5, 6	1, 2, 3, 4, 5, 6	1, 2, 3, 4, 5, 6

The role which uses the *role 1* security parameters is assigned to the ordinary users of the system, namely, employees, the Customer Service Reps (CSRs), who have access to the enterprise applications and server's resources. They cannot perform any security manager or system operator activities. Their role within the system includes the usage of the enterprise-specific applications (the VoIP software, self-services software, speech technology applications, and many more) provided by the enterprise servers (being under the supervision of the security managers).

Users assigned the defined roles can use the same types of enterprise resources (FTP, WWW, applications, databases, and others too numerous to mention). However, in fact, they work with different physical assets. As mentioned above, single session performed by the user consists of operations appropriate for the user's role.

Step 3 (identification and evaluation). During our research, we were able to perform the actual analyses for only one client accessing the server in a single session. Remaining results were evaluated to grow linearly, along with the number of incoming session requests. We examined time spent on handling client connections, accompanying amount of the consumed kilowatt-hours, and the resulting financial costs along with emissions of carbon dioxide.

To satisfy the *confidentiality, authenticity, or integrity protection* security objective we assumed that all the utilized applications are tunnelled by miscellaneous TLS protocol versions. Proposed variants of the TLS protocol together with equivalent cryptographic algorithms are summarized in Box 1.

In following subsections we present and discuss obtained results and consider remaining security objectives. In time, energy, and QoP analyses, we estimated the results that can be applied to the working central processing unit. Broader picture is presented in financial, economic, and environmental analyses, where we took into consideration not only the CPU but the server itself.

The Time Analysis. Consider, for instance, *role 1*, in the first scenario. Here, the user has access to the e-mail, FTP, WWW, and application servers and to the data center resources. A single session between the client and the server can carry the traffic with the maximum size of 50 MB. The communication channel is protected by the TLS protocol in version 1 and it takes exactly 0.28 seconds to perform (Table 6). Nevertheless, having identical conditions, changing only the channel protection type, time extends to 1.2025 seconds (for *role 2*). As *role 3* is the most secure communication type example (having other conditions equal to *role 1* and *role 2* at the same time), it is reasonable to presume that it takes the longest time to accomplish (1.2683 seconds). Such analysis provides serious argumentation to believe that the assigned role (thus the level of protection) can influence data flow and thus overall system performance.

Supposing that the time of the chosen security operation assessed by the time analysis module was equal to τ seconds and assuming that there was a given time interval equal to $\phi = 3\,600$ seconds, for the server working under φ% of machine load, it is quite straightforward to calculate the maximum number of users the server is able to manage within the given time interval:

$$\text{users}_{\max} = \frac{\varphi \cdot \phi}{100 \cdot \tau}, \tag{7}$$

where τ is the total time of a single session in seconds, φ is the machine load in percentage, and ϕ is the considered time interval in seconds.

However, to prove our hypothesis, we estimated the hourly server load of the server being accessed by users with distinct roles. Utilizing results obtained by AQoPA along with

Table 6: Server's performance results obtained by AQoPA suggest that the assigned role (and thus, the proper data flow management) matters if it comes to the system's performance.

Action performed (access)	Scenario		
		RBAC role	
	Role 1	*Role 2*	*Role 3*
E-mail	0.0448 s	0.1266 s	0.1865 s
FTP	0.0560 s	0.1266 s	0.0932 s
Web (WWW)	0.0168 s	0.0506 s	0.0746 s
Application(s)	0.0224 s	0.0760 s	0.1119 s
Data center	0.1400 s	0.0000 s	0.2798 s
Server S1 in DMZ	0.0000 s	0.0633 s	0.0000 s
Server S2 in DMZ	0.0000 s	0.1898 s	0.0000 s
DMZ production servers	0.0000 s	0.0000 s	0.1492 s
General office assets	0.0000 s	0.0000 s	0.0373 s
Total time (full session)	0.28 s	0.6329 s	0.9325 s

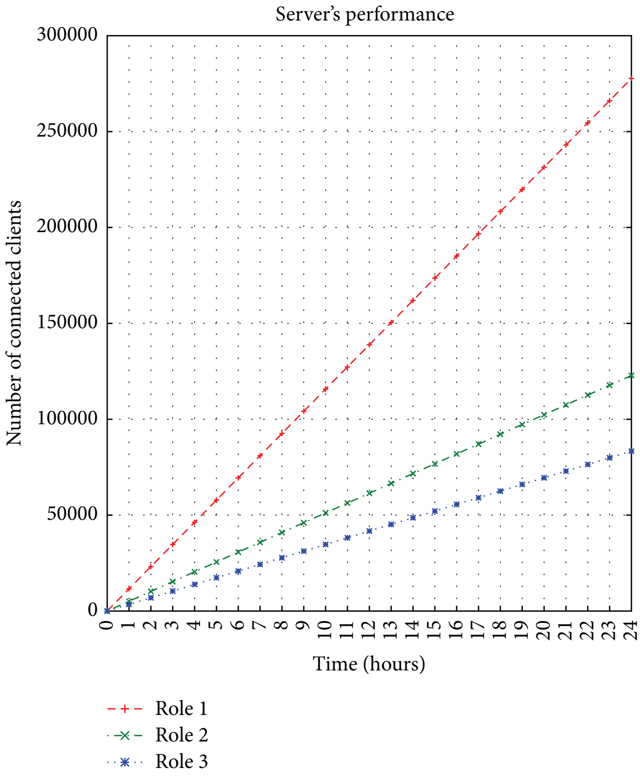


Figure 3: Server's performance in the considered *scenario*.

