A Semantic Model for Security Evaluation of Information Systems

Elena Doynikova, Andrey Fedorchenko and Igor Kotenko*

St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, St. Petersburg, Russia
E-mail: doynikova@comsec.spb.ru; fedorchenko@comsec.spb.ru; ivkote@comsec.spb.ru; ivkote1@mail.ru
**Corresponding Author*

Received 18 January 2020; Accepted 18 January 2020;
Publication 23 March 2020

Abstract

Modern information systems are characterized by huge security related data streams. For cyber security management in such systems, novel models and techniques for efficient processing of these data streams are required. The paper considers development and application of a semantic model for security evaluation. The proposed model is represented as the ontology of metrics that is based on the relations between sources of security related data, primary features of initial security data and goals of security evaluation. The set of hierarchically interconnected security metrics is mapped to the data features and security evaluation goals. The relations between these metrics within the proposed ontology provide the basis for security evaluation technique. The paper introduces the proposed ontology and its foundations, and briefly describes the developed technique. The analysis of data in the open sources is conducted, and the case study is provided to show applicability of the approach.

Keywords: Security evaluation, security assessment, semantics, metrics, ontology, cyber attacks, intelligent data analysis.

Journal of Cyber Security and Mobility, Vol. 9_2, 301–330.
doi: 10.13052/jcsm2245-1439.925

1 Introduction

In recent years the need to develop new methods in the field of intelligent data analysis become more obvious, and it relates to the cybersecurity area as well.

It follows from the fact that a lot of monitoring tools have been developed by the current moment. These tools allow gathering a huge amount of data on the analyzed information system, including gathering a huge amount of events occurring in the system. Prominent representatives in this area are Security Information and Event Management (SIEM) systems. Besides, a lot of security related knowledge databases were developed. In particular, National Institute of Standards and Technology (NIST) [1] has made a great contribution in this area. It maintains National Vulnerability Database (NVD) [2]. Another important organization in this area is MITRE corporation [3]. It supports such initiatives as, for example, Common Weakness Enumeration (CWE) [4], that is the database of software weaknesses, Common Attack Pattern Enumeration and Classification (CAPEC) [5], that is the database of attack patterns, and other security databases.

At the same time, there is a challenge for modern cybersecurity community that is how to process all available security related information to increase security of information systems from cyberattacks.

The relevance of this task also raises no questions as soon as society is increasingly dependent on information systems. Information systems are used to control all key aspects of society, including healthcare, finance, housing (smart home concept in scope of more wide IoT concept), transport (smart cars), industry (industrial internet of things), and many others. On the other hand, the relevance is confirmed by statistics on cyberattacks and statistics on losses from them, including such attacks of recent years as Ghost Net in 2009, Stuxnet worm in 2010, Spamhaus DDOS-attack in 2013, Carbanak in 2015, Petya ransomware in 2017, and many others.

In recent years the researchers and producers of the security management tools focused on intelligent data analysis. The promising approach in the area of intelligent data analysis is semantic approach, as soon as it allows structuring existing knowledge in the subject area, tracing dependencies between different objects, processes, and events, and concluding on the causes and consequences of various events based on the revealed interconnections. An interesting initiative in this area is the Cyber Security Body Of Knowledge (CyBOK) project that outlines main areas and aspects of cybersecurity [6]. As soon as the approach proposed in this paper requires an accurate structuring

and detailing security related knowledge, the areas outlined in CyBOK (including software security, network security, malware, and others) and their further detailing can be used as the basis for development of our approach. At the same time, the proposed in this paper approach is based on the set of hierarchically interconnected security metrics for security assessments [6–8]. The structure of subject areas introduced in CyBOK will allow one to evaluate what cybersecurity areas are covered with metrics by our approach.

The proposed approach assumes using an ontology of security metrics to trace dependencies among available security data sources, available raw security data, metrics calculated on their base (divided by the security assessment goals) and security assessment goals. From our point of view the technique that uses logical inference based on the mentioned dependencies will allow answering security related questions considering all available data and their interconnections, and to overcome the challenge of knowledge processing in the security management area.

The basics of our approach are as follows:

1. An ontology of security data sources and security data [8]. The difference of ontology proposed in this study consists in extension with a new class of instances, namely, security metrics. Representation of security metrics as separate instances of ontology allows using relations between the metrics, security data and security data sources for calculating the integral metrics reflecting the security state of analyzed system.
2. A set of hierarchically interconnected security metrics [8, 9]. These metrics allow assessing security of information systems on different stages of their operation and with varying degrees of accuracy depending on the available security data, new knowledge obtained in the process of security assessment, and security assessment goals.
3. The foundations of ontology of security metrics [9] that was focused on one security question, namely, "what is an attack goal?". We extended it with new security metrics and detailed it in terms of their interrelations. The current version of the ontology is intended to answer other security questions as well, thus, it is more universal.

The contribution of this paper is as follows:

- The so-called ontology of security metrics.
- The technique for its application to answer security related questions.
- The case study that demonstrates application of the technique and ontology.

The ontology and the related technique were firstly introduced in the paper presented on the IWCC'2019 workshop held in conjunction with the ARES'2019 [10]. In this paper we are extending the results of this paper, including related works, specification of the approach, and the case study.

The ontology combines the sources of security data, objects of security assessment process, and security metrics. These objects are interconnected in a way to efficiently calculate a complex of different security metrics. There are different standards for security assessment and countermeasure selection, the ontologies of subject area, the techniques and metrics for security assessment, the security information and events management systems, and security databases. But there is no an integrated efficient automated adaptive mechanism to develop and calculate security metrics for security assessment and countermeasure selection that is applicable for systems of different types, considering conditionally unlimited amount of initial information related to security. As it was already mentioned, we propose the ontology of security metrics as a such mechanism. Analysis of related research shown that currently the ontology with the described characteristics does not exist.

The paper is organized as follows. In Section 2 the related works are considered. Section 3 provides foundations of the ontology including analysis of open sources of security related data, as well as introduces the ontology and the inference technique on its basis. In Section 4 some statistics is given to prove hypotheses underlying the ontology, an application of the ontology is demonstrated on a case study and the obtained results are discussed. The paper ends with conclusion.

2 Related Work

As it was mentioned in the introduction, currently there are a lot of sources of security related data and a lot of tools for their gathering. Let us to outline the following types of related works: security monitoring tools, sources of security related data, and security related ontologies and techniques.

