Information & Security

I. Skiter, I. Buraka & A.Sigayov
vol. 47, no. 3 (2020): 306-316
https://doi.org/10.11610/isij.4722

Published since 1998

ISSN 0861-5160 (print), ISSN 1314-2119 (online)

Research Article

Design of Technical Methods for Analysing Network Security Based on Identification of Network Traffic Anomalies

Ihor Skiter [a] (✉), *Ivan Burmaka* [a], *Andriy Sigayov* [b]

[a] *National Technological University, Chernihiv, Ukraine, https://stu.cn.ua/*

[b] *National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute," Kyiv, Ukraine, http://kpi.ua/*

ABSTRACT:

The article presents the design of a system for analysing technical networks with three main components. The attack generator monitors the network, checks its response, stability, and effectiveness to counter external threats. The database contains data about network parameters, their behaviour over time, network status, incidents, anomalies, etc. The network monitoring module uses information from the database for qualitative analysis of the network status.

The technical data analysis system of the distributed information system consists of two subsystems: the "Attacker" and the "Analyzer." The "Attacker" is a scanning tool for targeted information monitoring. It generates streams of network attacks with the aim to test the network response, stability, and effectiveness of network protection. The subsystem "Analyzer" collects information in predetermined periods of time, establishes criticality levels of network parameters; determines the time of the last criticality levels' change, records criticality levels values, and reports on the status, errors and script execution.

ARTICLE INFO:

RECEIVED: 12 JUNE 2020
REVISED: 09 SEP 2020
ONLINE: 22 SEP 2020

KEYWORDS:

database, network parameters, network monitoring, network traffic, anomaly

✉ Corresponding Author: E-mail: skiteris_mods@ukr.net

Introduction

The security problems of distributed information networks (DIN) or corporate networks (computer networks of ministries, large organizations, firms, banks, etc.) are associated with their following characteristics:[1]

- The complexity and heterogeneity of the used software and hardware
- A large number of corporate network nodes, their territorial remoteness, the inability to monitor all network parameters simultaneously, especially in real-time.

The DIN security structure consists of two main parts:

- Technical solutions that provide the required network security level.
- Methods and models of network state analysis used by them

The network security technical solution provides a mechanism of interaction between security subsystems, organization of communication channels, and the protection of information in them.

Methods and models for analysing the network state provide the identification of a non-standard network behaviour using modern mathematical and algorithmic apparatus.[2]

Technical means of network security (Fig.1) mainly consist of:

- Intrusion Detection Systems (IDS)
- Abuse Detection Systems (ADS).

These systems are used for monitoring network activity and primary qualitative analysis of network resources.[3]

This study examined the technical component of information security DIN from the point of view of designing decisions taking algorithms, applied methods and models that will provide effective, efficient and adaptive network security.

The design of the technical network analysis system

During the monitoring of the distributed information system, the assessment, control of the object, management of the state of the object are made depending on the influence of external and internal factors.[4]

The relevance of system development is the need to notify the system administrator about critical system states, failures, and other system problems, as well as to view information about the state of the DIN in real-time.

The design of the technical network analysis system, in our opinion, should be based on three main components: generator of attacks, database, and monitoring module (Fig.2).

This section provides a detailed description on the methods and procedures used in the study, and how they were selected among relevant methods.

The functions of the attack generator are monitoring of a network, checking of its reactions, stability, and efficiency for counteraction to external threats. A database that contains data of network parameters, their behaviour over time,

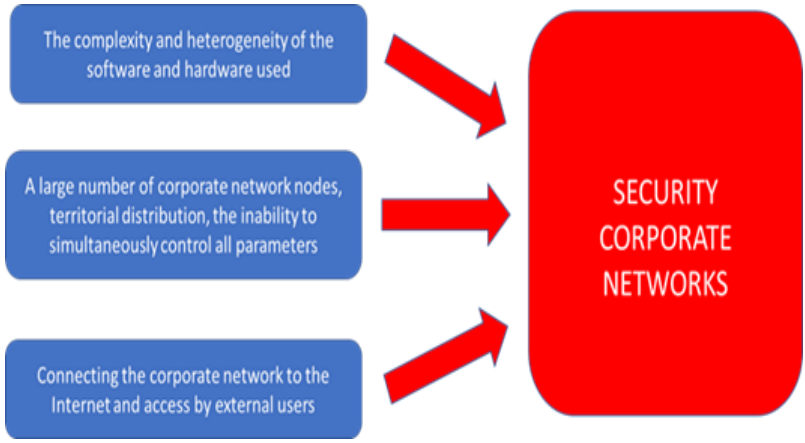

Figure 1: Security factors for a distributed information network.