those that have been estimated, we evaluated the maximum number of clients (sessions) with different authorization permissions the server is able to handle within an hour having 90% of CPU load. Our assessment is quite straightforward; knowing that, for the existing server, it takes 0.28 seconds to handle user assigned *role 1* and using the simplified formula for CPU utilization, one can easily calculate the number of served clients (considering given conditions and using (7)):

$$\text{users} = \frac{90 \cdot 3\,600}{100 \cdot 0.28}, \tag{8}$$

which results in about 11 571 handled employees per hour (Figure 3). Assuming the linear growth, one can estimate that, during two hours, the machine is able to manage about 11 571 · 2 = 23 142 connections, while through three hours this value increases to 11 571 · 3 = 34 713 handled users (and so on). The same type of assessment was performed for the remaining roles (*role 2* and *role 3*).

The performed time analysis confirmed that the server is capable of dealing with the maximum of 3 474 users within an hour having 90% of CPU load (handling employees assigned *role 3* permissions). Assuming the linear growth of the number of served users, one can say that within 24 hours single server is able to manage 24 · 3 474 = 83 376 employees having *role 3* privileges. When the company decides to increase the number of handled *most secure* users' connections by, for instance, two times, and keep the CPU load at the same level simultaneously, there is a need for another server to handle them all. Managing 83 376 · 2 = 166 752 employees assigned *role 3* permissions greatly exceeds server capabilities. To serve 166 752 clients, single server would have to work for two days, not as we want it to be, for 24 hours. However, when we change the protection type to TLSv1 leaving the CPU load on the same level, it then becomes possible to handle all connections without buying new equipment, since the maximum number of users having *role 1* privileges (277 704), who can be dealt with in 24 hours, is greater than the number of employees assigned *role 3* permissions served by server during two days. Using simple math we can say that during 24 hours server which deals with *role 1* permissions set is able to manage roughly 67% more connections than the one which works for 48 hours with users having *role 3* privileges (or, in other words, server handling users of the third role is capable of realizing about 40% less connections).

Analysing obtained results and assuming 90% of machine load, one can easily notice that the server is able to handle clients with the first authorization level faster than the same number of users permitted to perform *role's 3* actions. Gathered results clearly indicate the relationship between the assigned role and consumption of server resources: the longer time the action needs to accomplish, the more the server resources are going to be used. The server, which works longer, utilizes more resources, thereby consuming a greater amount of energy.

The Energy Usage Analysis. Besides the presented time analysis, we modeled the energy usage for our scenario. We collected the data required by the AQoPA's Energy Analysis Module from the official documentation of the CPU which was used for the metrics generation [33]. In our analysis, we considered only the *busy* time of the CPU. We simply ignored the left-over time and focused only on power consumed while performing security operations. Table 7 contains the results, expressed in joules, gathered by the AQoPA tool.

The QoP Analysis. In [31] different versions of the TLS protocol were analysed: their security *quality* was assessed in terms of *confidentiality*, *integrity*, *availability*, and *authorization*.

Table 7: Server's energy analysis results obtained by the AQoPA's energy analysis module (joules).

Action performed (access)	Scenario		
	RBAC role		
	Role 1	*Role 2*	*Role 3*
E-mail	2.1284 J	6.0129 J	8.8610 J
FTP	2.6605 J	6.0129 J	4.4305 J
Web (WWW)	0.7981 J	2.4051 J	3.5444 J
Application(s)	1.0642 J	3.6077 J	5.3166 J
Data center	6.6512 J	0.0000 J	13.2915 J
Server S1 in DMZ	0.0000 J	3.0064 J	0.0000 J
Server S2 in DMZ	0.0000 J	9.0194 J	0.0000 J
DMZ production servers	0.0000 J	0.0000 J	7.0888 J
General office assets	0.0000 J	0.0000 J	1.7722 J
Total power usage (full session)	13.3024 J	30.0644 J	44.305 J

Table 8: The qualitative interpretation of QoP evaluation of the analysed versions of TLS protocol.

Security mechanisms	Security attribute		
	Confidentiality	Integrity	Availability
RC4 + MD5	Very low	Very low	Very high
AES/CBC 128 + SHA-1	High	Medium	Medium
AES/CBC 256 + SHA-512	Very high	Very high	Very low

Authors presented the quality of protection evaluation of the chosen security mechanisms and defined logical, consistent steps that help assess the quality of security operations. Since in their work researchers evaluated two of our four TLS versions, we summarized their results in Table 8, adding the qualitative interpretation of QoP evaluation, computed for the remaining cases with the help of the SME tool. The complete evaluation of the chosen TLS versions was presented in [31]. The following examination is based on the analysis performed in [31].