The security monitoring tasks have been researched for few decades. There are a lot of research works in this area. The research results were practically implemented within SIEM systems that are the tools for processing and analysis of security events from different sources for improving organization's security management. Currently the research focus changed to the intelligent data processing. Particularly, the User and Entity Behavior Analytics (UEBA) systems become popular. In contradistinction to SIEM systems, these systems use analytical methods (including machine learning)

to detect anomalies indicating security violations. Currently there are many commercial solutions in the field of UEBA systems, including Securonix UEBA, Micro Focus Security ArcSight UBA, Splunk User Behaviour Analysis, Forcepoint UEBA, Exabeam Advanced Analytics. As well as there are many commercial solutions in the field of new generation SIEM systems, including the solutions that use analytical methods: QRadar SIEM from IBM [11], ArcSight ESM from Micro Focus [12], Splunk Enterprise Security from Splunk [13], LogRhythm NextGen SIEM Platform from LogRhythm [14], and others.

Experience with such solutions has shown that machine learning methods are not sufficient for effective security monitoring, security incidents analysis, determining and analyzing the causes and consequences of attack actions, and responding to incidents. To expand the capabilities of SIEM and UEBA, semantic models can be used. We assume that the ontology we are developing and technique of its application will form the basis of the new generation SIEM systems.

Additionally to SIEM systems the sources of the security related data can include intrusion detection and intrusion prevention systems, databases of system assets, security scanners, knowledge and security databases, and others. Currently there are a lot of security related knowledge and data bases. Particularly, the already mentioned CyBOK project is security related knowledge base [6].

Security databases are any sources of security data that can be used within security management tasks. At the moment, rather extensive list of sources of security data can be constructed, including the data on attacks (CAPEC) [5], weaknesses (CWE) [4], software and hardware (NVD) [2], vulnerabilities (Common Vulnerabilities and Exposures (CVE) database [15], NVD, Open Source Vulnerabilities Data Base (OSVDB) [16], Vulnerability Notes Database (VND) [17], SecurityFocus project with BugTraq [18], and IBM X-Force [19]), exploits (Exploit DataBase, EDB [20], Metasploit [21]), configurations (NVD). Additionally to description of the appropriate objects these databases contain different security metrics, what is of interest for our research. Especially it relates to the vulnerability metrics from NVD, attack metrics from CAPEC and weaknesses metrics from CWE.

Semantic models and approaches are used to solve problems in various areas including security management. Particularly, ontologies [22–24] showed themselves as a good way to integrate information. There are ontologies designed to solve particular security tasks, including the vulnerability-centric ontologies for security analysis [25–27], ontologies for

security decision support [28, 29], a common ontology for Security Content Automation Protocol (SCAP) that is developed for automation of security management [30].

The most universal cyber security ontology from our point of view is an Unified Cybersecurity Ontology (UCO) [31, 32]. UCO integrates various security information for security assessment. It uses the standards CVE [15], CWE [4], CAPEC [5], Common Configuration Enumeration[1] (CCE), etc. It differs by the fact that it includes instances representing information and communication objects (i.e. files, network addresses, processes, operation systems, etc.). Besides, it includes instances representing network state and information about an attacker. The disadvantages of this model for our goals are as follows: it does not allow integrating information from different sources of the same type; it requires manual setup while implementing for the specific system; and representation of the system state in real time requires the model modification on the fly that is not a trivial task.

In [33] the authors propose access control ontology and an approach to distribution of access requests on its base. The model considers the relationships between all areas of access control including subjects, objects and actions (i.e. grant or revoke). This model can be used to extend our ontology for security management in the future work.

In [34] the authors provide the security framework for decision support to increase the security of industrial systems. The authors argue that the developed model should be adaptive as soon as the Internet of Things (IoT) is characterized by high variability. It means that it should be possible to modify a security decision support system on fly, i.e. it should be possible to add new security attributes. The authors notice that a Model-Driven Development (MDD) has the required properties and allows developing adaptive systems using an adaptive model. They also suppose that connection between MDD and Ontology-Driven Development (ODD) allows using a formal model suitable for vulnerability detection, risks forecasting and assessment, and intrusion detection in real time. The proposed ontology combines assets, vulnerabilities and their severity level, threats and OSI level that they affect, security tools including their characteristics, types, and related security properties. The disadvantages of this model for our goals are as follows: the proposed ontology is limited by the IoT, while we consider the systems of any type; the proposed ontology is a rather high level one, i.e. it is not detailed

[1] https://nvd.nist.gov/config/cce/index

enough, including in terms of security metrics; the proposed ontology does not consider security data sources.

In the thesis [35] the Association Rule Interactive Post-Processing using rule schemas and Ontologies (ARIPSO) approach is proposed. It combines knowledge discovery in databases, namely, the association rule mining technique, and knowledge engineering to integrate users' knowledge and consequently to decrease the number and enhance the quality of rules. The underlying model implies integration of user domain knowledge (the user specifies concepts he/she knows), user expectations (the user specifies rules he/she needs), and operations (the user maps the actions to each expectation, these actions should be implemented if the rule is met). Further, the association rules are applied considering the model specified by the user. It decreases the number of rules. This idea can be used in future to extend our approach for generation of ontologies that satisfy the security goals of specific organizations.

In [36] the ontology for security assessment and countermeasure selection is proposed. The authors focus on representation of known attacks and use the following sources: MITRE [3] sources, Open Web Application Security Project (OWASP)[2] results and Web Application Security Consortium (WASC)[3] results. They connect information from these sources with attack steps. The limitation of this approach is that only known attacks are considered. Besides, it does not allow constructing and processing the ontology dynamically. Thus, the high skills and time costs are required to represent known attacks. Though it is an interesting and useful initiative, processing of all known attacks will take huge amount of resources.

Thus, to this moment the ontological approach demonstrated application prospects for security management tasks. There are various ontologies in this area, but they have some disadvantages, namely, cover only limited number of object types, while for creating a complete picture of the system security state all set of objects of subject area should be considered including their interrelations. Besides, there are ontologies that consider specific types of systems, while we aim to create an universal model. Also, the existing models usually require manual setup and do not allow modification on fly. The global goal of our research is to develop an ontology that will allow calculating security metrics that answer on the important security questions using inference mechanism based on the relations between data sources,

[2]https://www.owasp.org/index.php/About_The_Open_Web_Application_Security_Project
[3]http://www.webappsec.org/

objects of security assessment subject area, and primary and integral security metrics. Considering the analysis of related works this task is not solved to this moment. In the future we also plan to introduce dynamics in our model through adding of security events and incidents obtained, for example, from a security information and event management system. Currently only one approach from the reviewed considers security events. But in scope of that approach the events are added manually during integration of the semantic model with MDD [34]. We believe that introduction of security events and incidents in the ontology itself will allow us to increase its adaptability and automate its modification on fly.

