Ontology-based Big Data Approach to Automated Penetration Testing of Large-scale Heterogeneous Systems

Taiana Stepanova
Peter the Great St. Petersburg Polytechnic University
Russia, St. Petersburg, Polytechnicheskaya St. 29
+7 (812) 552-76-32
tatiana.stepanova@ibks.ftk.spbstu.ru

Alexander Pechenkin
Peter the Great St. Petersburg Polytechnic University
Russia, St. Petersburg, Polytechnicheskaya St. 29
+7 (812) 552-76-32
alexander.pechenkin@ibks.ftk.spbstu.ru

Daria Lavrova
Peter the Great St. Petersburg Polytechnic University
Russia, St. Petersburg, Polytechnicheskaya St. 29
+7 (812) 552-76-32
daria.lavrova@ibks.icc.spbstu.ru

ABSTRACT
Global corporations and government organizations are nowadays represented in cyberspace in the form of numerous large-scale heterogeneous information systems, which implement corresponding business, technological and other types of processes. This extends the set of security analysis tasks, stated for these infrastructures, and tangles already existing tasks. This paper addresses the challenge of increasing penetration testing automation level through the adoption of semi-automatic knowledge extraction from the huge amounts of heterogeneous regularly updated data. The proposed solution is based on the novel penetration testing ontology, which gives a holistic view on the results of security analysis. Designed ontology is evaluated within the penetration testing framework prototype and binds together the conceptual (process) abstraction level, addressed by security experts, and technical abstraction level, employed in modern security analysis tools and methods.

Categories and Subject Descriptors
H.m [Information systems]: Miscellaneous

General Terms
Design, Experimentation, Security, Theory.

Keywords
Big Data, ontology, penetration testing, large-scale systems.

1. INTRODUCTION
Importance of digital infrastructure for modern companies in all industries (not only IT) rapidly increases. This extends the set of security analysis tasks, stated for these infrastructures, and tangles already existing tasks. Global corporations and government organizations are represented in cyberspace in the form of numerous large-scale heterogeneous information systems (IS) [1], which implement corresponding business, technological and other types of processes (the term "process" is used at the conceptual level of abstraction) according to company profile. It is widely discussed, which security assessment methods are the best choice for the IS still penetration tests (pentests) are extensively used to provide for advanced security analysis. When penetration tests are conducted to determine actual security level of these systems, threat models are usually described in terms of appropriate processes and then are manually translated into the set of attack vectors, aimed at violating particular process, implemented in the computer system [2]. Attack vector is usually a sequence of several steps triggering specific vulnerabilities, discovered in IS components by a security expert. Here the term "vulnerability" is understood in the most broadest sense as a security flaw, which arises from computer system design, implementation, maintenance and operation, including possible exposure to social engineering, physical access to servers, etc.

SIN '15, September 08 - 10, 2015, Sochi, Russian Federation
© 2015 ACM. ISBN 978-1-4503-3453-2/15/09...$15.00
DOI: http://dx.doi.org/10.1145/2799979.2799995

Requirements for solutions, designed for IT security analysis, are formed with respect to several determinant influence factors, triggered by overall computerization and internetization:

— Huge number of multi-level components in modern IS. At the application level, for instance, there are dozens of servers and services, which, in turn, are equipped with hundreds of potentially vulnerable program modules (often specific for application area).

— All cross-component relationships (both intra- and inter-level) should be considered during the process of security analysis.

— Inability to proportionally increase the number of security experts, conducting penetration test; and/or time period, allocated for security analysis.

Here and after we are speaking about security analysis of large-scale heterogeneous highly distributed information systems, because requirements for penetration tests of small (and, moreover, isolated) systems stay more or less constant in time.

Factors, listed above, help to define requirements for security analysis process:

1) Ability to bring together, store and process massive amounts of poly-structured data. This turns in the need for implementation of Big Data approaches in order to reach acceptable completeness level of the dataset being analyzed. Acceptability here is determined by the probability of qualified, correct conclusion, to which security expert comes as a result of analyzing amounts of data, mentioned above (data, associated with the target system and its infrastructure). It is hardly possible to formally calculate this probability, so informal estimation, bolstered by references to best practices [3-4], is considered to be sufficient.

2) High level of automation of the analysis process in order to leverage an acceptable functional completeness. This means that sufficient number of attack vectors were checked for feasibility and acceptable number of IS functions were tested.

Pentest automation is a vital aid for reducing all the amount of collected data (with the minimal loss of valuable information) to that point, where security experts have enough physical abilities to analyze this reduced dataset as a whole. Since original data sources may contain terabytes of raw data, mentioned reduced (adapted) dataset have to accumulate mostly processed data (difference between "data" and "information" in terms of DIKW pyramid [5] is not essential in this context), i.e. knowledge. Knowledge, that could be extracted automatically, in combination with knowledge, occasionally found by a security expert and added to this knowledge base.

Thus, this paper addresses the challenge of increasing penetration testing automation intensity through the adoption of semi-automatic knowledge extraction from the huge amounts of heterogeneous regularly updated data.