When the analysis is finished, one can finally interpret the results. As is apparent from Table 8, TLS version 1 (RC4 + MD5) stands for the most CPU-efficient solution. The crucial part of the TLS protocol, expressed by the means of the *authorization* attribute, indicates that AES/CBC 256 + SHA-512 and AES/CBC 256 + SHA-1 are the best options when it comes to the server authorization. Comparing the results obtained with the SME tool, one can choose those security mechanisms, which satisfy given requirements best in terms of *confidentiality*, *integrity*, and *availability*.

New Type of Analysis: The Financial Analysis. In this section, we present a brief overview of predicting the total budget required to manage a data center, mainly focusing on a method for measuring its total cost and indicating possible gains. In addition, we try to confirm our thesis about the influence of the security management and data flow on the total cost of the data center maintenance by the analysis of the defined scenario, together with the estimation of its accompanying costs. Performing our estimations, we utilized formulas presented in Section 5.

Determining the total cost of the working machine, the biggest attention should be drawn to the CPU, since this component is the largest energy consumer in a typical computer. Using (1), one is able to estimate the daily, weekly, monthly, and annual cost of power consumption for each CPU in the enterprise. Although the CPU is clearly the largest consumer of energy, total energy utilization of a single server consists of power usage of all its components. To obtain total power consumption of a single server, one needs to take into consideration the power usage of its HDDs, RAMs, network cards, video cards, and so forth. Depending on the server's workload, the energy utilized for the cooling system (the amount of the power needed to ensure a proper temperature for the busy machine and its components) must be covered as well. Let us have a closer look at, for instance, CPU's fan power consumption. One should keep in mind that *idle* server requires less energy and thus cooling than the *busy* one. In some certain scenarios CPU fans contribute much to the power requirements of PC components. Among factors that affect fan energy usage one can enumerate the fan speed (measured in RPM), fan size (80 mm, 92 mm, 120 mm, 140 mm, and 200 mm), and the fact whether it has LED lights. Typical power consumption of a 80 mm case fan (3,000 RPM) ranks between 2.4 and 3 W [34].

As it can be seen from the time analysis, to handle the exact number of incoming connections, the server needs to spend more time on serving users assigned *role 3* privileges, than those having *role 1* permissions. Based on Figure 3, to deal with about 20 000 users, the machine works through approximately 1.7 hours (*role 1*), but, considering *role 3* accesses, this value increases to about 5.8 hours. Suppose that our single server uses CPU together with 3 W fan (both working for twenty-four hours a year). During 5.8 *busy* hours, fan utilizes about $(5.8 \cdot 3\,\text{W})/1\,000 = 0.0174$ kilowatt-hours (which costs approximately 0.00105966$), while processing *role 1* requests for 1.7 hours requires about three times less money (approximately $(1.7 \cdot 3\,\text{W})/1000 = 0.0051$ kWhs, 0.00031059 dollars).

The conclusion is that the role-switching (the security level switching) has undeniable impact not only on CPU energy utilization but also on the amount of power consumed by cooling and, in fact, on all the working components of the machine, which strictly translates into cost savings. In this section, however, we focused only on the energy usage of the CPU as the main unit utilized for the cryptographic operations and made only a brief note about its influence on remaining energy consumers. Power consumption of other server components can be found on the manufacturer's websites and on the websites which perform independent hardware benchmarks and tests. The analysis of the optimal energy consumption of a single server along with all its components is left for the future work. Nevertheless, performed research provides serious argumentation to believe that the reduction of the CPU usage, and thus the amount of the utilized energy, entails significant economic profits. As stated before, cost savings are in fact even higher than those estimated here, since, saving the CPU power usage, we reduce the amount of energy utilized for the cooling system, as well as decreasing power consumption of all the server's physical components. Our study showed that, ensuring required security, it is possible to reduce the power consumption and increase cost savings at the same time. At first glance, figures presented here may seem irrelevant; however, when, put in the context of a large data center environment, they can quickly become very significant.

As it might seem on the basis of the so far performed analyses, the total cost of the data center utilization does not depend only on the power consumption of the server's CPU. Calculating the total cost of operating a data center one needs to take into account both fixed and variable costs, which are affected by complex and interrelated factors. From the economic analysis of the power consumed by the central processing unit, one can note that, with the exact number of served users, it is possible to save money, only by switching the level of protection from the *strongest* to the *weakest* one. The financial analysis performed earlier proved that, reducing protection mechanisms, one can expect significant financial profits. Analysing results estimated during our research, one can see that changing the level of protection it is possible to handle the required number of users and increase financial profits even more (since the reduction of the CPU load implicates the decrease of the power used by all server components, resulting in lower costs). However, in the economic stage of the analysis process, we need to look at power consumption, heating and cooling, and the data center footprint.