Finally, at this moment there are publicly available databases of security data, there are ontologies intended for integration of various types of data, and there are numerous metrics for security management. All this creates required basis for our ontology of security metrics. We started this research in [8, 37] where we aimed to integrate various security data sources, and proceeded it in [9], where we proposed the first upper level ontology for attack goals determination. This paper extends the previous ontology with new security metrics and details it in terms of their interrelations to answer security questions using logical inference.

3 Ontology of Metrics

The proposed ontology combines sources of security data, objects of security assessment process, and security metrics. It is described in Section 3.2. Sources of data incorporate currently available security databases, such as the weaknesses database CWE [4], the attack patterns database CAPEC [5], the database of vulnerabilities, configurations and platforms NVD [2], etc. Sources are considered in details in Section 3.1. Objects of security assessment incorporates such items as “weakness”, “attack”, “vulnerability”, “configuration”, “product”, etc. Finally, security metrics are classified per objects, sources and security related questions. They are described in Section 3.3.

3.1 Sources of Security Data

As it was mentioned above, we use security databases as the sources of security related data. In the introduction an extensive list of sources was given, including CAPEC database of attack patterns, CWE database of weaknesses, NVD database of products, configurations and vulnerabilities, CVE,

OSVDB, VND, SecurityFocus project with BugTraq, and IBM X-Force databases of vulnerabilities, EDB and Metasploit databases of exploits.

We analyzed the schemes of these databases to extract security related data and their interconnections. The NVD, CAPEC and CWE are the most interesting databases from the interconnections and metrics point of view.

The NVD database is the most complete dictionary of the software and hardware represented in the Common Platform Enumeration (CPE) format, their configurations in CCE format and vulnerabilities in the CVE format. The vulnerabilities in NVD are specified using properties that can be divided in four groups: identifying properties (such as description), identifying metrics (properties of the CVE format, such as vulnerability's id, publication date, modification date), evaluation metrics (vulnerability's scores), and references. References incorporate links to other sources (e.g. links to weaknesses in the CWE database). It allows connecting products with weaknesses, i.e. determining weaknesses of the system under analysis for further selection of means to increase its security.

The CWE database is the most complete dictionary of the software and hardware weaknesses that can lead to the vulnerabilities. Weaknesses are described by the set of fields specified by the CWE scheme [38]. There are reference fields that contain links to other databases, namely:

- Related Attack Patterns (links to attack pattern instances in other databases). This field allows connecting weaknesses in CWE with attack patterns in CAPEC.
- Observed Example (links to the real-world examples of exploitation of weaknesses, usually there are vulnerabilities in the CVE format). This field can be used to connect NVD and CWE databases, but rather small number of examples prevents its real application.
- Applicable Platforms (this field represents languages, operation systems, architectures, paradigms, and technologies that can result in weaknesses). The operation system can be represented using CPE. But rather small number of filled using CPE fields doesn't allow applying this field for connecting software and hardware with its weaknesses.

It should be noticed that before application of this database data preprocessing is required, aimed at their normalizing and eliminating errors and inconsistencies.

The CAPEC database incorporates descriptions of methods and ways for exploitation of software or security system weaknesses to implement cybersecurity threat [5]. Within CAPEC the attack patterns are joined in categories,

and categories are joined in catalogues. Catalogues are constructed depending on views. There are views by attack technique or by attack scope. The attack patterns are represented using parameters and their values that are specified by CAPEC scheme [39, 40]. The key types of elements of the scheme are as follows: entities, elements, elements-references, enumerations, attributes. A catalogue, a view, a category and an attack pattern are entities. They are further specified using elements and attributes, and their possible values that are given using enumerations. Elements-references are used to specify the structure of CAPEC database or for the external links (to other security related data sources).

For example, attack patterns are specified using following elements: Description, Execution_Flow, Typical_Severity, Likelihood_Of_Attack, Prerequisites, Skills_Required, Resources_Required, Indicators, Consequences, Mitigations, Example Instances, Related_Weaknesses, Taxonomy_Mappings, Alternate Terms, Related_Attack_Patterns, References, Notes, Content_History; and following attributes: ID, Abstraction, Status. Where Example Instances, Related_Weaknesses, Taxonomy_Mappings, Related_Attack_Patterns, and References are reference elements.

Elements and their values from various security related databases are the basis for primary metrics in our ontology that is shown in Section 3.3, while reference elements from different databases are the basis for relations in our ontology. Thus, the products provided in the NVD database in the CPE format have vulnerabilities provided in the CVE format. Vulnerabilities in NVD have links to weaknesses in the CWE database. It allows connecting products with weaknesses. Weaknesses in CWE, in their turn, have links to attack patterns in CAPEC. These connections are used in the ontology under development for logical inference to answer security related questions.

3.2 Ontology of Metrics

The proposed ontology is the basis of our approach to security assessment and countermeasure selection. The ontology combines four basic classes of concepts:

1. the class of data sources,
2. the class of security information,
3. the class of infrastructure objects that participate in security management process, and
4. the class of security metrics.

One of the novel features of the developed ontology is that security metrics are outlined in the separate classes, i.e. each metric is a separate concept, while valued metric is a separate instance (object of the class). As the result, the metric instances are connected with objects (security information and infrastructure objects) and data sources via object properties (that describe the type of relation between the concepts and instances) instead of data properties (that describe information that is specific for the concept or instance). For example, the concepts "Exploit" and "Vulnerability" are connected via the object property "implements": Exploit implements Vulnerability. In the previous version of ontology we represented vulnerability metrics as data properties [8], for example, the "Vulnerability" concept had the data property "CVSSv2" (metrics of the Common Vulnerability Scoring System, CVSS, of version 2 that evaluate vulnerabilities [41, 42]).

Currently, we specify this metric via the object property "evaluates" as follows: CVSSv2 evaluates Vulnerability. It allows one to construct the following sequence of links:

1. the link of integral metrics that represent system security state with primary security metrics;
2. the link of primary security metrics with security information and objects of security management area;
3. the link of objects of security management area with data sources, and, consequently,
4. the link of security metrics with data sources.