Semantic technologies are crucial for knowledge extraction, processing and storage. The suite of methods developed in the Semantic Web [6], such as ontologies, semantic annotation, Linked Data [7] and semantic Web services [8], are nowadays implemented as principal solutions for different purposes in healthcare, education, finance and other industry sectors. The next section contains a review of the current state of the art for our research field – existing approaches to automation of knowledge processing and their potential for security analysis purposes.

2. STATE OF THE ART AND RELATED WORKS

Known tools used for penetration testing, such as vulnerability and port scanners (Nessus, Nexpose, nmap, xSpider, MaxPatrol, OWASP ZAP, Acunetix, Metasploit and others), mostly provide vulnerability-centered information, not prepared for implementation of knowledge extraction and other semantic-based techniques, that could enhance expert security analysis. Ontology-based approaches said to be nearly the most efficient and promising part of the expanding theoretical framework, associated with the area of semantic analysis. Therefore, the overwhelming majority of existing tools and methods [9-13], aimed at knowledge processing, utilize ontology-based analysis techniques. The term "ontology" stands for a formal naming and definition of the types, properties and interrelationships of the entities that really or fundamentally exist for a particular domain of discourse [14]. Common ontology classification is based on the abstraction level of concepts under consideration – i.e. based on the scope of the ontology, or on the domain granularity [15] (figure 1).

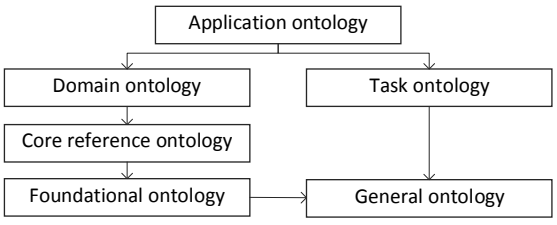

Figure 1. Common ontology classification

Local (or application) ontologies are built for that areas, where it is not supposed to share the experience. Such ontologies represent point of view for the single user or developer. In [16] this ontology class is positioned as retrieved as a result of merging two other classes: domain ontology and task ontology. Task ontologies capture particular problem-solving processes. Domain ontologies describe knowledge about the target object and corresponding "world" of a certain task. Core reference ontologies describe the standard, used by a community of people, and are often resulted from several domain ontologies. General ontologies describe concepts, not specific to any particular area of expertise. Foundational (or top-level or upper level ontologies) are the most general ontologies, suitable for a wide range of application areas, and are often called meta-models or conceptual schemes [17]. For each consistent ontology there is corresponding meta-ontology. Ontology for penetration testing of large-scale heterogeneous systems (that is to be developed within the solution of problems, assigned above) could be attributed to the first class – application ontology (with the only clarification that it is intended for the use inside the entire community of security experts).

Problem being solved lies at the intersection of two adjacent areas:

1) Ontology-based big data analysis.

2) Ontology-based solutions in computer security.

In both directions lots of research projects are guided, but in the area of big data analysis ability of processing truly *Big* Data is quite often declared only nominally. This is fair mostly for proprietary products, for which it is essential to follow the most recent technological trends (and Big Data, of course, is one of them) as it determines their financial success. The major Big Data characteristics are well-known so called five "V" (volume – too large to be processed by conventional data management techniques, velocity – impermanence, high updating speed, verbosity (variety) – have too diverse structure, veracity – high quality and understandability, value (for business processes)) [18]. Thus, ontology-based big data analysis should take into consideration all these characteristics and, moreover, be able to store and process three data types: structured, semi-structured and unstructured. This means, that the whole set of data sources could be divided into three subsets (typical example of structured data source – relational databases, semi-structured – XML files, unstructured – plaintext and media files), and these data sources are fundamentally not mergeable without the loss of valuable information. Most of the Big Data-associated methods nowadays don't fully support Big Data processing, ignoring any of the listed characteristics. Most often unstructured data processing is not taken into account, and primary importance is assigned to frequency of data updates and data heterogeneity [19].

Ontology-based approaches, utilized for computer security purposes, are employed in several fields:

— SIEM-systems [20-21], attack and intrusion detection systems [22-23]. In [24-25] ontologies are used to organize federated access to database, containing results of cyberspace monitoring in order to detect targeted threats.

— Risk management in security incident detection tasks [26-27].

— Vulnerability discovery and analysis [28-29].

— Computer security models. For instance, in [30] ontology-based approach is used to design the test environment that efficiently replaces the real system during the experimental security testing.

In addition to this, several existing research papers [31-32] have yielded the expanding needs for absent fundamental holistic cybersecurity ontology.

Thus, one can notice, that both reviewed approaches to knowledge extraction and processing, and mentioned penetration testing solutions represent the field of expertise in technical-level terms. In other words, vulnerabilities and impact of the appropriate exploits are in the focus. At the same time, the expert analysis (in particular, the part involving simulation of multistage attacks and APTs) is carried out at the business process level. The duty of establishing correspondence between technical and business process level also falls on the shoulders of an expert. Moreover, this problem of establishing the correspondence is privately solved once again for each particular system under test. This paper intends to contribute narrowing the gap between information that security expert is given by his professional tools, and knowledge, he needs in order to make a conclusion about system security level.