Determining the approximate, total cost of a whole data center, we assumed to use Dell PowerEdge R710 servers and supposed that the average lifetime of a single machine is equal to about three years. However, we did include neither the network nor the storage footprint (nor its equipment) in our estimation. Server-specific numbers (such as BTU) utilized in our evaluation were obtained from the technical guide of the Dell PowerEdge R710 server.

Although it does not seem to result directly from the economic analysis of a single machine, the presented solution can bring real, meaningful cost savings. In our approach economic profits actually come from the number of working machines together with its load factor. Modification in the configuration of utilized security mechanisms lets one obtain significant benefits. Presented approach brings additional possibilities: by switching security mechanisms from the *strongest* to the *weakest*, it is possible to provide effective services and maximize the utilization of hardware resources at the same time. Since we can accomplish exact goal using *weaker* security mechanisms, in many situations it is wasteful to assign too many hardware resources to perform the given task. Applying the proposed solution to the existing IT environment, one can observe serious reduction in IT costs while increasing the efficiency, utilization, and flexibility of their existing computer hardware. Table 9 explores this concept in more detail.

Table 9: CPU load equals 90%; number of connections to handle is given for the scenario. Estimated values represent the total annual cost of the energy and cooling usage, rounded up to the nearest dollar.

Scenario (users to handle ≈ 15 824 764 800)		
	Server(s)	$\varsigma_{\text{power+cooling}}$
Role 1	156	57 051$
Role 2	352	128 728$
Role 3	520	190 168$

As it was proved by the time analysis, server working with *role 3* permissions is able to handle about 3 474 users within an hour having 90% of CPU load. Since we assumed that the number of users grows linearly, within 24 hours, it gives us $3\,474 \cdot 24 = 83\,376$ employees a day, resulting in $83\,376 \cdot 365 = 30\,432\,240$ connections a year per server. If we assume that we have at our disposal the whole data center, it will turn out that we can serve roughly $30\,432\,240 \cdot 520 = 15\,824\,764\,800$ users assigned *role 3* permissions a year. To handle exact number of users assigned *role 1* privileges with 90% of CPU load and being aware that using *role 1* permissions server is capable of dealing with 11 571 users within an hour, it is trivial to compute number of physical machines capable of managing quite same number of employees, with different security mechanisms applied:

$$\text{users}_{\max_{S1R1}} \cdot \chi \cdot \rho \cdot \mu_{S1R1} \approx \text{users}_{\max_{S1R3}} \cdot \chi \cdot \rho \cdot \mu_{S1R3}, \qquad (9)$$

where $\text{users}_{\max_{S1R1}}$ is the maximum number of users assigned *role 1* permissions that can be handled within an hour by the single server, $\text{users}_{\max_{S1R2}}$ is the maximum number of users assigned *role 3* permissions that can be handled within an hour by the single server, χ is the total amount of hours when the single server was busy, ρ is the total amount of days when the single server was busy, μ_{S1R1} is the number of servers being capable of dealing with the given traffic (supposing managing $\text{users}_{\max_{S1R1}}$ employees an hour, having *role 1* privileges through χ hours for ρ days), and μ_{S1R3} is the number of servers being capable of dealing with the given traffic (supposing managing $\text{users}_{\max_{S1R3}}$ employees an hour, having *role 3* privileges through χ hours for ρ days), which, in our case, results in

$$\begin{gathered} 11\,571 \cdot 24 \cdot 365 \cdot \mu_{S1R1} \approx 3\,473 \cdot 24 \cdot 365 \cdot 520, \\ \mu_{S1R1} \approx 156. \end{gathered} \qquad (10)$$

From Table 9, it can be seen that, to handle about 15 824 764 800 connections (a year) with *role 3* permissions assigned, it is necessary to utilize all physical machines in the enterprise. However, exact amount of employees can be dealt with by about 156 servers, if we change the communication channel protection type to *role 1* privileges set. Thus, it is clear that power delivery and cooling savings can increase to roughly 70%.

As assumed earlier in the paper, performing day-to-day tasks, single server consumes on average about 0.3 kilowatts per hour. According to our model and analysis performed by the AQoPA tool, while processing defined security operations, CPU used up roughly about 0.04 kilowatts for each role, per all sessions within an hour, having 90% of CPU load. Using (1) and assuming that the cost of cooling is approximately equal to the cost of consumed energy, it is quite straightforward to estimate the total, annual cost of

Table 10: Approximate annual cost of energy consumption (in US dollars). Number of handled users is equal to 33 288 000 per server a year; CPU load varies between the roles.

Server(s)	Scenario		
	RBAC role		
	Role 1 (CPU load ≈ 29.55%)	*Role 2* (CPU load ≈ 66.80%)	*Role 3* (CPU load ≈ 98.43%)
	$\varsigma_{\text{power annual}} + \varsigma_{\text{cooling annual}}$	$\varsigma_{\text{power annual}} + \varsigma_{\text{cooling annual}}$	$\varsigma_{\text{power annual}} + \varsigma_{\text{cooling annual}}$
1	335	354	370
13	4 356	4 601	4 810
520	174 238	184 054	192 394

Table 11: Approximate annual environmental impact (in pounds of CO_2). Number of handled users is equal to 33 288 000 per server a year; CPU load varies between the roles.