This allows linking the metrics of various objects and using these links to calculate security metrics using logical inference, i.e. get new knowledge in the security assessment area. The advantages of the proposed ontology and the cyber security assessment approach built on its basis are extensibility in terms of metrics (i.e. we can easily add and link new metrics) and universality (i.e. the proposed ontology can be used for security assessment of systems of various types). The extensibility of the ontology allows one to create the complete system of interconnected known security metrics in the future. The ontology incorporates concepts and relations between them. The following types of relations can be specified:

1. the relations of the class inherence hierarchy including parental relations between concepts and membership relations between concepts (types) and specific data sources;
2. the relations between the metrics and concepts, objects and concepts, and metrics and objects concepts (object properties); and

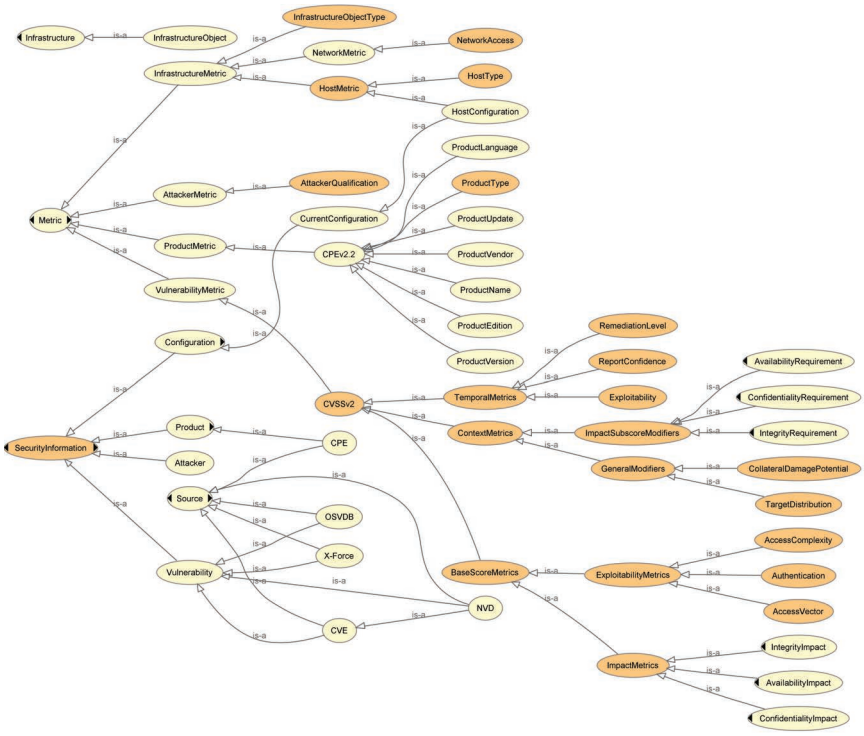

Figure 1 The simplified inheritance hierarchy of the proposed ontology.

3. the relationships between entities and possible variants of description of their individuals (data properties) [8].

The fragment of the simplified inheritance hierarchy of the proposed ontology is provided in Figure 1. The ontology model is implemented using OWL (Web Ontology Language) of version 2 and the description logic of the type DL (Descriptive Logic).

In accordance with OWL2 the root of hierarchy is “Thing” entity (not represented in Figure 1). The ontology combines the following four main classes of concepts: “Source” (data source), “SecurityInformation” (security information), “Infrastructure” (infrastructure object) and “Metric” (security metric).

The first group of concepts incorporate the security data sources, namely, the weaknesses database CWE, the attack patterns database CAPEC, the database of vulnerabilities CVE, the database of vulnerabilities,

configurations and platforms NVD [2], the database of exploits ExploitDB [20], and others [37]. In Figure 1 these concepts are combined by the root entity "Source".

The second group of concepts incorporate security information, namely, "product", "configuration", "weakness", "attack", "attacker", "vulnerability", "countermeasure", "exploit" (some concepts, namely "exploit", "countermeasure", etc. are omitted in Figure 1 to simplify the figure) [8, 9]. In Figure 1 these concepts are combined by the root entity "SecurityInformation". The subclasses of the listed concepts are omitted in order not to overload the figure. For example, for "attack" entity it will be "attack step". The "attack" entity and "attack step" are connected via parental relation. While "attack" and "CAPEC" entities are connected via membership relation. And "vulnerability" and "product" are connected using relations between concepts and/or entities (object properties), namely, via property "implementedIn": Vulnerability implementedIn Product.

The third group of concepts incorporates the infrastructure objects such as "network", "workstation", etc. In Figure 1 these concepts are combined by the root entity "Infrastructure". The specific concepts are omitted to simplify the figure.

The fourth group of concepts incorporates the security metrics. On the one hand, we outline different classes of metrics considering the objects they evaluate [37]: infrastructure metrics, attack metrics, attacker metrics, event metrics, response metrics and integral metrics. In its turn, each class of metrics contains subclasses. For example, infrastructure metrics contain network metrics (including access metrics and connectivity metrics) and host metrics (including host type and host configuration). Metrics are also connected with each other and objects using relations between concepts and/or entities (object properties). On the other hand, we outline different types of metrics. Namely, we outline identifying metrics, i.e. metrics that uniquely identify an object among the other objects.

For example, the concept "Product" has the following identifying metrics considering Common Platform Enumeration (CPE) standard [42]: "part", "vendor", "product", "version", "update", "edition", and "language". In this terms, an attack goal is attack metric, that can get the values "challenge, status, thrill", "political gain", "financial gain" and "damage" [9]. Further, we plan to clarify these values using the metrics provided in the CAPEC database for attack patterns. Besides, we outline evaluation metrics that evaluate object from the security aspects' point of view. For example, CVSS metrics [41, 42].

Figure 2 Relationships between the classes of metrics and the classes of objects.

The interconnections between the classes of metrics and the classes of objects are represented in Figure 2. Dotted lines denote inheritance relationships of the ontology's classes, while dashed lines represent equivalent classes. Object properties between the classes are denoted by the solid lines and have appropriate labels. The domain of the object property is the class nearest to the label. For example, the "Vulnerability" class is the domain of the "usedBy" property, while the "Attacker" class is the range of its possible values, i.e. "CVE-2015-5374 usedBy Attacker_1".