3. REVISED PENETRATION TESTING METHODOLOGY

In this section we generalize our expertize in security analysis of large-scale heterogeneous systems, best practices and existing methodologies [3-4, 33-34] to describe the main penetration testing steps (figure 2). This description forms the revised penetration testing methodology that will serve as a basis for subsequent formalization in terms of semantic analysis.

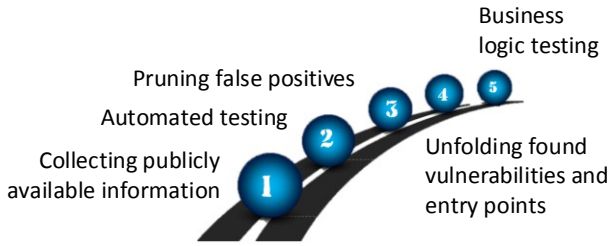

Figure 2. Generalized penetration testing scheme

Basic conceptual operations, unfolding the above steps, are illustrated in figure 3.

Arrows indicate basic relationship types: black solid arrow correspond to data flow from the output of the previous pentest stage to the input of the next stage; green dotted arrow designates a required sequence of steps (each next step requires previous to be completed).

The foundations for remote revealing of structure, architecture and infrastructure of the object under test are results of the publicly available data analysis and active scan results. Publicly available information is provided through the use of pentest tools and specialized sources, including, but not limited to:

— Search engines (both general purpose engines such as Google, Bing, Yandex, Baidu, DuckDuckGo, ixquick, and special purpose ones: Shodan, PunkSpider), as well as databases that contain lists of Google/Bing/etc. dorks, that could be used to find usernames, passwords, e-mail list, password hashes and other important information (Google Hacking Database, Bing Hacking Database).

— Web-sites that contain statistics and registration information (whois, domaintools, netcraft and others).

— Social networks and forums.

— DNS name finders (dnsmap, dnswalker).

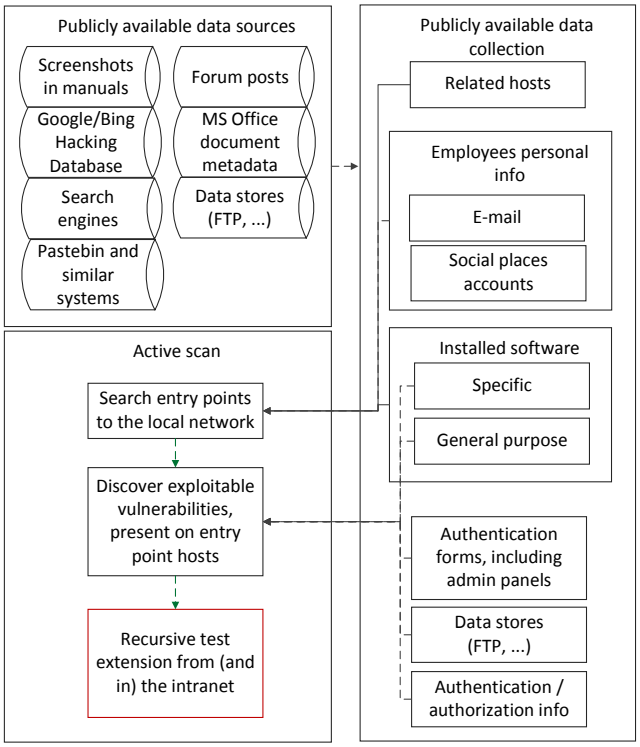

Figure 3. Basic aspects of the revised penetration testing methodology

The process of penetration testing and analysis of gathered knowledge appears to be fragmented across several tools and workarounds, and there exists no well accepted framework for performing common security analysis tasks such as exploring implicit and explicit entity relationships. This lack of an adequate and seamless tool chain potentially hinders the broad uptake of automated comprehensive business-oriented penetration testing and security analysis. In the next section we start to address this situation by translating the described methodology to ontology concepts and relations.

4. PENETRATION TESTING ONTOLOGY

Process of penetration testing ontology development was based on common well-known standards, taxonomies and methodologies (while there is no universal one) [35-36] and intensive discussions we held with cybersecurity operators. Process of ontology creation included the following steps:

1) Form the thesaurus of the research domain and appropriate tasks and subtasks. This dictionary incorporates open community-driven security-connected standards and taxonomies (ARF, CAPEC, CCE, CCSS, CEE, CPE, CRF, CVE, CVRF, CVSS, CWE, CWSS, CybOX, IODEF, MAEC, MMDEF, OCIL, OVAL, SWID, WS-Agreement, XACML, XCCDF) [37] and several general purpose ontologies such as FOAF [38].

2) Identify concepts. The whole set of concepts could be divided into four categories [39]: entities, actions, relationships and attributes. In addition to this classification the set of concepts for the proposed penetration testing ontology could be divided into three other subsets: subset, representing the process of security analysis; subset, representing the technical description of the system under test; and subset, representing "logical" or "conceptual" description of the analyzed system (at the business

process level). Further described ontology views are based on the listed subsets.

3) Build hierarchy of concepts.

4) Identify non-hierarchical relations.

5) Develop rules.

6) Populate ontology by discovering new instances of concepts and relations.

7) Extend hierarchy of concepts.