Server(s)	Scenario		
	RBAC role		
	Role 1 (CPU load ≈ 29.56%)	*Role 2* (CPU load ≈ 59.36%)	*Role 3* (CPU load ≈ 98.45%)
1	5 117	5 405	5 650
13	66 519	70 267	73 451
520	2 660 770	2 810 679	2 938 038

the physical machine dealing with defined scenario. Table 10 comprises assessed costs for single machine, a rack, and the whole data center.

Being aware of the annual usage cost of the single server, we were hereby able to compute the amount of money spent on any number of working machines.

In conclusion, such security mechanisms switching can offer a variety of economic advantages: it permits one to increase the scale of server infrastructure without purchasing additional pieces of hardware and allows resources to be used more efficiently. In addition to savings in hardware costs, security level switching decreases the amount of floor space and maintenance expenditures. Such server consolidation also reduces the overall footprint of the entire data center. That means far fewer servers, less networking gear, a smaller number of racks needed, all of which translates into less data center floor space required. Consolidating server onto far fewer physical machines means lowering monthly power and cooling costs in the data center.

New Type of Analysis: The Carbon Dioxide Emissions Analysis. The energy usage estimation performed for the example call center can be a good start for the research on the efficient carbon dioxide emissions. The following analysis confirms the statement that the more energy we save, the less CO_2 our machine will produce. When it comes to the emissions of CO_2 analysis of an example IT environment, we proposed using (6) to estimate its produced volume. Values estimated with the help of (6) are summarized in Table 11. They refer to the single physical machine (the rack and the data center), performing actions defined in our security-based data flow management RBAC model, and represent the total annual environmental impact of server's energy usage, rounded up to the nearest pound.

Table 12: CPU load equals 90%; number of connections to handle is given for the defined scenario. Estimated values represent the total, approximate annual environmental impact (in pounds of CO_2), rounded up to the nearest pound.

Scenario (users to handle ≈ 15 824 764 800)		
	Server(s)	Pounds of CO_2
Role 1	156	871 218
Role 2	352	1 965 790
Role 3	520	2 904 045

Same as in the case of the financial and economic analyses, it is possible to estimate the amount of the carbon dioxide, which can be saved if only we perform the proposed security-switching. As it is apparent from Table 12, changing protection mechanisms from the *strongest* to the *weakest* ensuring required security and fulfilling all the security objectives at the same time, it is possible to decrease the amount of the carbon dioxide released to the atmosphere by about 9% (roughly pounds) a year per data center.

According to [35], the average amount of released carbon dioxide to the atmosphere per kWh was about 0.84 368 kg (1.86 pounds) in April 2014; therefore a single working server equipped with the considered CPU produced roughly 5 585 pounds of carbon dioxide a year (having ≈90% of CPU load and handling 3474 users assigned third role in the considered scenario). Regarding the whole data center, it released about 2 904 045 pounds of carbon dioxide to the atmosphere, which can be simply translated to 1 317 253 kilograms of CO_2. Comparing the above estimated values for *role 3* to those assessed when dealing with the first one, it can be seen that switching between the *strongest* and the *weakest* security mechanisms one can reduce the emissions of CO_2 by ≈9%, that is, about 277 268 pounds (125 767 kilograms) per data center.

In accordance with the information at [36], the annual amount of released carbon dioxide saved by switching between the roles (from *role 3* to *role 1*) for the data center is equivalent to about 27 passenger vehicles, 14 152 gallons of gasoline consumed, 135 088 pounds of coal burned per year, or 17.3 homes' electricity use for one year. However, bear in mind that the data presented here correspond to the average values. Let us have a closer look at, for instance, number of gallons of gasoline consumed (e.g., by the passenger vehicles). Here, the number of gallons of gasoline consumed depends on many different factors, that is, the type of the gearbox (automatic or manual), engine size, or even the weight and the shape of the car. More profound analysis requires consideration of all of these factors.

Steps 4 and 5 (recommendation and implementation). To serve as a specialized solution for secure systems, our decision support system took as an input characteristics of the environment's traffic, that is, the number of established connections, available services, utilized protocols along with their configuration details, and all the possible security mechanisms that can be used in the considered environment (encryption, decryption, hashing algorithms, and many more). Besides the mentioned components, to obtain the proper configuration of available parameters that is the most effective in given circumstances, one needs to provide hardware metrics as well. Since usually complex IT environments are characterized by high dynamism, it is reasonable to define the approximate time of the analysis (a month, a year, with, for instance, one-hour interval, when the traffic characteristic can change and the whole analysis process can be repeated). The goal of the system is the efficient management of the incoming connections in terms of utilized physical resources, system load, financial costs, CO_2 emissions, and the amount of consumed power. At the output, one receives the estimated number of physical machines capable of handling given traffic together with their assessed load. Results of the remaining analyses are available as well. Retrieved information enables users to find the most appropriate solution to their problems. Let us summarize advices suggested by our DSS for the proposed case study. When it comes to the time analysis, the conclusion is simple: if one is interested in security mechanisms that are the most *time-effective*, our decision support system indicates that TLSv1 is the best option. Looking for the security solution with the *lowest energy consumption*, our DSS suggests that RC4 and MD5 are the right choice. The variety of alternatives examined by the QoP analysis facilitates decision-making process as well. For instance, if we do care about the *confidentiality*, *integrity*, and *availability*, TLSv3 achieves the strongest possible security level (it is in fact *the most secure* among others). From the economic point of view, TLSv1 seems to be a quite interesting alternative: designed decision support system suggests, that if we choose RC4 and MD5 instead of AES/CBC 128 and SHA-1 or AES/CBC 256 and SHA-512 as utilized security mechanisms, we will save about 18 156 dollars per data center on average a year.