The top-level classes can include subclasses of metrics. Some ontology's concepts (classes) that represent the types of metrics and the classes of metrics, as well as some relations, are omitted in Figure 2 in order to simplify the figure. The classes and subclasses of metrics include the set of metrics that identify and evaluate objects of the corresponding class. For example, the concept "Vulnerability" is connected with the "VulnerabilityMetric" class: "Vulnerability has VulnerabilityMetric" (this link is not represented in Figure 2). This class contains, in its turn, "CVSSv2" subclass: "CVSSv2 is-a VulnerabilityMetric". "Vulnerability" and "CVSSv2" are connected via the "NVD" concept, i.e. not every vulnerability has a CVSS metric, but every vulnerability in the NVD database has a CVSS

metric: “NVD is-a vulnerability” and “NVD hasCVSSv2Metric CVSSv2” (Figure 2). The “CVSSv2” subclass includes the “BaseScoreMetric” (vulnerability base score), as well as “Temporal” (vulnerability score in time) and “Context” (vulnerability score considering environment) metrics that are not represented in Figure 2. In its turn, the “BaseScoreMetric” metric is calculated using the “ImpactMetric” (damage for the security properties from the vulnerability exploitation) and “ExploitabilityMetric” (likelihood of vulnerability exploitation) metrics: “ImpactMetric is-a BaseScoreMetric” and “ExploitabilityMetric is-a BaseScoreMetric”. “ImpactMetric” is calculated using the “IntegrityImpact”, “ConfidentialityImpact” and “ExploitabilityImpact” metrics. “ExploitabilityMetric” is calculated using the “Authentication” (shows if additional authentication is required to exploit the vulnerability), “AccessComplexity” (represents complexity of vulnerability exploitation) and “AccessVector” (depicts if the vulnerability can be exploited remotely) metrics [41, 42].

It should be noticed that all these metrics are available in the vulnerability databases including the integral vulnerability metric “BaseScoreMetric” (base CVSS score).

Let us consider another case, when integral metric is not available in the security database but it is calculated using available primary metrics. An attacker is characterized by the attacker skill level. “Attacker skill level” is an integral metric that represents attacker skills and should be calculated on the basis of metrics of other objects, i.e. it can be calculated on the basis of complexity of his/her attack steps. Considering that attacker implements the attack that consists of the attack steps that implement the vulnerabilities, we calculate “Attacker skill level” as the maximum “AccessComplexity” of the vulnerabilities that are implemented by the attack steps. Thus, the “Attacker skill level” is calculated using the connections between the attacker, the attack, the attack steps and the appropriate vulnerabilities. It should be noticed that “Attacker skill level” also depends on other metrics, such as “tools complexity”, “steps success rate”, “trace coverage rate”. Therefore, the ontology should be extended in the further work. The flexible model structure of the provided ontology allows one to add new security metrics without modification of the already existing statements.

To calculate integral security metrics on the basis of the proposed ontology we suggest using the ontological inference technique. It supposes collection of available security data and assigning values to the known metrics first (obtained from the security databases), bypassing security data, objects and metrics via links (starting from the already valued concepts)

and determining the calculation mechanism for the unknown integral metrics considering the logical types of links.

3.3 Security Metrics from Open Data Sources

As it was mentioned above, we use security databases as the source of data on primary security metrics and then use the latter to calculate integral metrics.

We analyzed the schemes of several databases to extract security metrics. Namely, we analyzed in details the NVD and CVSS metrics that it contains, the CWE database and underlying scheme, the CAPEC database and underlying scheme, and briefly analyzed the CVE database, the CCE and CPE dictionaries in scope of the NVD, ExploitDB database and X-Force database [19].

CVSS metrics score vulnerabilities on the scale from 0 to 10 depending on their severity for the analyzed system. These are metrics of evaluation type. The detailed analysis of CVSS is provided in [44, 45]. The CVSS metrics were discussed above and are already added to the ontology in Figure 1.

Besides, there are identifying metrics for the vulnerabilities that can be found in the CVE database. These metrics are as follows[4]: "vulnerability identifier", "version" (in its turn, specified by the metrics "version" and "date of release", where "version" is a number or range of numbers), "ProblemType" (description of the problem), "Description" (description of the vulnerability), and "AssigningCNA" (organization assigned the vulnerability identifier).

The CWE metrics[5] specify and assess weaknesses. They combine identifying metrics and evaluation metrics. Identifying metrics include the following metrics:

- weakness identifier,
- weakness name,
- potential mitigations (mitigations for the weakness),
- exploitation factors (factors that increase the "likelihood of exploit" for the weakness),

While evaluation metrics includes the following metrics:

- use frequency,
- detection complexity,

[4]https://cve.mitre.org/cve/cna/rules.html#Appendix_B
[5]https://cwe.mitre.org/documents/schema/index.html

- elimination complexity,
- likelihood of exploit,
- memory,
- system process,
- common consequences (including scope, impact, likelihood),
- functional area (possible values: Authentication, Authorization, Code Libraries, Counters, Cryptography, Error Handling, Interprocess Communication, File Processing, Logging, Memory Management, Networking, Number Processing, Program Invocation, Protection Mechanism, Session Management, Signals, String Processing, or functional area independent),
- affected resources (it can be CPU, File or Directory, Memory, System Process, other),
- category (can be used to classify attack goals), etc.

It should be noticed that some of these metrics are connected with vulnerability metrics. The CWE metrics are not added to the ontology in Figure 1 yet.

The CAPEC metrics[6] specify and assess attacks. They also combine identifying metrics ("attack pattern identifier", "attack pattern name", "prerequisites", etc.) and evaluation metrics ("confidentiality impact", "integrity impact", "availability impact", "skills required", "typical severity", "likelihood of attack", etc.). In their turn, these metrics are connected with vulnerability metrics and weaknesses metrics.

4 Results and Discussion

The main results of the conducted research are as follows:

- The ontology of security metrics that combines the sources of security data, objects of security assessment process, and security metrics. The essence of proposed ontology is revealing of the relationships from raw data to the answers to security related questions.
- The ontological inference technique for integral security metrics calculation to answer security related questions.

The introduced ontological model is implemented in Protege 5.5.0 using the language OWL (Web Ontology Language) of version 2.0. The figures are implemented in MS Visio and Graphviz (the case study below). The

[6]https://capec.mitre.org/documents/schema/index.html

semantic model currently contains 639 axioms including 418 logical axioms, 221 declarations; 86 classes; 54 object properties. It should be noticed that it is the first version of the ontology. The complete domain ontology will be much broader.

The ontology will allow answering such questions as:

- "What is the security risk for the system?"
- "What is the goal of attack considering the fixed security events?"