Formally, an ontology is a triple $\{X, R, F\}$ [40], where $X = \{x_i\}$ is a set of concepts, $R = \{r_i\}$ is a set of relations between concepts and $F = \{f_i\}$ is interpretation function both for concepts x_i and relationships r_i. This model is quite general in nature, while in practice more accurate models are used for ontology authoring. For instance, widely used Protégé Frames model, developed for implementation in Protégé ontology editor and based on the frame representation and OKBC (Open Knowledge Base Connectivity) protocol. Protégé, a free open-source ontology builder from Stanford University, is the tool, employed to build the proposed penetration testing ontology.

However, in essence, all modern models oriented for real-world implementation are either equivalent to the mentioned above general $\{X, R, F\}$-model, or exist as its' limited views. Therefore, they could be always represented in terms of this general ontology model, for example, as shown in figure 4 for Protégé Frames model.

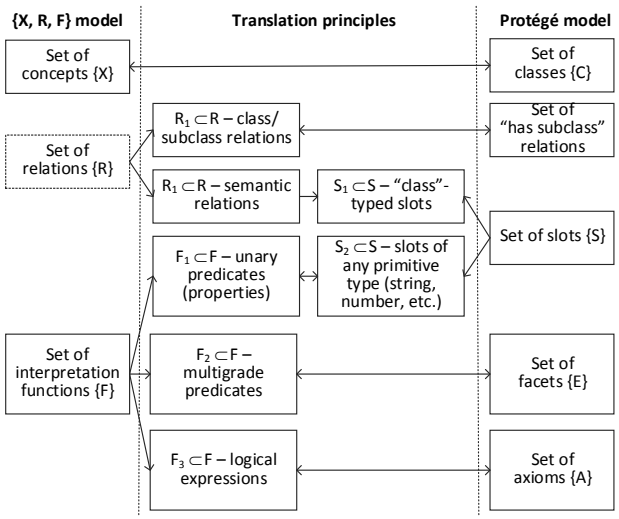

Figure 4. Formal association between general $\{X, R, F\}$-model and Protégé Frames model

According to this, for ontology authoring any model could be utilized. Our ontology was authored on the basis of general model (with the help of authoring approach, based on ontology views) and then translated into OWL and edited in Protégé. OWL is a product of long evolution in knowledge representation techniques, which is described in [41]. General ontology views are more suitable for understandable visualization of the developed ontology in the format of published paper. All presented general views correspond to the more detailed created OWL ontology, containing 519 concepts.

Ontology views are used, because the proposed ontology is essentially multidimensional, and simultaneous visual representation of all the views will greatly complicate its perception. The term "view" has been used informally, in the sense of a perspective or a viewpoint that the end user will take of the ontology. The definition of an ontology view is less well established, than database or XML view. It has been claimed that an ontology view should result in a new (smaller) ontology derived from, but independent of, the source ontology. We are holding the alternative opinion, proposed in [42], that the ontology view might be the set of terms and definitions within a certain radius of a selected term, a set which does not itself constitute an ontology but a connected subgraph, and represent $\{X, R\}$-part of the general ontology model. Additional motivation for computing ontology views (except improved human comprehension and discussion) is similar to that for database views: extracting a smaller subgraph of terms from larger structure should improve the efficiency of automated querying and reasoning.

At first, we describe the scheme of relationships between concepts, central to our ontology (figure 5), and then continue with the ontology view description for each central concept. This approach to ontology visualization corresponds to used definition-driven approach to ontology creation [43].

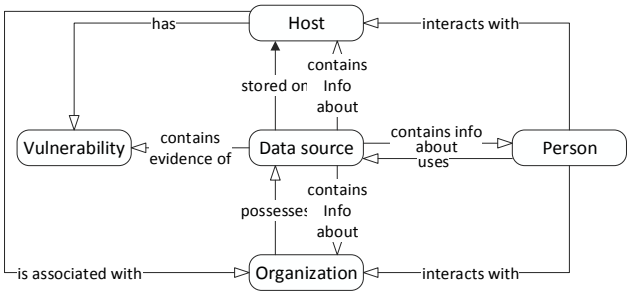

Figure 5. Penetration testing ontology central concepts

Concepts, central for the holistic view of penetration testing process, are interconnected as follows. During each pentest stage **hosts**, associated with the **organization** under test, could be exposed. The major elements, that allow to estimate the security level of the overall infrastructure, are exploitable **vulnerabilities,** present on these hosts. Vulnerabilities could be discovered either due to active scan results, or based on information, contained in **data sources** (configuration files, documents, manuals, e-mail inboxes and outboxes, contact lists, etc.), processed in organization. Found data sources could also comprise information about other hosts (for example, such data sources as network traffic dumps or RDP connection lists). The last central concept is **person** (human) and his virtual identities – accounts, created by or for a particular person in various virtual communities and systems. Person could interact with hosts through his virtual identities – for instance, as an admin user of some server.

Each relation, as well as each concept, is a part of a more detailed classification. For example, *association* between host and organization can be "host (IP) *is registered* for the organization", "host (IP) *serves* organization" – for example, in case of a complex indirect relation, when host is not registered for this organization, but serves as a database for another host, which is registered for this organization, etc.

Notation used for ontology view diagrams is shown in figure 6.