6.2. Security Objectives Realization. By choosing the role-based-access control model as the example realization of the data flow management, we answered questions about the *security management* and *access control* (SO1). We defined three roles, which differ in applied security mechanisms, and mapped them to established enterprise functions, in order to *provide management direction and support for information security in accordance with business requirements and relevant laws and regulations* [29].

As mentioned before, the TLS communication channel protection allowed us to fulfill the *confidentiality, authenticity*, *or integrity protection* security objective (SO2). While securing the communication channel, we utilized cryptographic algorithms and usage practices selected according to best practice.

Regarding the *system capacity management* (SO3), we performed the multilevel analysis according to the proposed DSS. We estimated and compared number of users that the server can handle during an hour, managing connections protected by different versions of the TLS protocol, and determined time and energy consumed by established sessions, along with assessing their economic and environmental impact. Such an approach allows us to adjust the security level to the capacity of the system while meeting defined security objectives. Performed, multidimensional analysis allows us to properly examine system capabilities and select those security mechanisms, which ensure security objectives realization together with decreasing financial costs and reducing negative environmental impact.

When it comes to the *assurance of correct and secure operation of information processing and handling facilities* (SO4), we modeled subprocesses responsible for secure information management: `AccessFTPServer`, `AccessWebServer`, `AccessAppServer`, `AccessServerS1`, `AccessServerS2`, `AccessGeneralOfficeAssets`, and `AccessDMZProductionServers`.

To maintain *the integrity and the availability of information* (SO5), we proposed modeling data center access (subprocess `AccessDataCenter`), which can simply be mapped to back-up process in the proposed, example implementation environment. The `AccessDataCenter` subprocess uses channel encryption to ensure the security of the operation and stores the back-up data at a secure, remote location.

Meeting the sixth security objective (SO6), we modeled an `AccessEmailServer` subprocess, which is responsible for providing a secure electronic message exchange.

Based on the gathered results, it is possible to implement the solution which satisfies given requirements best. It is also worth mentioning that each of the proposed roles (regardless of utilized security mechanisms) fulfills all the defined security objectives.

7. Conclusions

In the paper we used Quality of Protection Modeling Language to prepare the model of example business scenario for the enterprise having role-based access control management implemented. We defined a scenario with dissimilar levels of security and investigated the performance of the server handling miscellaneous number of users with different RBAC roles assigned. On the basis of the gathered results, we indicated that the user access control management has a meaningful impact on overall system's performance. Our research proved that drawing attention to the system's efficiency while implementing the role-based access control policy is crucial from the user access control point of view, usability, and security management. Furthermore, the ability of preparing the access control management security model in Quality of Protection Modeling Language confirmed its extensibility and flexibility with the role-based access control functionality. Our analysis showed that it is possible to achieve a balance between security and performance.

In addition, on the basis of the performed analyses, we proposed the foundations of the model-driven decision support system that helps one make the decision and determine which of the available security mechanisms meets given requirements best. Using introduced system, one can choose between several decision-making techniques, which can simply be translated into different methods of analysis. With the presented framework it is possible to find the fastest, the cheapest, the most secure, the most energy-efficient, or the most environment friendly solution and obtain estimated results that one can actually compare.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work is supported by the Polish National Science Centre Grant 2012/05/B/ST6/03364.

References

[1] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, "The cost of a cloud: research problems in data center networks," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 1, pp. 68–73, 2009.

[2] J. G. Koomey, *Estimating total power consumption by servers in the U.S. and the world [Ph.D. thesis]*, 2007.

[3] C. D. Patel and A. J. Shah, *Cost Model for Planning, Development and Operation of a Data Center*, HP Laboratories, Palo Alto, Calif, USA, 2005.

[4] The official web page of the QoPML project, 2012, http://www.qopml.org.

[5] A. O'Connor and R. Loomis, "Economic analysis of role-based access control," Tech. Rep., National Institute of Standards and Technology, 2010.

[6] "A comparison of security analysis techniques for RBAC models," in *Proceedings of the 2nd Annual CCWIC*, 2010.

[7] A. K. Agarwal and W. Wang, "On the impact of quality of protection in wireless local area networks with IP mobility," *Mobile Networks and Applications*, vol. 12, no. 1, pp. 93–110, 2007.

[8] B. Ksiezopolski and Z. Kotulski, "Adaptable security mechanism for dynamic environments," *Computers & Security*, vol. 26, no. 3, pp. 246–255, 2007.