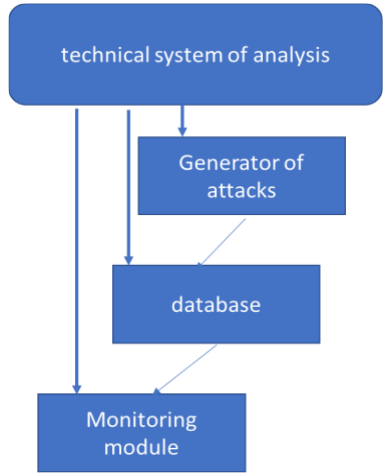

Figure 2: The structure of technical system DIN analysis.

data on network status, incidents, anomalies, etc. The network monitoring module, based on the information from the database, performs a qualitative analysis of the network status at the level of "stable-unstable."

The structure of the technical system for analysing DIN data consists of two subsystems (Fig.3): subsystem "Attacker" and subsystem "Analyzer" which includes the database, the WEB-server, and the WEB-client.

The operation algorithm of the system has such main steps:

- Monitoring the system with the subsystem "Attacker"

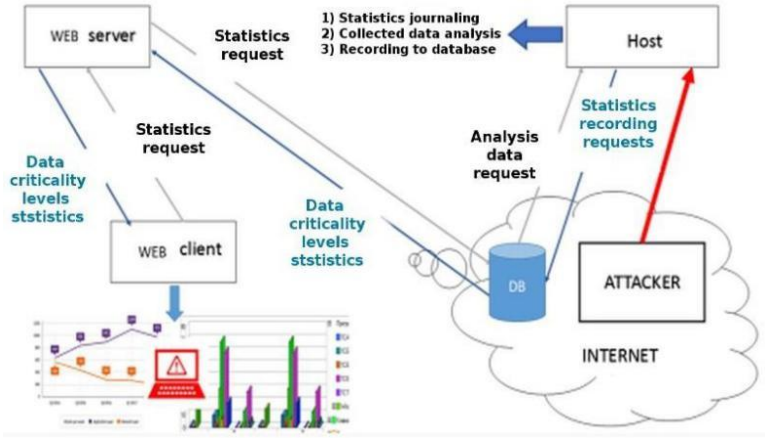

Figure 3: A structure of the technical system of analysis of data is in the Distributed (Corporate) Information Network

- Recording statistics on the state of the network
- Formation of a database about system parameters and
- Comparison of current values of network parameters with critical values - primary qualitative analysis (subsystem “Analyzer”).

The design of the subsystem "Attacker"

Purpose of the subsystem: is to scan target system and collect data about available vulnerabilities which can be used by attackers and generate based on it streams of network attacks.

Objectives of the subsystem: the module of the attack generator is designed to generate the flow of a raw traffic,[5] which comes to the monitoring system in order to test its response, stability and ability to respond to such attacks.

The objects of automation are the processes of scanning the characteristics of the monitoring system, their temporary storage in a file and generating attacks based on them.

Scanning includes the collection of the following data:

- Domain name of the system
- Block of used IP addresses
- OS version
- Open TCP, UDP ports
- Active services in the system.

This information is collected by the hacker manually and is used to generate a stream of invalid / empty packets.[6] The designed subsystem must provide the following quantitative indicators that characterize the degree of conformity to its purpose:

- Number of attempts to generate attacks – ones per minute

- Number of packages for one generation – adjustable
- Number of parameters by which packets are generated – 7 (listed above).

The main objects of the subject area are a controller (contains a key and parameters from the user, depending on the selected row Mode, carries scanning or commits attack); a parser (makes an attack or scan parameters in the configuration file); a scanner (analysing victim system); a generator (generates a packet and sends them to the target coordinates).

Designing process contains few steps, which was represented as a set of UML diagrams (context diagram, decomposition diagram, use case diagram class diagram, sequence diagram, and component diagram). The main components of the system and their functions are presented on logical representations – the model of analysis (Fig. 4), containing the main classes Controller, Scanner, Parser, Generator.

These classes are required to implement the highlighted use cases.

The implementation model is presented on (Fig.5) and is an architecture for the subsystem "Attacker." The subsystem was implemented on Python 2.7 because few libraries was unavailable for Python 3.x.

The "Attacker" subsystem provides the technical ability to monitor DIN in the control mode, to train, and to verify the effectiveness of the security status.

The design of the subsystem "Analyzer"

Designed to collect, store information about the state of the network, as well as to establish the criticality level of its states according to the selected parameters.[7]

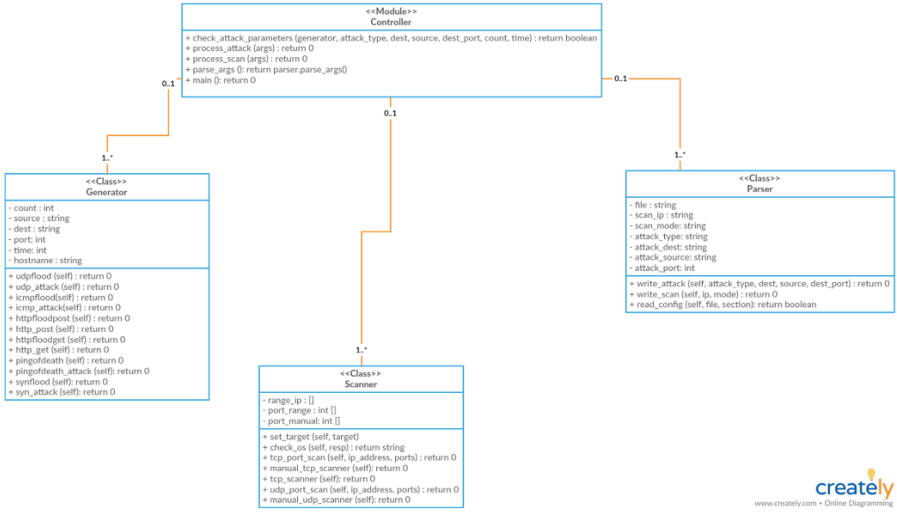

Figure 4: Subsystem "ATTACKER" Analysis model (logic level).

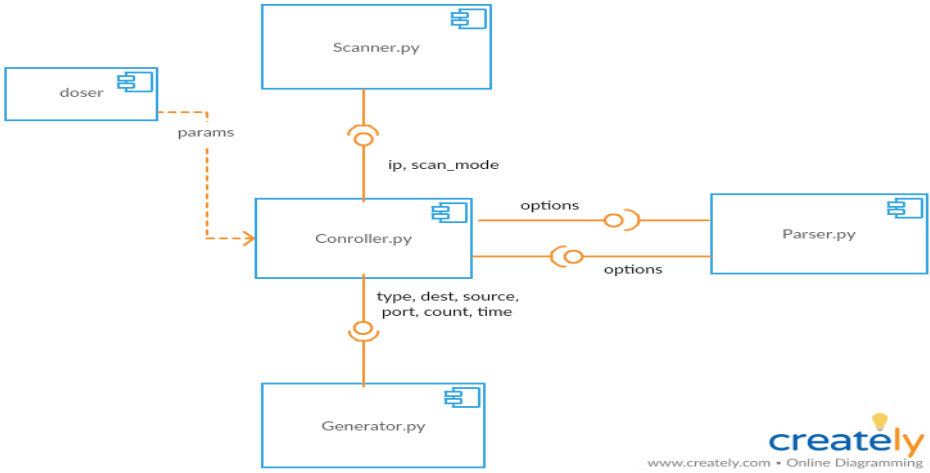

Figure 5: Architecture of a subsystem "Attacker" (implementation model).

The architecture of the system includes an information collection server (WEB-server) and a server on which the module for displaying network parameter state works (WEB-client) (Fig.6).