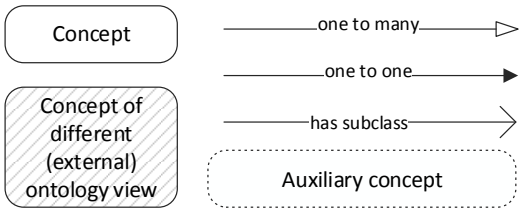

Figure 6. Ontology view notation

Hereafter schematic ontology views are presented in the form of semantic graphs (connected triples {subject vertex – predicate edge – object vertex}), which reflect the essence of an appropriate view. Fully detailed ontology views were developed in OWL, but are too large to match the format of the article. Also, to avoid unnecessary complication of the presented ontology schemes, inverse relationships are not pictured (for example, the host *has* a vulnerability – the vulnerability *is present* on the host). In the diagram with relationships of the central concepts all edges have corresponding inverse edges that are omitted. Also the majority of "has subclass" relations were omitted on the provided schemes – they enhance existing classification approaches and ontologies and will be described in future papers. Data source-centric view consists of data sources classification and corresponding classification of other entities relations to the data sources, and is not presented in this paper.

Figure 8 presents the most essential aspects of the proposed ontology. Host-centric view describes information about what is installed on the host (both software and hardware) and network infrastructure. Not only trivial relations within LAN are taken into consideration, but also interconnections, discovered in RDP connection files, ssh and telnet connection logs, etc. With the help of used knowledge database of service fingerprints (for example, for different industrial services, implemented through SCADA systems) installed software could be identified based on the port scan results. Person-centric (or virtual identity-centric) view shows, which people are involved in the tasks of tested organization and what information is available about these people and principles of their interaction with the virtual infrastructure. Vulnerability-centric view allows to analyze available technical information about security flaws and estimate system security metrics. Organization-centric view binds processes and use cases, typical for the organization under test, with gathered technical and social information, allowing to describe attack vectors in (business) process terms. Here we consider, that each organization implement number of processes, which are represented in computer infrastructure as various interconnected software services (for instance, set of services, employed on corresponding services, for the supply chain process).

5. ONTOLOGY CAPABILITIES

Proposed penetration testing ontology has been used within the prototype of software tool that allows to partially automate the analysis of data, retrieved from passive and active security scans and available data sources. The general structure and usage algorithm of the developed tool is shown in figure 7. Along with the collected data, stored in the experts' internal data store, our tool addresses external data sources, associated with the information system under test. In particular, it is useful to detect hidden patterns of historical data, related to the company and to identify behavior profiles of the employees. The need for access to external sources has required the use of an OBDA (ontology-based data access) module, which was implemented on the basis of OnTop [44] – widely used Protégé plugin. OBDA is a paradigm of accessing data through a conceptual layer [45]. The terms in the conceptual layer are mapped to the data layer using special mappings that associate to each element of the conceptual layer a query over the data sources (for example, structured data stored in relational databases – complex SQL query). Formally, an OBDA module is a triple $\{T, S, M\}$, where T is the intentional ontology level (called TBox in OWL), S is the original data source and M is a setting of mapping assertions $\varphi(x) \leftarrow \phi(x)$. Here $\varphi(x)$ is a query over S and $\phi(x)$ is a query over T. The standard method for data access through an ontology is to ingest the data into an ontological database, where the data elements are encoded together with their extant relationships. This does not work in a Big Data scenario, since ontological databases do not have the horizontal scalability needed to handle data at high volume, velocity and diversity.

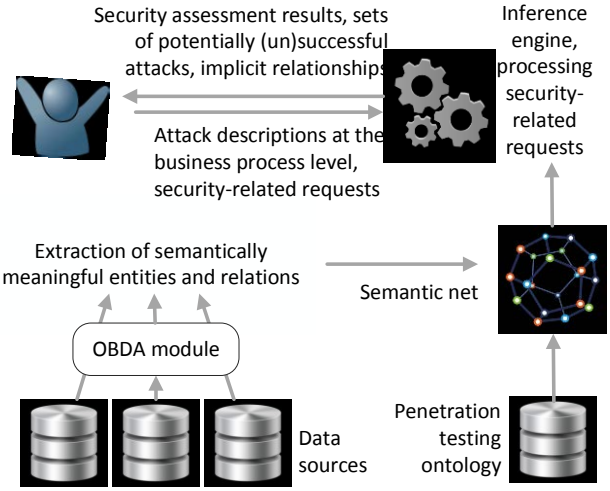

Figure 7. Basic workflow scheme of automated penetration testing with proposed prototype tool

Mappings and inference rules at the current development stage are manually formed by a security expert, but, of course, are reusable across results of different penetration tests. After extraction of semantically meaningful entities and relations form the data sources, the semantic net is automatically built based on the developed penetration testing ontology and corresponding mappings (in OWL terms this semantic net is called ABox, "an assertion box"). Functional features of the inference engine (based on Quest component of OnTop) allows to eliminate tedious manual work with the help of following functions:

— Automatically perform some penetration actions that can result in ABox population. Let's consider the following example: during the active scan phase some host was discovered and automatically (by our tool) was assigned "mail server" role (due to discovered open ports). Also let us know the domain name of this server. If the semantic net contains the certain chain of action-concepts, our tool will enable next security analysis subprocess:

- For the found e-mails either add new virtual identity to the knowledge base or link mentioned mail server will the already existing virtual identity.
- Search the Internet for known e-mail addresses on this domain (discover associated virtual identities).

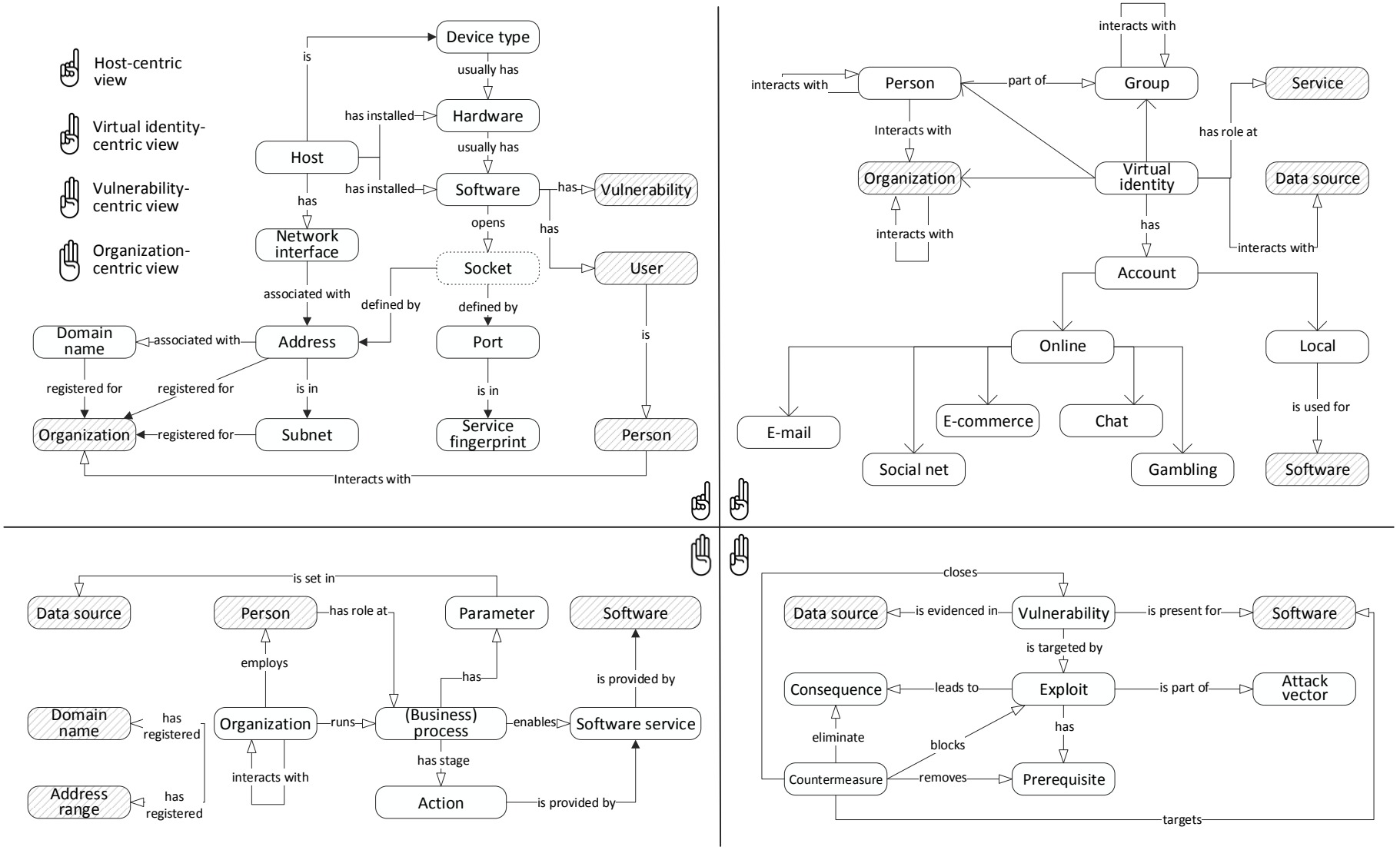

Figure 8. Penetration testing ontology views

- As there are list of authentication/authorization records, associated with each virtual identity, check whether usernames and passwords, already known for this identity, suit to the discovered mailbox.

- If virtual identity is assigned a role in some process (business, technological, etc.), its' influence sphere in the information system could be revealed. Then this knowledge is used to form social engineering attack vector.

— Automatically test feasibility of potential attack vectors (with the help of semantic graph). Attack vectors, formulated at the conceptual level (process level), are strongly dependent of the process, implemented in the information system. Therefore, attack vectors are built manually by a security expert. Objective of the pentest system is to estimate the success probability of the specified attack. This estimation is based on the knowledge about installed software, discovered vulnerabilities, possible exploit chains, etc.

— Automatically identify explicit and implicit relationships between people, groups of people, organizations and technical entities. For instance, during the scan of the network, belonging to a certain organization X, in the registered for this organization address range (determined by the whois call), there was found a business card website of another organization Y. Y is the vendor of software, specific precompiled for each customer. For this software there are available manuals, providing typical configuration info. This allowed our tool to find this software installed on the target network (specific port numbers were retrieved from the manual) and determine, that organizations X and Y are connected with a "customer-supplier" relation. Moreover, employees of both organizations could be found (on LinkedIn social network, for example) along with their e-mail addresses. This chain could be continued by checking specialized forums for essential information: there could be found discussion between employees of the Y company, which are engaged in the product maintenance services for the company X. This again gives a good basis both for social engineering attack vector and search direction of indirect backdoor entry points to the local network of company X.

Even at this early development stage the pentest tool prototype has shown, that designed penetration testing ontology helps to mitigate mentioned challenges regarding security testing. It allows to process requests of the security experts (in form of SPARQL and SWRL queries), formulated in terms of the conceptual (process) level promptly and flexibly. Moreover, implementation of this ontology allows to accommodate the needs for security analysis knowledge visualization in the form, providing the holistic view of system security state, including both implicit and explicit facts.

6. CONCLUSION AND FUTURE WORKS

Solving the challenge of increasing penetration testing automation intensity through the adoption of semi-automatic knowledge extraction from the huge amounts of heterogeneous regularly updated data requires an integrated approach, that we propose to base on the designed penetration testing ontology. This ontology presents a holistic view on the results of penetration testing and binds together the conceptual (process) abstraction level, addressed by security experts, and technical abstraction level, employed in modern security analysis tools and methods. For the penetration testing ontology we have introduced the ontology views that support the ubiquitous and collaborative utilization of this ontology.

Designed ontology has been evaluated within developed penetration testing framework prototype, which allows to achieve automated query answering for the queries, formed by a security expert in terms of business, technological or another process, implemented in the information system under test. However, there remain several stages of security analysis process, which still need to be automated. In terms of the semantic analysis these issues could be described as follows:

— Clearly visualize, based on proposed ontology views, extant implicit and explicit knowledge about the system under test.

— Improve rule-based inference engine functionality for efficient answering under high inconsistency.

— Automate mapping extraction both for mappings between ontology and data sources and for mappings between different ontologies. The latter will facilitate the simultaneous use of multiple ontologies.

Adding appropriate solutions for the tasks listed above to developed penetration testing ontology-based framework is considered to provide tremendous value and is subject of our future works.

7. ACKNOWLEDGMENTS

Project is financially supported by the Ministry of Education and Science of the Russian Federation, Federal Program "Research and Development in Priority Areas of Scientific and Technological Sphere in Russia for 2014-2020" (Contract No.14.575.21.0100; November 14, 2014).

8. REFERENCES

[1] Weske, M. Concepts, Languages, Architectures (Vol. 14) . Berlin: Springer-Verlag. New York, Inc., Secaucus, NJ, United States, 2007.

[2] Ju An Wang and Minzhe Guo. OVM: An ontology for vulnerability management. In Proceedings of the CSIIRW'09, pages 34:1–34:4, New York, NY, USA, 2009.

[3] Commercially Available Penetration Testing Best Practice Guide, CPNI, available at: http://www.cpni.gov.uk/Documents/Publications/2006/2006030-GPG_Penetration_testing.pdf

[4] OWASP Testing Guide, available at: https://www.owasp.org/images/5/52/OWASP_Testing_Guide_v4.pdf

[5] Sharma, N. The DIKW origin, available at: http://www-personal.si.umich.edu/~nsharma/dikw_origin.htm, 2004.

[6] T. Berners-Lee, J. Hendler, O. Lassila, , "The semantic web", Scientific American, no. 284, pp. 35-43, 2001.

[7] T. Berners-Lee, "Linked data", available at: http://www.w3.org/DesignIssues/LinkedData.html

[8] A. McIlraith, T. C. Son, H. Zeng, "Semantic Web Services," IEEE Intelligent Systems, vol. 16, pp. 46–53, 2001.

[9] Daiyi Lia et al. An ontology-based knowledge representation and implement method for crop cultivation standard. Mathematical and Computer Modelling V.58, 2013, 466–473.

[10] Yuh-Jen Chen, Development of a method for ontology-based empirical knowledge representation and reasoning. Decision Support Systems, Volume 50, Issue 1, December 2010, Pages 1–20.

[11] Jiangning Wu. A Framework for Ontology-Based Knowledge Management System, available at http://www.iiasa.ac.at/~marek/ftppub/Pubs/csm05/wu.pdf .

[12] Rodriguez-Muro, M., Kontchakov, R., Zakharyaschev, M.: Ontology-based data access: Ontop of databases. In: Proc. of the 12th Int. Semantic Web Conf. (ISWC 2013). vol. 8218, pp. 558–573. Springer (2013).

[13] Hari Rajagopal, JENA: A Java API for Ontology Management. Colorado Software Summit, October 23-28, 2005.

[14] Gruber, T. R. 1995. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. International Journal of Human and Computer Studies, 43(5/6): 907-928.

[15] Roussey Catherine, Pinet François, Kang Myoung-Ah, Corcho Oscar. An Introduction to Ontologies and Ontology Engineering. Chapter in: Use of Ontologies to Support Information Interoperability, 2010, Springer, p. 9-38.

[16] Fonseca, F., Egenhofer, M., Davis, C., Borges, K.: Ontologies and knowledge sharing in Urban GIS. Comput. Environ. Urban. Syst. 24(3), 232–251 (2000).

[17] Fonseca, F., Davis, C., Camara, G.: Bridging ontologies and conceptual schemas in geographic applications development. Geoinformatica 7(4), 355–378 (2003)

[18] De Mauro, Andrea; Greco, Marco; Grimaldi, Michele (2015). "What is big data? A consensual definition and a review of key research topics". AIP Conference Proceedings 1644: 97–104.

[19] McAfee. SIEM: Keeping Pace with Big Security Data, available at http://www.mcafee.com/ca/resources/reports/rp-siem-keeping-pace-big-security-data.pdf.

[20] Kotenko, I. & Novikova, E., 2013. Analytical Visualization Techniques for Security Information and Event Management, 2013, 21st Euromicro International Conference, pp. 519-525.

[21] Blake Bryant, 2014, A Method for Implementing Intention-Based Attack Ontologies with SIEM Software. FishNet.

[22] Palo Alto Networks® and Splunk: Combining Next-generation Solutions to Defeat Advanced Threats, 2013.

[23] Jansse, T., Grady, N., Big Data for Combating Cyber Attacks, Semantic Technology for Intelligence, Defense and Security (STIDS 2013).

[24] Michael Atighetchi et al. Federated Access to Cyber Observables for Detection of Targeted Attacks, Military Communications Conference (MILCOM 2014), Baltimore, MD, October 6 - 8, 2014.

[25] Farah Layouni, Yann Pollet. An Ontology-Based Architecture for Federated Identity Management. AINA '09 Proceedings of the 2009 International Conference on Advanced Information Networking and Applications, pages 162-166.

[26] M. Marques et al., An Ontological Approach to Mitigate Risk in Web Applications. In the Proceedings of SBSeg 2014.

[27] F.-H. Liu et al., Constructing Enterprise Information Network Security Risk Management Mechanism by Ontology. Tamkang Journal of S. and En., Vol. 13-1, pp. 79-87 (2010).

[28] Kamongi, P. et al. VULCAN: Vulnerability Assessment Framework for Cloud Computing. 2013 IEEE 7th International Conference, 2013, Page(s): 218 – 226.

[29] Ju An Wang, Minzhe Guo: OVM: an ontology for vulnerability management. CSIIRW 2009: p. 34.

[30] Henk Birkholz et al. Enhancing Security Testing via Automated Replication of IT-Asset Topologies. Proceedings of ARES '13, Pages 341-349.

[31] Atilla Elçi. Isn't the Time Ripe for a Standard Ontology on Security of Information and Networks, SIN '14 Proceedings, p. 1.

[32] HL7 Version 3 Standard: Security and Privacy Ontology, Release 1, May 2014.

[33] Tatiana Stepanova, Dmitry P. Zegzhda: Applying Large-scale Adaptive Graphs to Modeling Internet of Things Security. SIN 2014: 479.

[34] Dmitry P. Zegzhda, Tatiana Stepanova: Stochastic Model of Interaction between Botnets and Distributed Computer Defense Systems. MMM-ACNS 2012: 218-225.

[35] York Sure , Steffen Staab , Rudi Studer. Ontology Engineering Methodologies (2006), In Semantic Web Technologies: Trends and Research in Ontology-based Systems, Pages 71-79.

[36] A. Zouaq et al. A Survey of Domain Ontology Engineering: Methods and Tools. Advances in Intelligent Tutoring Systems Studies in Computational Intelligence, 2010, pp 103-119.

[37] T. Takahashi, H. Fujiwara, Y. Kadobayashi, “Building Ontology of Cybersecurity Operational Information”, 6th Annual Cyber Security and Information Intelligence Research Workshop, Apr. 2010.

[38] FOAF Vocabulary Specification 0.99, Namespace Document 14 January 2014, available at http://xmlns.com/foaf/spec/.

[39] Anna Estellés, Amparo Alcina. A model for formalizing characteristics in Protégé-OWL, available at http://ceur-ws.org/Vol-578/paper16.pdf

[40] Krassimir Markov, Vitalii Velychko, Oleksy Voloshin (ed.) Information Models of Knowledge ITHEA® Kiev, Ukraine – Sofia, Bulgaria, 2010.

[41] Horrocks, I., Patel-Schneider, P.F., van Harmelen, F.: From SHIQ and RDF to OWL: The making of a web ontology language. J. of Web Semantics 1 (2003).

[42] Simone Braun et al. The Ontology Maturing Approach for Collaborative and Work Integrated Ontology Development: Evaluation Results and Future Directions, 2013.

[43] Markel Vigo et al. Overcoming the pitfalls of ontology authoring: Strategies and implications for tool design, Open Access funded by Engineering and Physical Sciences Research Council, 2014.

[44] Timea Bagosi et al. The Ontop Framework for Ontology Based Data Access, available at http://www.ghxiao.org/publications/2014-csws-ontop.pdf, 2014.

[45] Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Poggi, A., Rodr´ıguez-Muro, M., Rosati, R.: Ontologies and databases: The DL-Lite approach. In: 5th Int. Reasoning Web Summer School Tutorial Lectures (RW 2009), vol. 5689, pp. 255–356. Springer (2009).