[9] E. LeMay, W. Unkenholz, D. Parks, C. Muehrcke, K. Keefe, and W. H. Sanders, "Adversary-driven state-based system security evaluation," in *Proceedings of the 6th International Workshop on Security Measurements and Metrics (MetriSec' 10)*, ACM, September 2010.

[10] S. Lindskog, *Modeling and tuning security from a quality of service perspective [Ph.D. thesis]*, Chalmers University of Technology, Gothenburg, Sweden, 2005.

[11] A. Luo, C. Lin, K. Wang, L. Lei, and C. Liu, "Quality of protection analysis and performance modeling in IP multimedia subsystem," *Computer Communications*, vol. 32, no. 11, pp. 1336–1345, 2009.

[12] C. S. Ong, K. Nahrstedt, and W. Yuan, "Quality of protection for mobile multimedia applications," in *Proceedings of the International Conference on Multimedia and Expo (ICME '03)*, vol. 2, pp. 137–140, Baltimore, Md, USA, July 2003.

[13] D. C. Petriu, C. M. Woodside, D. B. Petriu et al., "Performance analysis of security aspects in UML models," in *Proceedings of the 6th International Workshop on Software and Performance (WOPS '07)*, pp. 91–102, ACM, February 2007.

[14] P. A. Schneck and K. Schwan, "Authenticast: an adaptive protocol for high-performance, secure network applications," Tech. Rep., Georgia Institute of Technology, 1997.

[15] Y. Sun and A. Kumar, "Quality-of-protection (QoP): a quantitative methodology to grade security services," in *Proceedings of the 28th International Conference on Distributed Computing Systems Workshops (ICDCS '08)*, pp. 394–399, IEEE Computer Society, Beijing, China, June 2008.

[16] U. R. F. Averweg, *Decision-Making Support Systems: Theory & Practice*, bookboon.com, 2012.

[17] D. J. Power, "A brief history of decision support systems," http://dssresources.com/history/dsshistory.html.

[18] D. J. Power and R. Sharda, "Model-driven decision support systems: concepts and research directions," *Decision Support Systems*, vol. 43, no. 3, pp. 1044–1061, 2007.

[19] Z. Shi, "Knowledge-based decision support system," *Journal of Computer Science and Technology*, vol. 2, no. 1, pp. 22–29, 1987.

[20] R. Matulevičius, H. Lakk, and M. Lepmets, "An approach to assess and compare quality of security models," *Computer Science and Information Systems*, vol. 8, no. 2, pp. 447–476, 2011.

[21] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Computer role-based access control models," *Computer*, vol. 29, no. 2, pp. 38–47, 1996.

[22] T. Lodderstedt, D. Basin, and J. Doser, "SecureUML: a UML-based modeling language for model-driven security," in *(UML) 2002—The Unified Modeling Language*, vol. 2460 of *Lecture Notes in Computer Science*, pp. 426–441, Springer, Berlin, Germany, 2002.

[23] J. Jürjens, *Secure System Development with UML*, Springer, New York, NY, USA, 2007.

[24] N. Sklavos, P. Kitsos, K. Papadopoulos, and O. Koufopavlou, "Design, architecture and performance evaluation of the wireless transport layer security," *The Journal of Supercomputing*, vol. 36, no. 1, pp. 33–50, 2006.

[25] A. Stubblefield, A. D. Rubin, and D. S. Wallach, "Managing the performance impact of web security," *Electronic Commerce Research*, vol. 5, no. 1, pp. 99–116, 2005.

[26] J. Jürjens, "Security and compliance in clouds," in *Proceedings of the 4th Pan-European Conference*, IT-Compliance, 2011.

[27] R. M. Savola, "Quality of security metrics and measurements," *Computers & Security*, vol. 37, pp. 78–90, 2013.

[28] B. Ksiezopolski, "QoP-ML: quality of protection modelling language for cryptographic protocols," *Computers & Security*, vol. 31, no. 4, pp. 569–596, 2012.

[29] International Standard ISO/IEC 27002, "Information technology—security techniques—code of practice for information security controls," 2013.

[30] K. Mazur, B. Ksiezopolski, and Z. Kotulski, "The robust measurement method for security metrics generation," *The Computer Journal*, 2014.

[31] B. Ksiezopolski, T. Zurek, and M. Mokkas, "Quality of protection evaluation of security mechanisms," *The Scientific World Journal*, vol. 2014, Article ID 725279, 18 pages, 2014.

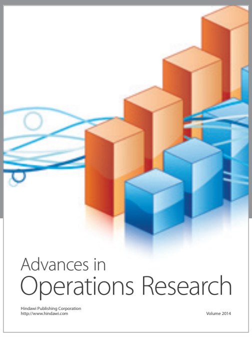
[32] D. Rusinek, B. Ksiezopolski, and A. Wierzbicki, "Security tradeoff and energy efficiency analysis in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 943475, 17 pages, 2015.