And more specific, such as:

- "What is the maximum severity of vulnerabilities in the servers of the infrastructure Internet-segment, that are implemented via network by the attackers with high skills?", etc.

Particularly, possibility to detect cyberattack goal considering fixed security events follows from the conducted analysis of dependencies between different attack categories in CAPEC (represented by the Category type of CAPEC scheme), related events (represented by the indicators type of CAPEC scheme), and their consequences (specified using CAPEC fields Scope and Technical Impact). CAPEC database contains 568 attack patterns. For 307 of them the Scope and Technical Impact fields have values. Possible values for Scope are as follows: Confidentiality; Integrity; Availability; Access Control; Accountability; Authentication; Authorization; Non-Repudiation; Other. Possible values for Technical Impact are as follows: Modify Data; Read Data; Unreliable Execution; Resource Consumption; Execute Unauthorized Commands; Gain Privileges; Bypass Protection Mechanism; Hide Activities; Alter Execution Logic; Other. Thus, there are 37 different combinations of Scope and Technical Impact values (these combinations can determine different attack goals). Absolute number of combinations in the CAPEC database, considering repetitions, is 1290.

We analysed the relations among Scope and Technical Impact values to see if consequences classes can be outlined. The statistics of joint using of possible values of Scope and Technical Impact shown few groups, for example, "Access Control" Scope results in such Technical Impact as "Execute Unauthorized Commands" (rarely); "Gain Privileges"; "Bypass Protection Mechanism"; "Hide Activities" (Figure 3). The Cramer's V considering patterns with empty values of Scope and Technical Impact is 0.5209203125184796. It allows concluding that these metrics correlate but they are not directly dependent, and that an attack goal can be defined by the scope and impact as connected metrics.

Figure 3 The statistics of joint using of Scope (horizontally) and Technical Impact (vertically) possible values.

Besides, to detect attack goals it is also required to consider infrastructure of system under analysis. For this goal in scope of our ontology infrastructure class of concepts is introduced.

Let us demonstrate the application of the developed ontology on a case study, that considers infrastructure of system under analysis to answer the following security question: “What is the maximum severity of vulnerabilities in the servers of the infrastructure Internet-segment, that are implemented via network by the attackers with high skills?”.

The scheme of case study is provided in Figure 4. We can outline from the description the following identifying metric - “target” with the value “servers of Internet-segment” (it is the metric of the “system” object). And we can outline the following evaluation metrics: “AccessVector” with value

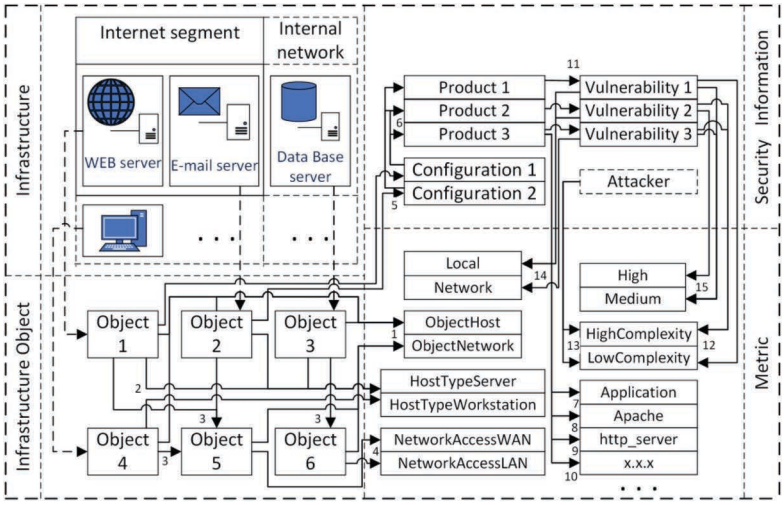

Figure 4 Case study description.

"network" (it is the metric of "vulnerability" object) and "Attacker skill level" with value "High" (it is the metric of "attacker" object).

The provided scheme can be conditionally divided on four parts:

1. an informal representation of the infrastructure (top left part) containing vulnerabilities with Medium CVSS score (that is the maximum severity of vulnerabilities in the servers of test infrastructure);
2. a formal representation of the infrastructure objects as appropriate class (Infrastructure Object – bottom left part);
3. the objects of the class "security information" (top right part) and
4. objects of the class "metric".

In this case study the target infrastructure is represented by internal and external (Internet) network segments, and by two types of hosts (servers and workstations). We translated the objects that comprise the infrastructure into the separate instances of the class InfrastructureObject (1–4). These mappings are represented with dotted arrows. The solid arrows in the figure denote the object properties of the developed ontology in the OWL terms. Their belongingness to the properties is denoted by the number (usually, it is located near the arrow ending).

It should be noticed that two objects (5 and 6) do not represent the infrastructure. These objects characterize the external and internal segments of the computation network, accordingly. The property 1

(hasInfrastructureObjectType) characterizes the infrastructure objects considering the metric of their types InfrastructureObjectType. The property 2 (hasHostType) specifies the host type considering the metric HostType.

The transitive property 3 (connectedWith) should be considered separately. It allows one to specify the interconnection between the infrastructure objects. By this way two networks are outlined – the internal and the external (Internet). The property 4 (hasNetworkAccess) in conjunction with the described above properties 1–3 allows one to determine the objects-servers that have Internet access to answer the question we put in the beginning of this section. Detailed specification of the target infrastructure on the initial stage of the logical inference narrows the search of security information satisfying the set selection criteria. The final object property the domain of which is objects of the infrastructure is the property 5 (hasConfiguration). It specifies hosts configuration. The property 6 (containsProduct) connects objects of hosts configurations with software and hardware products. In its turn, the identifying metrics of the products are ProductType, ProductVendor, ProductName, ProductVersion, etc. The connection with the product instances via these metrics is implemented using the properties 7 (hasProductType), 8 (hasProductVendor), 9 (hasProductName) and 10 (hasProductVerstion), accordingly. The property 11 (containsImplementationOf), introduced in our previous research [45], implements the relationship between the instances of the security information classes: products and vulnerabilities. The properties 12 (hasAccessComplexity), 14 (hasAccessVector) and 15 (hasBaseScoreMetric) represent the relationships of vulnerability instances with the metric classes AccessComplexity, AccessVector and BaseScoreMetric, accordingly. The property 13 (hasAttackerQualification) should be also highlighted. It is equivalent to the metric AccessComplexity of the vulnerability exploited by the attacker. Thus, an answer on the security question in the beginning of this section considering the described model (Figure 2) is the instance of the metric class BaseScoreMetric – Medium (that is the correct answer considering the test infrastructure).

The provided case study serves to clear the relationships between the classes of metric instances, security information and objects of target infrastructure. In real life experiments and further application for the security management number of links and class instances significantly complicated the figure interpretation. Besides, the scheme does not represent inverse object properties and top-level object properties for the same reasons.

Finally, we have compared our ontological model with related ones. The results are provided in Table 1.

Table 1 Qualitative comparison of the ontologies for security management

Ontology\ Characteristic	Domain	Information	Data sources	Advantages	Disadvantages
Ontology for Vulnerability Management [25–27]	Vulnerability assessment	Vulnerabilities	NVD, CVE, CWE	Revealing information on vulnerabilities and limitations of security requirements	Limited domain
Ontology for SIEM system [28]	Security decision support	Information (security and configuration) and operation	SIEM	Allows selecting the security measures	Not completed; limited with SIEM information
Cyber security ontology [30]	SCAP support	Configuration, software, vulnerability, policy, remediation, incident	CCE, CPE, CVE, CWE	Cover a lot of data sources to get complete information security picture	Not completed; upper-level; doesn't detail metrics
Unified Cybersecurity Ontology [31, 32]	Cyber security	Information and communication objects (including network state and information about attackers)	CVE, CWE, CAPEC, CCE	Cover a lot of data sources to get complete information security picture	Manual setup; does not allow integrating information from different sources of the same type; doesn't detail metrics

Semantic access control model [33]	Access control	Subjects, objects and actions (i.e. grant or revoke) of access control	Expert knowledge, scanners	Allows detecting errors in policy	Limited domain
Ontology for decision-making in industrial systems [34]	Security decision support	Assets, vulnerabilities and their severity level, threats and OSI level they affect, security tools	System monitoring tools, expert knowledge	Allows modification on fly	Limited domain (IoT); doesn't detail metrics; does not consider security data sources
Cyber security ontology [36]	Security assessment; Security decision support	Information and communication objects	MITRE sources, Open OWASP and WASC results	Reveals relations between known attacks and security information	Limited with representation of known attack; not dynamical, requires high skills and time costs
The proposed ontology	Cyber security; Security assessment; Security decision support	Information and communication objects, metrics	Open data sources (NVD, CVE, CWE, CAPEC, etc.) and network monitoring tools, SIEM	Allows calculating security metrics for security assessment and countermeasure selection using inference mechanism and integrate cyber security knowledge to answer security questions	Not completed

It shows the advantages of our model, namely, level of detail, application of the inference mechanism to calculate security metrics that represent security state and allows selecting security measures, and integrating the cyber security knowledge to answer security questions.

5 Conclusion

The paper analysed security data sources, features of the data contained in them and their internal and external relations. The conducted analysis is aimed at development novel approaches to processing huge streams of gathered security related data (both static, from open source databases, and dynamic, from security monitoring tools).

As the result the ontology of security metrics and related technique were developed. The essence of the proposed ontology consists in representation of features of security related data and security goals with the set of metrics mapped to them. The essence of the proposed technique is application of relations between security data and security goals to calculate appropriate metrics to answer security related questions.

The paper described the ontology, its concepts and interrelations between them. The idea of security evaluation technique was briefly described. Analysis of security data sources was conducted as well as the case study was specified to demonstrate applicability of the proposed ontology and related technique.

The future work will be devoted to extension of the set of rules for the ontology to include all known open security data sources, the data contained in them and their interrelations. Besides, we plan to include dynamic data to the ontology, such as security events and incidents. The security evaluation technique will be evolved and experiments for different types of systems will be conducted.

Finally, as soon as we complete our ontology we plan to share it with interested experts that have an expertise and experience in real-life to get feedback and to jointly develop it to get useful and applicable in the industry tool.

The reported study was funded partially by RFBR according to the research project № 19-07-01246 and by the budget (the project No. 0073-2019-0002).

References

[1] NIST official website. URL: https://www.nist.gov/ (access date: 01.12.2019).
[2] NVD official website. URL: https://nvd.nist.gov/ (access date: 01.12.2019).
[3] MITRE corporation official website. URL: https://www.mitre.org/ (access date: 01.12.2019).
[4] CWE official website. URL: https://cwe.mitre.org/ (access date: 01.12.2019).
[5] CAPEC official website. URL: https://capec.mitre.org/ (access date: 01.12.2019).
[6] CyBOK official website. URL: https://www.cybok.org/ (access date: 01.12.2019).
[7] I. Kotenko, M. Stepashkin, E. Doynikova, 'Security analysis of information systems taking into account social engineering attacks', Proc. of the 19th International Euromicro Conference on Parallel, Distributed, and Network-Based Processing, 2011, 611–618.
[8] I. Kotenko, E. Doynikova, A. Fedorchenko, A. Chechulin, 'An ontology-based hybrid storage of security information', Information Technology and Control', 18, 3, 2018.
[9] E. Doynikova, I. Kotenko, 'Approach for determination of cyber attack goals based on the ontology of security metrics', Proc. of the IOP Conference Series: Materials Science and Engineering, vol. 450, 'Data protection in automation systems', 2018.
[10] E. Doynikova, A. Fedorchenko, I. Kotenko, 'Ontology of metrics for cyber security assessment', Proc. of the 14th International Conference on Availability, Reliability and Security (ARES 2019), August 26–29, 2019, Canterbury, United Kingdom, ACM, New York, NY, USA, 8 pages, 2019, https://doi.org/10.1145/3339252.3341496.
[11] IBM official website. IBM QRadar SIEM. URL: https://www.ibm.com/us-en/marketplace/ibm-qradar-siem (access date: 17.09.2019).
[12] Micro Focus official website. ArcSight Enterprise Security Manager (ESM). URL: https://www.microfocus.com/ru-ru/products/siem-security-information-event-management/overview (access date: 17.09.2019).
[13] Splunk official website. Splunk Enterprise Security. URL: https://www.splunk.com/en_us/software/enterprise-security.html (access date: 17.09.2019).

[14] LogRhythm official website. LogRhythm NextGen SIEM Platform. URL: https://logrhythm.com/products/nextgen-siem-platform/ (access date: 17.09.2019).
[15] CVE official website. URL: https://cve.mitre.org/ (access date: 01.12.2019).
[16] OSVDB official website. URL: https://blog.osvdb.org (access date: 01.12.2019).
[17] US-Cert official website. URL: https://www.kb.cert.org/vuls/ (access date: 01.12.2019).
[18] SecurityFocus project official website. URL: http://securityfocus.com (access date: 01.12.2019).
[19] IBM X-Force Exchange official website. URL: https://exchange.xforce.ibmcloud.com/ (access date: 01.12.2019).
[20] Official website. URL: https://www.exploit-db.com (access date: 01.12.2019).
[21] Official website. URL: https://www.metasploit.com (access date: 01.12.2019).
[22] M. Horridge, 'A practical guide to building OWL ontologies using Protege 4 and CO-ODE tools', The University Of Manchester, 2011.
[23] Protege User Documentation, retrieved May 20, 2019 from https://protegewiki.stanford.edu/wiki/Main_Page.
[24] Web Ontology Language Overview, retrieved May 20, 2019 from https://www.w3.org/TR/owl-features.
[25] G. Elahi, E. Yu, N. Zannone, 'A modeling ontology for integrating vulnerabilities into security requirements conceptual foundations', Lecture Notes in Computer Science 5829, 99–114, 2009.
[26] M. Guo, J. A. Wang, 'An ontology-based approach to model common vulnerabilities and exposures in information security', Proc. of the 2009 ASEE Southeast Section Conference, 2009.
[27] J. A. Wang, M. Guo, 'Security data mining in an ontology for vulnerability management', Proc. of the International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing, Shanghai, 2009, 597–603.
[28] G. G. Granadillo, Y. B. Mustapha, N. Hachem, H. Debar, 'An ontology-based model for SIEM environments', Global Security, Safety and Sustainability & eDemocracy, Springer Berlin Heidelberg, 2012, DOI: 10.1007/978-3-642-334481_21.

[29] I. Kotenko, O. Polubelova, I. Saenko, E. Doynikova, 'The ontology of metrics for security evaluation and decision support in SIEM systems', Proc. of the ARES 2013, 2013.
[30] M. C. Parmelee, 'Toward an ontology architecture for cyber-security standards', Proc. of the 2010 Semantic Technology for Intelligence, Defense, and Security. Fairfax, 116–123, 2010.
[31] Z. Syed, A. Padia, T. Finin, L. Mathews, A. Joshi, 'UCO: a Unified Cybersecurity Ontology', Proc. of the AAAI Workshop on Artificial Intelligence for Cyber Security, Phoenix, Arizona, USA, 195–202, 2016.
[32] Unified Cybersecurity Ontology, retrieved May 20, 2019 from https://github.com/Ebiquity/Unified-Cybersecurity-Ontology.
[33] S. Javanmardi, M. Amini, R. Jalili and Y. Ganjisaffar, 'SBAC: a Semantic-Based Access Control model', 2006.
[34] B. Mozzaquatro, R. Melo, C. Agostinho, R. Jardim-Goncalves, 'An ontology-based security framework for decision-making in industrial systems', Proc. of the 4th International Conference on Model-Driven Engineering and Software Development, 779–788, 2016.
[35] C. Marinica, 'Association Rule Interactive Post-processing using Rule Schemas and Ontologies – ARIPSO', 2010.
[36] A. Aviad, K. Węcel, W. Abramowicz, 'The semantic approach to cyber security. Towards ontology based body of knowledge', Proc. of the 14th European Conference on Cyber Warfare and Security, 328–336, 2015.
[37] I. Kotenko, A. Fedorchenko, A. Chechulin, 'Integrated repository of security information for network security evaluation', Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 6, 41–57, 2015.
[38] CWE official website. CWE schema documentation. URL: https://cwe.mitre.org/documents/schema/index.html (access date: 01.12.2019).
[39] CAPEC Scheme Documentation. Official web-site of MITRE corporation. Access: https://capec.mitre.org/documents/schema/index.html (access date: 01.12.2019).
[40] CAPEC Scheme. Official web-site of MITRE corporation. Access: https://capec.mitre.org/data/xsd/ap_schema_v3.1.xsd (access date: 01.12.2019).
[41] E. Doynikova, I. Kotenko, 'CVSS-based probabilistic risk assessment for cyber situational awareness and countermeasure selection', Proc. of the 25th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP 2017), IEEE, St. Petersburg, Russia, 2017, DOI: 10.1109/PDP.2017.44.

[42] P. M. Mell, K. A. Scarfone, S. Romanosky, 'A complete guide to the common vulnerability scoring system version 2.0', FIRST Forum Incident Response Security Teams, 2007.
[43] D. Waltermire, P. Cichonski, K. Scarfone, 'Common platform enumeration: applicability language specification version 2.3', NISTIR 7698, 2011.
[44] FIRST, 'Common vulnerability scoring system v3.0: specification document', Forum Incid. Response Secur. Teams, 2015, DOI: https://doi.org/10.1109/msp.2006.145.
[45] E. Doynikova, A. Chechulin, I. Kotenko, 'Analytical attack modeling and security assessment based on the common vulnerability scoring system', Proc. of the FRUCT 2017, 2017.

Biographies

Elena Doynikova received her PhD in St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS) in 2017. In 2015 she was awarded the medal of the Russian Academy of Science in area of computer science, computer engineering and automation. Currently she is a senior researcher of computer security problems laboratory, SPIIRAS. Research interests: information systems security, risk analysis and security decision support methods, security metrics, information security risk management. She is the author of more than 50 publications and has participated in several projects devoted to information systems security research.

Andrey Fedorchenko is a junior researcher of computer security problems laboratory, SPIIRAS. Research interests: computer network security, intelligent data analysis, intrusion detection, malware. He is the author of more than 40 publications and has participated in several projects devoted to information systems security research.

Igor Kotenko graduated with honors from St. Petersburg Academy of Space Engineering and St. Petersburg Signal Academy. He obtained the Ph.D. degree in 1990 and the National degree of Doctor of Engineering Science in 1999. He is Professor of computer science and Head of the Laboratory of Computer Security Problems of St. Petersburg Institute for Informatics and Automation. He is the author of more than 500 refereed publications. He has a high experience in the research on computer network security and participated in several projects on developing new security technologies. For example, he was a project leader in the research projects from the US Air Force research department, via its EOARD (European Office of Aerospace Research and Development) branch, EU FP7 and FP6 Projects, HP, Intel, F-Secure, etc. The research results of Igor Kotenko were tested and implemented in more than fifty Russian research and development projects.