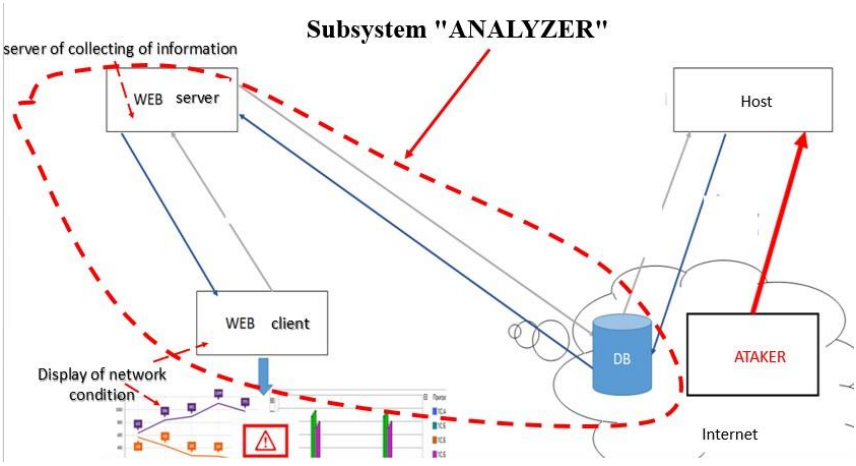

Figure 6: The structure of the subsystem "Analyzer."

Subsystem "Analyzer" is intended for working on the server, which can be attacked by an attacker. At the same time, the subsystem works in the background and sends data for writing to the remote database. This allows in critical situations like server overload, denial of service, or failure to communicate via

the Internet to conduct a separate remote control of the system through a graphical display module.

The “Attacker” subsystem, as mentioned above, is intended for testing and calibration of the reference values of network parameters. As well as attacks on a server with a running subsystem for collecting and analysing statistics.

The subsystem “Analyzer” has the following functionality:

- Collection of information at predetermined time periods
- Establishing the criticality level of network parameters
- Determining the time of the last criticality level change
- Recording criticality level values
- Saving records in the database
- Reporting about the status of subsystem, errors, and script execution.

Subsystem “Analyzer” architecture is represented as a set of UML diagrams (context diagram, decomposition diagram, use case diagram, class diagram, sequence diagram, and component diagram). The UML chart of use cases based on functionality is shown in the following (Fig.7), state diagram on (Fig.8).

Subsystem was implemented for Linux systems with Bash scripting language and MongoDB.[8] The “Analyzer” subsystem processes information, generates arrays of statistical data of network parameters, assesses the level of criticality, generates records and stores them in the database.

The functionality of the subsystem is divided between its components and has a few main categories for classification:

- Management
- Interaction with the database
- Collection of indicator values
- Logging.

The diagram of the components and relationships between them is presented on the (Fig.9).

The component “a set of scripts” collects statistics on individual parameters. The component “connectors DB” provides reacting with the database, receiving criticality level parameter values, time of criticality level change, recording. The component “controller” produces at the board components, scripts, data analysis, records forming. Thus, the technical side of the security of a distributed information network can be implemented based on the proposed design solutions.

Design Results

During the modelling, the existing ready-made solutions were used and the scenarios of the developed module were formed, the main functionality of the system was highlighted. The main and alternative scenarios of the system operation, the system's response to errors were developed.

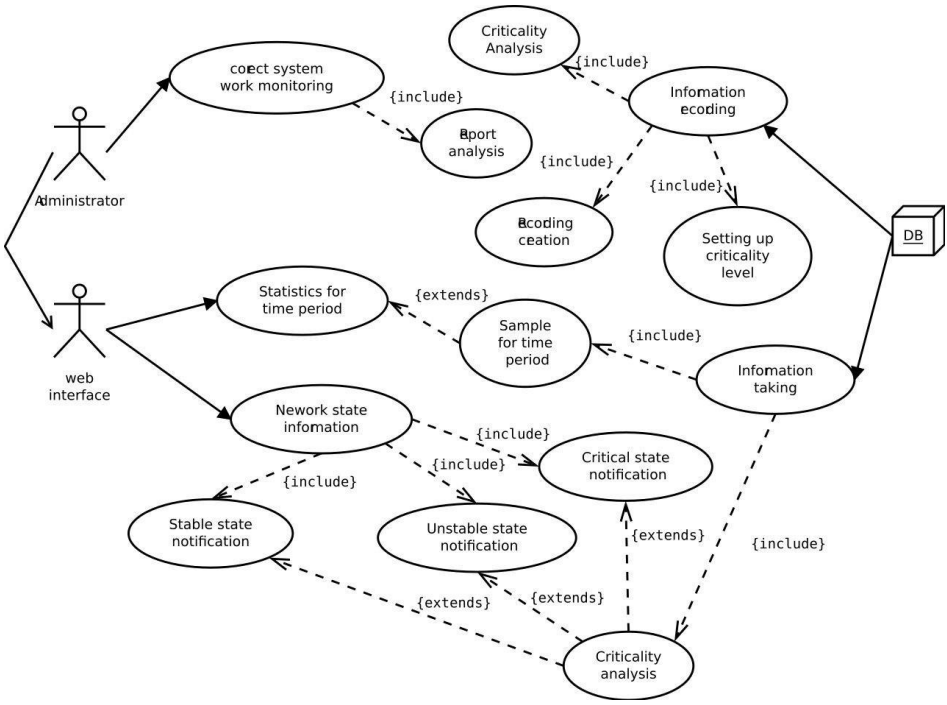

Figure 7: The UML chart of use cases based on functionality.

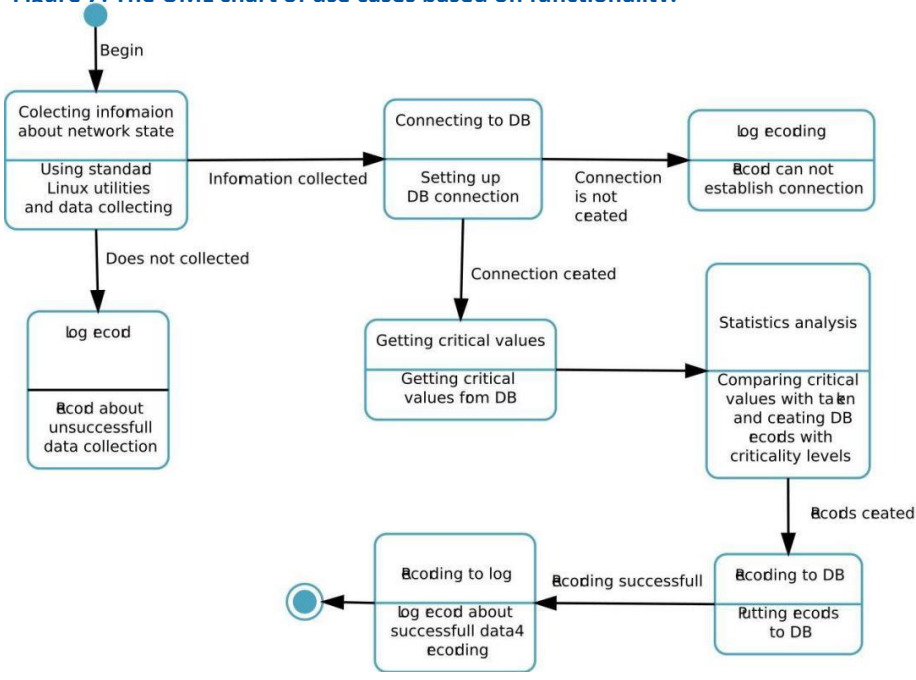

Figure 8: The state diagram.

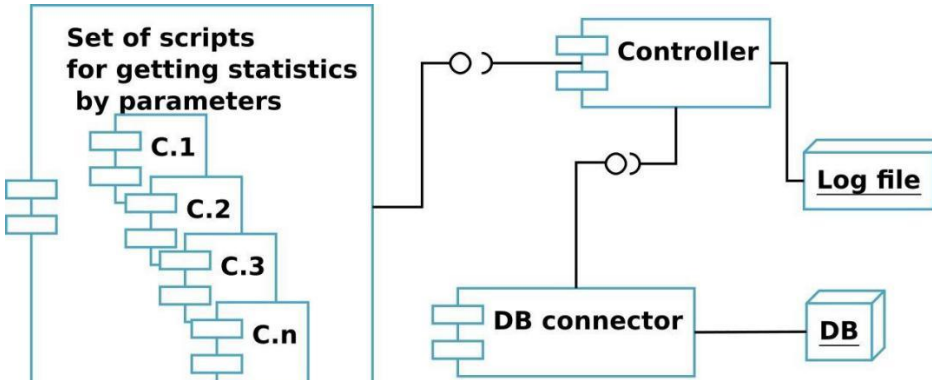

Figure 9: The Component Diagram.

The designed system will collect indicators automatically in the background without consuming many resources of the node and without affecting it. A method has been developed that will analyse the collected indicators from the network based on their respective reference values, and record the collected and analysed statistics to the storage of information.

Another very important advantage is that the node status information will be stored separately from the node itself, so that the graphical display module has constant access to statistics for display. Even when the monitored node is not online, turned off, or even failed to nodes in such a way that it would not be able to provide data for display.

The module receives indicators of the state of the node, their analysis, and the formation of records for their storage in the information repository, for its further display and notification of dangerous effects from the network to the node. The general project makes it possible to access the information repository and display network status data graphically, as well as notify the system administrator of possible external influences.

Conclusions

The functionality of the project provides the ability to conduct testing, conduct network attacks from a remote site to an object, and collect values. This makes it possible to determine acceptable and critical values of the network on the node, that is, the parameters in which the node performs all its calculations without the risk of failure. The obtained values of indicators are registered in the information repository for further analysis.

Currently, the system is being tested and preparing for full-fledged experiments on real distributed information systems. Technical methods for analysing network security which was developed can be used as a separate subsystem for evaluating complex security level of corporate network of critical infrastructure objects.[9]

Acknowledgements

The work was carried out and funded under the NATO project CyRADARS (Cyber Rapid Analysis for Defence Awareness of Real-time Situation) – grant agreement number: G5286.[10]

References

1 Maarten van Steen and Andrew S. Tanenbaum, “A Brief Introduction to Distributed Systems,” *Computing* 98 (2016): 967–1009, DOI: 10.1007/s00607-016-0508-7.

2 Nikolai Stoianov, Vitalii Lytvynov, Igor Skiter, and Svitlana Lytvyn, “Traffic Abnormalities Identification Based on the Stationary Parameters Estimation and Wavelet Function Detailization,” Mathematical Modeling and Simulation of Systems, *Selected Papers of 14th International Scientific-Practical Conference, MODS, 2019 June 24-26, Chernihiv, Ukraine: An Advances in Intelligent Systems and Computing* 1019 (2019): 83-95, https://doi.org/10.1007/987-3-030-25471-5.

3 Ashok Kumar and Venugopalan Srinivasagopalan Rajan, “Intrusion Detection Systems: A Review,” *An International Journal of Advanced Research in Computer Science* 8, no. 8 (2017): 356-370, http://dx.doi.org/10.26483/ijarcs.v8i8.4703.

4 Jeffrey Joyce, Greg Lomow, Konrad Slind, and Brian Unger, “Monitoring Distributed Systems,” *ACM Transactions on Computer Systems* 5, no. 2 (1987): 121-150, https://doi.org/10.1145/13677.22723.

5 Sunny Behal and Krishan Kumar, "Characterization and Comparison of DDoS Attack Tools and Traffic Generators: A Review," *International Journal of Network Security* 19, no. 3 (2017): 383-393.

6 Timur Bikbulatov and Ilya Kurochkin, “Simulation of DDoS attack on software defined networks,” *AIP Conference Proceedings* 2181, no. 1 (November 2019): 020022.

7 Manish Joshi and Theyazn Hassn Hadi, "A review of network traffic analysis and prediction techniques," *arXiv preprint arXiv* (2015): 1507.05722.

8 MongoDB, https://www.mongodb.com/

9 Vitalii Lytvynov, Mariia Dorosh, Iryna Bilous, Mariia Voitsekhovska, and Valentyn Nekhai, “Development of the Automated Information System for Organization’s Information Security Culture Level Assessment,” *Technical sciences and technologies* 1, no. 19, (March 2020): 124-32, https://doi.org/10.25140/2411-5363-2020-1(19)-124-132.

10 NATO SPS Project CyRADARS (Cyber Rapid Analysis for Defence Awareness of Real-time Situation), https://www.cyradars.net.

About the Authors

Ihor **Skiter,** PhD (physical & mathematical science), Ass. Prof. of Software Engineering Department of Chernihiv National Technological University. Scientific interests: data analysis, mathematical decision-making methods, system analysis, mathematical modelling of complex systems, econometrics.
https://orcid.org/0000-0003-2334-2276.

Ivan **Burmaka,** PhD student, Software Engineering department of Chernihiv National Technological University. Scientific interests: computer networks protection, artificial intelligence, blockchain systems.
https://orcid.org/0000-0002-7476-5757.

Andriy **Sigayov**, Professor in Igor Sikorsky Kyiv Polytechnic Institute. Scientific interests: artificial intelligence, machine learning, transfer learning explainable artificial intelligence.
https://orcid.org/0000-0002-8121-3782