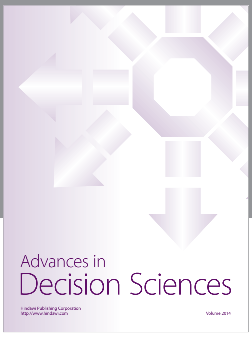
[33] http://ark.intel.com/products/52577/Intel-Xeon-Processor-X5675-12M-Cache-3_06-GHz-6_40-GTs-Intel-QPI.

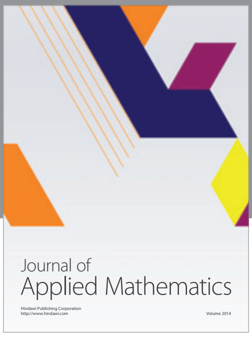
[34] R. Chheda, D. Shookowsky, S. Stefanovich, and J. Toscano, "Profiling energy usage for efficient consumption," *The Architecture Journal*, vol. 18, pp. 24–27, 2008.

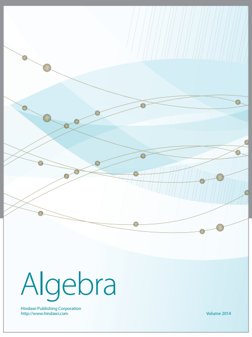
[35] U.S. Energy Information Administration, "How much carbon dioxide is produced per kilowatthour when generating electricity with fossil fuels?" http://www.eia.gov/tools/faqs/faq.cfm?id=74&t=11.

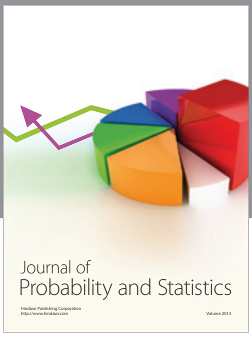
[36] Inventory of U.S. Greenhouse Gas Emissions and Sinks: 1990–2011, Chapter 3 (Energy), Tables 3-12, 3-13, and 3-14, U.S. Environmental Protection Agency, Washington, DC, USA, Inventory of U.S. Greenhouse Gas Emissions and Sinks: 1990–2011, Annex 2, Methodology for estimating CO2 emissions from fossil fuel combustion, Table A-36. U.S., Environmental Protection Agency, Washington, DC, USA, greenhouse gas equivalencies calculator, http://www.epa.gov/cleanenergy/energy-resources/calculator.html.


Advances in
Operations Research

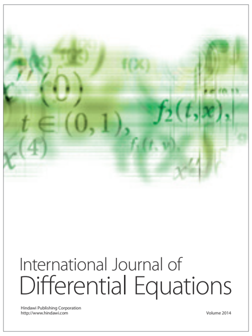

Advances in
Decision Sciences


Journal of
Applied Mathematics

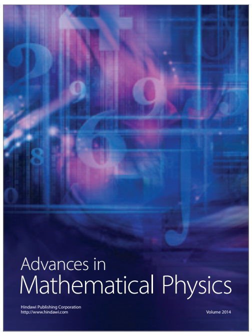

Algebra

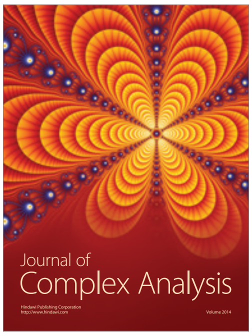

Journal of
Probability and Statistics

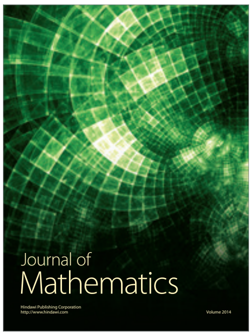
The Scientific
World Journal

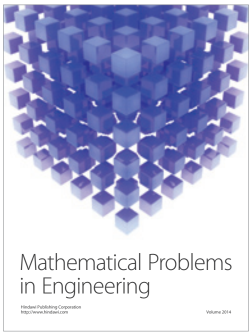

International Journal of
Differential Equations

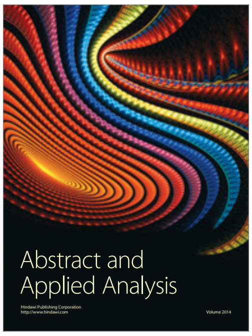
International Journal of
Combinatorics

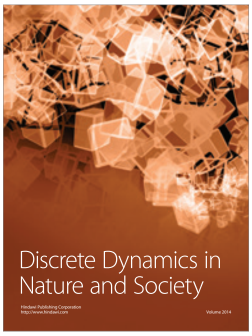

Advances in
Mathematical Physics

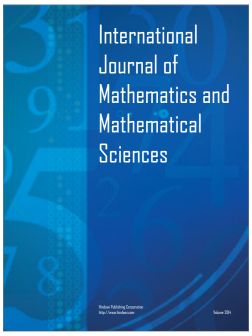

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

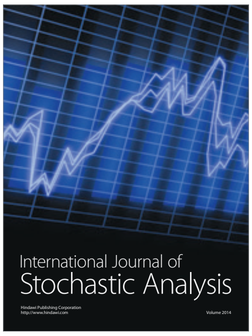

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization