РЕЗЮМЕТА НА НАУЧНИТЕ ТРУДОВЕ И УЧЕБНИ ПОСОБИЯ

на доц. д-р инж. Христо Георгиев Вълчанов
за участие в конкурс за заемане на академичната длъжност: ПРОФЕСОР
по професионално направление
5.3 „Комуникационна и компютърна техника”
. научна специалност „Компютърни системи, комплекси и мрежи“
учебна дисциплина „Администриране на локални и Интернет мрежи“
към катедра „Компютърни науки и технологии“
Факултет по изчислителна техника и автоматизация
обявен от Технически университет – Варна,
ДВ, брой 29 от 31.03.2023г.

Резюметата на научните трудове и учебни пособия са организирани в раздели както следва:

	Трудове за участие в конкурса за „Професор“	**брой**
В.4	**Публикации равностойни на монографичен труд на тема „Изследвания в областта на приложението на SDN и блокчейн технологиите за изграждане на интелигентни решения за облачни услуги ”**	**14**
Г	**Публикации извън групата на монографичния труд**	**56**
Г.7	Публикации в реферирани и индексирани в световноизвестни бази данни с научна информация	22
Г.8	Публикации в нереферирани списания с научно рецензиране	34

В.4 Публикации равностойни на монографичен труд на тема „Изследвания в областта на приложението на SDN и блокчейн технологиите за изграждане на интелигентни решения за облачни услуги"

B.4.1 Veneta Aleksieva, Hristo Valchanov and Anton Huliyan, Application of Smart Contracts based on Ethereum Blockchain for the Purpose of Insurance Services, 8-9.11.2019, Varna, BIA 2019, p. 69-72, ISBN 978-1-7281-4754-3, IEEE Catalognumber: CFP19U85-ART, DOI: 10.1109/BIA48344.2019.8967468

This paper presents an experimental implementation of smart contracts based on Ethereum blockchain for insurance services. A decentralized crypto-token, based on ERC20 standard for smart contract, is implemented. A web-based interface is created for sales of these crypto-tokens. The results from the experimental tests are presented.

The classic claim process can be improved by using smart contracts and blockchain technology. Information of loss can be sent from the insuree or directly from sensors mounted in the insured object (smart asset) to an automated claim processing application. For the relevant insurance policies provided by the smart contract, the customer will receive a real-time confirmation. The claim is automatically processed by a smart contract based on business logic, using the information provided by the insuree.

This approach automatically uses additional sources (statistics, reports) to evaluate claims and to calculate loss. Depending on the insurance policy, a smart contract can automatically calculate personal liability. In certain situations, a smart contract may activate an additional claim assessment. If the claim is approved, payment to the insuree is initiated by smart contract.

The advantages of the new approach, based on smart contracts on blockchain technology, can be seen in several aspects. Claim submission is simplified and automated. Thanks to the direct exchange of loss information between insurers, this approach eliminates the need of brokers and reduces the time needed to process a claim.

The embedded business logic in blockchain smart contract eliminates the need for loss adjuster to review every claim (except in specific situations). The insurer has access to the origin of the loss, which helps him to identify potential fraud attempts. The process of payment for loss is automated by the smart contract on the blockchain, with no need of an intermediary claims agent.

The proposed solution with smart contracts for insurance is based on the ERC20 standard. It has been implemented experimentally on Ethereum blockchain. The results of the experiments show that the proposed solution is fully operational in terms of managing automatic payments on approved claims for loss.

B.4.2 V. Aleksieva, H. Valchanov and A. Huliyan, "Implementation of Smart-Contract, Based on Hyperledger Fabric Blockchain", 2020 21st International Symposium on Electrical Apparatus & Technologies (SIELA), Bourgas, Bulgaria, 2020, pp. 1-4, doi: 10.1109/SIELA49118.2020.9167043.

The paper presents a solution for creation of a smart contract based on Permission Blockchain, in particular Hyperledger Fabric.
The proposed smart-contract is implemented on the computer with AMD Ryzen 5 2600 with 6 cores/12threads, 3.4GHz, 16GB DDR4 3200Mhz and SSD Nvme 500GB Read/Write Speed 3,500/2,700 MB/s. The operating system is Linux Ubuntu 16.04 LTS 64bit. In addition, Docker Engine Version 17.03 and Docker-Compose Version 1.8 are used.
The Blockchain network topology is next: There is one company (R1), which has one order node (O1) and one peer node (P1). It works with two different companies - R2 and R3. Each one of them has own consortium with main company. They are implemented in two independent channels – C1 and C2. Each consortium has two peers – C1 has P3 and P1, C2 has P2 and P1. As the peer P1 works for main company R1, it participates in two channels. L1 is a copy of Blockchain of C1, L2 is a copy of Blockchain of C2.
The Blockchain business solution is implemented by providing a connection between its individual organizations, for storage and exchange of information, as well as for its processing. The data are visible only between the organizations that have access rights, for which channels of communication have been established between them. To maintain the correctness of the data during recording and storage, peers are configured within the organization to maintain the operability of the network.
The Blockchain network uses a Docker container for the implementation of the Hyperledger Fabric. It uses the tool Docker Compose for defining and executing of multi-container Docker applications.
Once the Blockchain network has been configured and started, the business logic that will be executed on it must be implemented. The chaincodes are created with the programming language Go.
The tests are presented with Hyperledger Explorer for Fabric 1.4.x under Linux Ubuntu.
The proposed implementation allows fast and secure migration of smart contracts between independent channels. Each channel has own business logic and it is invisible for participants in other channels..

B.4.3 V. Aleksieva, H. Valchanov and A. Huliyan, "Implementation of Smart Contracts based on Hyperledger Fabric Blockchain for the Purpose of Insurance Services", 2020 International Conference on Biomedical Innovations and Applications (BIA), Varna, Bulgaria, 2020, pp. 113-116, doi: 10.1109/BIA50171.2020.9244500.

The paper presents an implementation of smart contracts for property insurance services, based on Hyperledger Fabric Blockchain.
The private Blockchain as Hyperledger Fabric Blockchain is the better solution for insurance business, because of its trusted nodes, there is no requirement for consensus protocol. The main advantages are the quick access to information, the cheaper transactions and the control on privacy level. Due to these facts, it is suitable and useful in many areas of insurance services.
In the presented use case, two channels are created: one for consortium 1 (Channel 1) of company Org1 (insurance company) and company Org2 (broker 1), and one for consortium 2 (Channel 2) of company Org1 and company Org3 (broker 2). Each channel has its own Blockchain, as well as smart contracts (codechain) that work alone with it. Each consortium has two participants. The two channels work in parallel with each other and they are not visible to participants outside those allowed by the rules of the consortium. A peer can contain a copy of the Blockchain and smart contracts of more than one channel.
The proposed smart-contract is implemented on the computer with AMD Ryzen 5 2600 6 cores/12threads, with Linux Ubuntu 16.04. In addition, Docker Engine Version 17.03 and Docker-Compose Version 1.8 are used. The name of the created smart contract is mycc and it is installed on Peer 0 on Org1 in Channel 1. In the implemented business logic, it is possible the client to change its broker. This means that his policy must move from one channel to another channel. There are two possible scenarios, after copying it – it will remain visible in Channel 1, and the changes made after copying Channel 2 will not be visible. The other scenario is that it will be deleted, so it will no longer be visible for Channel 1 participants.
The testing of the use case operability is performed by sending requests to the installed chaincodes and checking their correct execution. The Hyperledger Explorer tool is used to visualize the created network for this experimental use case. To find information about a person, who is written on the Blockchain network, the function queryOwnerByName from smart contract runs with the script.
Smart contracts provide the opportunity to create policies, monitor their status and through the business logic that can be described in them, to automate the process of processing insurance claims.
Through smart contracts it is possible to create an insurance policy, to determine the insurance risk, to execute of insurance claims. The Blockchain also optimizes the reinsurance process, as well as the operations of brokers. In areas, where supply-side monitoring is required, this new solution will improve the insurance process.

B.4.4 V. Aleksieva, H. Valchanov and A. Huliyan, "Smart Contracts based on Private and Public Blockchains for the Purpose of Insurance Services", 2020 International Conference Automatics and Informatics (ICAI), Varna, Bulgaria, 2020, pp. 1-4, doi: 10.1109/ICAI50593.2020.9311371.

This paper presents an experimental implementation of smart contracts based on Hyperledger Fabric Blockchain for insurance services in comparison with another implementation of smart contracts based on Ethereum Blockchain.
The use case is the same in each implementation: the insurance company (Org1) works with four companies – Org2 (broker 1), Org3 (broker 2), Org4 (broker 3), Org5 (broker 4). Each consortium has two participants – an insurance company and one broker company. The company Org1 has its own peer Peer0, provided that it participates in four consortia and has copy of four smart contracts. The Peer0 has main role in the insurance process, because he administrates the relations between insurance and broker's companies in each consortium.
The proposed solution was developed with Metamask, Truffle and Ganache under the MacOS High Sierra operating system. Ganache creates a local Blockchain based on Ethereum, which can directly execute commands as well as perform tests. Metamask is used, as there is no need to download a local copy of the Blockchain. A connection to a site makes a connection with Ethereum. Metamask takes care of all requests from and to the Blockchain network. Metamask can perform the Ethereum wallet function and support sending and receiving Ethers and ERC20 tokens. Truffle is used for the implementation of the smart contract. It is an integrated system for compilation of the written smart contracts, and it uploads them on the Ethereum network.
The same use case in Hyperledger Fabric is based on four channels. The proposed smart-contract based on Hyperledger Fabric is implemented on the computer with processor AMD Ryzen 5 2600 6 cores/12 threads and with operating system Linux Ubuntu 16.04. In addition, Docker Engine Version 17.03 and Docker-Compose Version 1.8 are used. The main difference from decision, based on public Blockchain, where the network exists, is that in private Blockchain the first step is to create the Blockchain network.
The proposed public solution with smart contracts for insurance is based on the ERC20 standard. It has been implemented experimentally on Ethereum Blockchain. The results of the experiments show that the proposed solution is fully operational in terms of managing automatic payments on approved claims for loss. In the proposed Smart contract, the business logic is more complex and the solution is more expensive then solution, based on private Blockchain, because it needs to pay for computing power with "ETH" tokens.
The proposed private solution with codechains on Hyperledger Fabric is more flexible, more secure, faster, and cheaper than previous public solution..

B.4.5 Veneta Aleksieva, Hristo Valchanov, Smart Contracts based on Hyperledger Fabric Blockchain for the Purpose of Health and Life Insurance Services, CIEES'21, AIP Conference Proceedings 2570, 020002, ISBN 978-073544375-4, DOI 10.1063/5.0099626

This paper presents a solution, based on smart contracts on a Blockchain, whereby the insurer pays directly to the hospital for the services provided in favor of the insured and only if the sum insured is insufficient, the patient pays the hospital.
The disadvantages of the classical process in Bulgaria from the point of view of the insuree are:

- The person must pay directly for his/her treatment, which will be reimbursed in weeks or months.
- At a time when a person's health is a top priority, s/he has to provide documents and visit an insurer to reimburse the costs incurred by him, sometimes repeatedly.

To avoid these shortcomings, the solution proposed in this paper is with a smart contract on a Blockchain. The steps of the process are:

1. The illness/accident of the insuree, for which treatment has to be applied.
2. The treatment includes medical examinations, hospital treatment, outpatient treatment (prescription drugs, monitoring of the condition of the insuree by the GP, control examinations by specialists).
3. In case the person is obligatory insured and / or voluntary insured for the payment of the treatment, a smart contract is executed, which checks whether the person is insured (if yes – it orders the coverage of the amounts by the NHIF, according to an approved list of amounts it covers, as for the rest of the treatment amounts checks the available insurance amounts for the person and orders the coverage of the amounts by the respective insurer, entering in the policy of the insured the spent amount and only in case the treatment amount cannot be covered by NHIF and insurer, the person pays extra with direct payment.

The proposed implementation is based on Hyperledger Fabric. For each insurance company is created own channel (consortium). Each channel has its own blockchain, as well as smart contracts (codechains) that work alone with it. Each consortium has two peers – Peer0 from Org1 and another peer from insurance's company. The four channels work in parallel with each other and they are not visible to participants outside those, allowed by the rules of the consortium. Peer0 has separate copies of the four blockchains. The business logic of the blockchain network is implemented by language Go.
With the proposed decision, the insuree will avoid direct payments. It will reduce paperwork, will eliminate the need for an expert for the insurer, which will reduce its operational costs and the risk of one insurance event to use two policies with overlap rather than supplementation. The experimental results are presented, which prove the applicability of the proposed solution.

B.4.6 D. Todorov, H. Valchanov and V. Aleksieva, "Load Balancing model based on Machine Learning and Segment Routing in SDN", 2020 International Conference Automatics and Informatics (ICAI), Varna, Bulgaria, 2020, pp. 1-4, doi: 10.1109/ICAI50593.2020.9311385.

This paper proposes a model which aims to reduce not only the overall load on the SDN network, but also to reduce the bandwidth and improve the routing mechanism on the SDN networks. It combines segment routing algorithm and load balancing mechanisms based on neural networks. The main purpose of this model is to investigate the most compatible neural network model for network load balancing and to minimize the network traffic between the controller and network devices.
There are different load balancing mechanisms in SDN, which uses two main approaches – static and dynamic load balancing. The shortcomings and problems with approaches and algorithms in related works are considered.
The model is developed as a cross platform SDN controller using C++ programming language. It implements the OpenFlow protocol and follows OS specific system calls for higher performance.
The system contains four main modules: SDN controller module, Prediction module, Path compute module and Path encoding module. Collected network parameters from the system are used to compute the optimal path based on neural network algorithms. The parameters are reduced to a single coefficient, which is then used for training and prediction purpose.
Using a Q-Learning algorithm, the process is split in two flows. First there are no data on which to make the prediction. To fill that data, the module starts to learn and receives a reward for every successful conjunction. Once the model is trained, the module can predict any flow change. Once the connection is established, the controller sends a status check packet to get the network information of the device.
When receiving the information, the controller stores it in a Network Global View database. After that, the controller and the switch start to exchange echo packets to track the connectivity between them. These packets are used to track the bandwidth of the device.
The prediction process monitors possible changes of the network load. If it detects such ones, it sends a signal to the Path Compute Module, to update the Flow Tables with needed routes to make the network load balanced. Once the optimal paths are computed, they are sent back and installed to the switches. The controller also notifies the Prediction process of dropped network devices, which will automatically trigger the Flow table changes.
The proposed architecture model combines neural network algorithms with segment routing for achieving better performance and network load balancing. It improves the QoS and provides ability to predict overload of network routes.

B.4.7 D. Todorov, H. Valchanov, V. Aleksieva, Shortest path routing algorithm with dynamic composite weights in SDN networks, ICAI'21,30.09.-02.10.2021, Bulgaria, pp. 193-197, doi: 10.1109/ICAI52893.2021.9639512 ISBN:978-1-6654-2661-9

In this paper is proposed a shortest path routing algorithm with dynamic composite weights in SDN networks using OpenFlow protocol. The algorithm choses the less loaded path based on dynamic node and edge weights by observing link loads between switches. In order to discover the network topology, the algorithm uses LLDP to find links between switches and relies to ARP messages to find hosts linked to switches. By this way it can perform routings through ghost switches, which does not support OpenFlow.
The experimental study is performed under Windows OS, and Mininet simulator is used to simulate network topology and traffic. The Mininet simulator is running on a Virtual Machine on the same host machine where the controller is running. The controller is developed using C++ programming language and implements OpenFlow for SDN management.
The examined routing algorithms are:

- the authors' Simple routing algorithm with link discovery between source and destination hosts from our previous work;
- Dijkstra's routing algorithm which routes the traffic based on minimum hop count;
- Proposed shortest path routing algorithm with dynamic composite weights.

The experimental results show that the algorithm performs better compared to Simple routing and Dijkstra's routing algorithm. During the topology discovery phase, it shows less network traffic even with the use of LLDP. This is due to a huge exchange of ARP messages between the switches used by Simple routing algorithm to link network devices. Also, the algorithm successfully achieved is main purpose to load balance the network traffic and provide better QoS.
It has a little increase of the latency during packet transfer, but this is due to a packet flow changes during the routing process in order to prevent congestions. During the experiments, all three algorithms have zero drop rate and all packets were transferred successful. In addition, all of examined routing algorithms have successfully processed network loops and have low use of memory and processing power from the controller..

B.4.8 D. Todorov, H. Valchanov, V. Aleksieva, Simple routing algorithm with link discovery between source and destination hosts in SDN networks, ICAI'21,30.09.02.10.2021, Bulgaria pp. 188-191, doi: 10.1109/ICAI52893.2021.9639742, ISBN:978-1-6654-2661-9

In this paper we present a simple routing algorithm with link discovery between source and destination hosts in SDN networks without taking into consideration the link cost. The algorithm reduces the messages passed between network devices and the controller, as well as the path computation for the flows. For the implementation and testing we have developed an OpenFlow controller which performs the main interactions with network devices and use Mininet emulator to perform experiments.
The system contains two main modules: SDN Controller module and Path Compute module. The SDN Controller module supports the main communication functions between the controller and switches. It stores information about connected devices and their flow tables in Global Network View database. To establish a connection between the controller and the switch, handshake packets are exchanged. After establishing a successful communication, the controller periodically sends echo packets to track switch accessibility.
The Routing module is responsible to find destination address in flow entry table and to provide the next hop for the packet to the controller. It doesn't take into consideration the network load to achieve load balanced traffic. The module takes into consideration the first served ARP message based on which makes the decision where to redirect the packet.
The experimental study is performed under Windows OS. To simulate network topology and traffic Mininet simulator is used. Mininet is running on a Virtual Machine on the host machine. The controller with routing algorithm is running on the host computer. The controller is developed using C++ programming language and implements OpenFlow for SDN management.
The experimental results show that the algorithm achieves its main purpose to reduce the network traffic between the controller and network devices during the discovery phase. The algorithm doesn't use Link Layer Discovery Protocol (LLDP) to find connections between network devices. By this way it eliminates the additional traffic and preserves the network bandwidth. The algorithm has zero drop rate and all packets are transferred successful. It also shows low times on transferred packets. Another advantage is successfully processing of network loops in mesh topologies and the lower use of memory and processing power from the controller.

B.4.9 Veneta Aleksieva, Hristo Valchanov, Monika Vangelova, Cloud Based System for Reservation of Medical Appointments, AIP, CIEES'21, AIP Conference Proceedings 2570, 020002, ISBN 978-073544375-4, DOI 10.1063/5.0099627

This paper presents a cloud-based system for booking appointments for clinic examinations and remote consultations. A comparison is made among three different solutions. The experimental results show that the proposed cloud-based solution is the best option in terms of response speed, scalability, easiest administration and cost-effectiveness.

The authors proposed web-based system CollosalClinic_Online. For implementation are used different tools and frameworks such as C#, HTML, CSS, JavaScript, Bootstrap, jQuery, Google API, ASP.NET. The development is in the integrated environment MS Visual Studio 2017, and the management of the relational database is with MS SQL Server 2019. The web server is Apache 2.4.46, and Internet Information Services 10.0 is used for web application and a site management, containerization and fast Cloud integration. It is tested on a local computer.

The second implementation is in the distributed environment with a platform VMware Workstation Pro v.12.5.1.

The third implementation is in the cloud Azure CDN. The application is accessed via the Internet with an URL generated by Microsoft Azure with an Azure domain, https://purple-forest-09d81c203.azurestaticapps.net. Microsoft Azure allows to build a dashboard to monitor the resources and performance of the system. A basic Dashboard is formed, in which all the necessary graphs for a real-time monitoring are built and adjusted.

A comparison was made among the three implementations. The results show that the Cloud based solution is the fastest, most efficient, it has excellent performance and fault tolerance. After comparing this solution with five other existing solutions for booking of medical examinations and according to the results of measuring loading time, downloading resources and the number of requests to the servers where the applications are hosted, Cloud implementation of the proposed system has the best performance indicators.

B.4.10 D. Todorov, H. Valchanov, V. Aleksieva, Comparative Evaluation of Traffic Load Balancing and QoS in SDN Networks, AIP, CIEES'21, ISBN 978-073544375-4, DOI 10.1063/5.0099807

In this paper is proposed different important criteria for implementing a comparative evaluation of traffic load balancing. At the end it is presented a complex comparative analysis of static and dynamic routing algorithms for traffic load balancing and QoS improvement in SDN. For static routing, three algorithms were compared – Open Shortest Path First algorithm, shortest widest path and simple routing with link detection proposed by the authors in other research. For dynamic routing three algorithms were compared – Extended Dijkstra's algorithm, Enhanced Interior Gateway Routing Protocol and dynamic routing with complex weights also proposed by the authors.
The experimental study and results are obtained under Windows OS, and Mininet simulator is used to simulate network topology and network traffic. On the same host OS where the controller is running, the Mininet simulator is ran as Virtual Machine. For the purpose of experimental study, the controller is developed using C++ programming language and implements OpenFlow for managing SDN network. The controller supports the main functionalities for managing SDN network and have in place implemented Open Shortest Path First, simple routing with link detection and dynamic routing with complex weights algorithms.
The controller implements all up to date versions of OpenFlow communication protocol and have modular design. It stores the global network view in operational memory and has the ability to install flow rules on network resources, as well as process each packet independently using packet in and packet out messages. The controller has the ability to detect switch accessibility using echo messages, which are exchanged each 5 seconds. It also supports ARP messages to discover connected hosts to network resources and stores their MAC addresses in operating memory by mapping the host to the corresponding switch with which it has a physical connection.
The tests are made with different topologies for each routing algorithm.
The authors propose the system of criteria for comparison and complex evaluation of these routing algorithms. If the criteria are observed separately, it can be seen that the proposed by authors static routing algorithm does not provide good results for "Network load" compared to the two other algorithms, but it has equal results with them for supporting all network topologies. Also, observing the criteria of dynamic routing mechanisms shows that the proposed by authors algorithm has equal results for "Packet drop rate", "Topologies" and "QoS support". Due to the wide range of criteria in the complex assessment, the overall score for geometric and arithmetic complex estimation of both proposed by authors algorithms is better than the algorithms with which it is compared.

B.4.11 H.Valchanov, V.Aleksieva. Novel blockchain - based models for healthcare and life science solution, CIEES'22, 2022, K2.4, Electronic ISBN:978-1-6654-9149-5, DOI: 10.1109/CIEES55704.2022.9990883

The health policy related to vaccination against infectious diseases of any country aims to limit epidemics and preserve the health and working capacity of citizens, prolong and improve their lives. Coverage of the entire population with vaccination is one of the key factors in achieving these goals. This paper proposes several models based on blockchain technologies that ensure vaccination of the entire population, ignore the possibility of duplication and falsification of information, minimize paperwork between institutions involved in the vaccination process, improve the planning process of necessary vaccines, etc. The models are implemented on the Hyperledger Fabric private blockchain.
The first model is focused on the patient. For each citizen at the time of his birth (or immigration), a separate channel is created, numbered with the personal identifier of the patient (PID), which acts as an individual immunization passport. Information about each vaccination is recorded in a separate block in the channel (consisting of the date of vaccination, type of vaccine, vial code, GP code and concomitant reactions to the vaccine.
The second model focuses on the immunization schedule for diseases for which mandatory vaccines are administered. A separate channel is created for each year. It stores vaccine information for people born that year.
The third model focuses on diseases for which vaccines are mandatory. Separate channels are created for each of the diseases subject to prevention through mandatory immunization. The smart contract compares the data with the information from the Central Civil Registration System (ESGRAON) about the vaccinated persons and the Ministry of Health receives accurate information about the necessary vaccines. RZI will receive information about vaccinated persons who are not vaccinated.
The proposed models are fully in line with the EU health policy for digitization and modernization of health systems. These models propose a change in the current document flow between institutions through the introduction of a blockchain-based national immunization registry. In this way, they guarantee the reliability of the vaccinations performed on both the person and the follow-up of unvaccinated persons for a specific mandatory vaccine. On the other hand, the models provide a better planning of the required quantities of vaccines.

B.4.12 H.Valchanov, V.Aleksieva. Smart Contracts based on Hyperledger Fabric Blockchain for the Purpose of Higher Education Subsidizing, CIEES’22, 2022, K2.5, Electronic ISBN:978-1-6654-9149-5, DOI: 10.1109/CIEES55704.2022.9990828

During their studies, many students drop out of one major, then apply and enroll in another. The state subsidizes several semesters of their studies, then again subsidizes the student's studies in another specialty. A number of students start but do not complete their studies, but this is also subsidized by the state. No research has focused on the state's losses from subsidizing education in these cases. This paper proposes a model where every citizen can receive a subsidy for their studies, but only for up to 10 semesters. He must then pay the full cost of his education. Under this model, students are motivated to complete their studies within the subsidized semesters, and the state does not suffer losses from refinancing the same student whose study period exceeds these semesters. The model is based on a smart contract on the Hyperledger Fabric platform.
There is a separate channel for each state university. The corresponding university can read and write blocks. The Ministry of Education has read-only access to this channel. The channel records in each new block information about an enrolled student for the corresponding semester. The smart contract for the respective channel calculates the amount of each subsidy according to the student's specialty and according to the current coefficients and the base subsidy determined for the respective year by regulations.
HyperLedger Fabric is implemented using Docker platform. Each of the Fabric peers runs in a separate container. Containers can run together on a single machine or each on a separate node. Since the communication is based on the TCP/IP protocol suite, it enables the application of the proposed model on distributed platforms.
The business logic of the proposed model is implemented by smart contracts based on the Go language. Interaction with them is done by calling their methods. Various objects are used to store information.
The use of a special container for executing commands allows efficient testing of the developed program code. Multiple scripts have been created containing queries calling smart contract methods with various data. This allows to automate the testing process by repeatedly executing in debug mode.
Experimental results are presented that prove the applicability of the proposed solution. The goal of future work is to implement the model in a distributed environment and develop an API user interface to smart contracts.

B.4.13 H.Valchanov, V.Aleksieva. Blockchain and IoT integration for smart transportation, International Conference on Electronics, Engineering Physics and Earth Science (EEPES 2022), Journal of Physics: Conference Series, Volume 2339, pp.1-8, Online ISSN: 1742-6596, Print ISSN: 1742-6588, DOI: https://doi.org/10.1088/1742-6596/2339/1/012012

The safe transport of special and dangerous goods is essential for the ecological environment and human health. Modern solutions are based on monitoring the parameters of their transport with sensors in real time, which allows a quick reaction to unexpected events. Despite the available information from the sensors, the process of proving an insured event and paying compensation is the same as for other insurances. This paper proposes a blockchain and IoT-based model that records chronologically the data from the sensors located in the goods vehicle and the smart contract that sends timely alerts to the stakeholders (including the insurer) when the sensor parameters exceed the set thresholds. An experimental implementation on HyperLedger Fabric is presented, which proves the applicability of the proposed model.
Damage information can be sent directly from the sensors installed in the insured object (container, vehicle) to an automated claim processing application. For the relevant insurance policies provided by the smart contract, the customer will receive real-time feedback. The claim is processed automatically by a smart contract based on set business logic using information provided by the insurer. The smart contract automatically uses additional sources (statistics, reports) to evaluate the claim and calculate damages. Depending on the insurance policy, the smart contract can automatically calculate personal liability. If the claim is approved, payment to the insured is initiated through the smart contract.
Testing of the proposed model and its implementation is done by sending requests to a smart contract. A web-based tool - Hyperledger Explorer - is used to visualize the network and extract statistical information.
The proposed model offers chronological and transparent tracking of sensor data located on the container/vehicle, sending timely alerts to stakeholders when sensor parameters exceed thresholds. The advantages are in reducing the documentation and operating costs of the insurer in the event of an insured event, eliminating the possibility of fraud, improving customer satisfaction when handling claims. The obtained results of the experiments prove the applicability of the proposed model.

B.4.14. H.Valchanov, V.Aleksieva. Novel Model for Hospitalization Tracking based on Smart Contracts and IoT, ICAI'22, pp.14-17, E ISBN:978-1-6654-7625-6, DOI: 10.1109/ICAI55857.2022.9959996

In recent years, fraudulent hospitalization fraud and abuse in healthcare has become a serious problem requiring monitoring of hospital bed occupancy. The control will increase the social, health and economic efficiency of health care spending, which in turn will improve the quality of health services. The paper proposes a new blockchain-based model. In this model, data from sensors located in hospital beds and a fitness tracker for each patient are recorded chronologically. At the same time, there is both monitoring of the location of the tracker and monitoring of the patient's vital signs. The smart contract sends timely alerts to stakeholders when sensor parameters exceed set norms.
The proposed model is part of a solution for smart hospitals, where patients are equipped with health devices that monitor their vital signs and share them with other authorized users on the blockchain network. This model can track events in real time without being able to be manipulated.
The business logic of the NHIF smart contract monitors hospitalizations in real time and, on this basis, automatically calculates the reimbursement amounts to the medical facility. In this way, constant control over the amounts claimed for treatment by the hospitals is carried out, the possibility of fraud is eliminated and the costs of the NHIF for claimed but not actually carried out hospitalizations are reduced. In this way, the funds from the NHIF are spent transparently and only for actually performed services.
The proposed model is implemented on a private blockchain - HyperLedger Fabric and using Docker containers. Each hospital is a separate organization with partners through which the respective blockchain is accessed. IoT components (sensors) record the information about the patient's indicators in the corresponding channel - only they have the right to write to the channels. The data from the channels is read by the partners of the respective hospitals, as well as by those of the insurance companies. The NHIF is a partner of all channels and can only read information from them.
A number of experiments have been conducted and the results presented. They clearly prove the applicability of the proposed model and its advantages over the traditional solution for hospitalizations.
The advantages of the proposed model can be expressed in the following directions: improving the quality of monitoring the health status of hospitalized patients, reducing document processing and operational costs, eliminating the possibility of fraud and offering better control over hospitalizations.

Г. Публикации извън групата на монографичния труд

Г.7. Публикации в реферирани и индексирани в световноизвестни бази данни с научна информация

Г.7.1 Veneta Aleksieva, Hristo Valchanov and Diyan Dinev, Comparison Study of Prototypes based on LiFi Technology, 8-9.11.2019, Varna, BIA2019, p.73-76, ISBN 978-1-7281-4754-3, IEEE Catalog number: CFP19U85-ART, DOI: 10.1109/BIA48344.2019.8967478

In this paper is proposed a prototype for LiFi data communication and a comparison study of proposed by the authors LiFi prototype with other similar LiFi prototypes. The limitations of the proposed prototype are:

- The maximum distance of transmission is when the LED bulb is at an angle of 90o with horizontal plane. This angle guarantees the maximum transmit distance of 80cm. This result is achieved in a fully dark room.
- The more we decrease the angle of the LED bulb, the more the distance for successful transmission decreases. When the angle is less than 40o the receiving of data is unsuccessful.

The investigation of the proposed prototype focuses on the impact of some environment factors (as sunlight illuminance, glass border and saline water). As the experiments with the prototype have been made, its limitations for maximum transmit distance, under different environment conditions, are determined.

The goal of the current study is to collect data for transmission through different environment. A prototype, which was designed for previous research was used, but in software some enhancements are made, such as transmission speed and error correction improvements.

The direct sunlight (in this experiment—7520 lux) leads to 100% loss of the transmitted information to the receiver, even if it is 1 cm away from the transmitter. As much as we decrease the illuminance of the sunlight by moving away from the window, the distance Dmax increases. When reaching 2.5 m (200 lux), Dmax is 60 cm. The value of Dmax of 80 cm is reached on distance of 4.0 m from the window (where the impact of sunlight is 0 lux). This is the same as the maximum transmitted distance, which is reached in a fully dark room.

If the thickness of a glass border is only 2 mm, the distance is the same as the distance without a border. But if the thickness grows up, Dmax decreases. With glass border of 12 cm Dmax is only 40 cm—the half of the maximum distance.

Experiments with fresh and salt water (10% saline and 20% saline) were performed. Air is excluded, as the transmitter and receiver modules are stuck to the glass of the aquarium. On distance 55cm only 0% saline water accepts the communication, but on distance 26cm up to 20% saline water accept the communication.

The key performance indicators are chosen for the estimation and comparison of the prototype, subject of the above mentioned experiments, and other LiFi prototypes. Based on them, the comparison is made. The estimation is made with two complex criteria—average arithmetic and average geometric estimation. Regarding to the results of the evaluation, it can be concluded that the above mentioned prototype is the best option for the purposes of the present study.

Г.7.2 Haka, V. Aleksieva and H. Valchanov, "Comparative Analysis of Traffic Prioritisation Algorithms by LTE Base Station Scheduler," 2020 21st International Symposium on Electrical Apparatus & Technologies (SIELA), Bourgas, Bulgaria, 2020, pp. 1-4, doi: 10.1109/SIELA49118.2020.9167040.

This paper examines the impact of the proposed by the authors traffic prioritization algorithm in LTE network, Round Robin (RR), Maximum Rate (MAX-Rate), Proportional Fair (PF), Exponential/Proportional Fair (EXP-PF), and these proposed by Myo and Akyıldız on the QoS in 4G LTE wireless mobile network.
The comparison based on the results of throughput, delay, packet delivery ratio (PDR) and packet loss ratio (PLR). To study the impact of traffic prioritization algorithms on QoS, the LTE simulation product proposed and further developed by the authors, is used. Experimental studies were performed for static and mobile UEs for one LTE cell, for which transmit power is 40W (46.02dBm), 20 MHz bandwidth, noise power is -160.99dBm, 100 available PRBs, 6 sectors cell, and radius of 770m. Number of users used 20, 50, 70 and 100. Distance of static UEs to eNodeB (m) used for the studies are as follows – 10, 90, 170, 250, 330, 410, 490, 570, 650 and 730 (55m for all mobile UEs). Required type of service is GBR, required RBs from every UE are 5555 and pay price for guaranteed service with value 5. Movement Speed for Mobile UEs (km/h) used for the studies are as follows – 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100.
The presented results show that with a smaller number of subscribers, the proposed algorithm provides higher values for the studied parameters for static subscribers located within a range of up to 250 meters from the eNodeB and provides higher values for the studied parameters for mobile subscribers moving at more than 80 km/h. With the increase in the number of subscribers, the serving becomes equable, but better values provided for the highest priority subscribers, while for the other algorithms the results are almost uniform.
The advantage of the proposed algorithm over others is that it serves with high priority requests from subscribers at a closer distance to the eNodeB and requests from mobile subscribers. Serving requests from subscribers closer to eNodeB improves the QoS because the channel quality of these subscribers is better and communication errors are less, which leads to faster service. Priority service for queries from mobile users improves the QoS because this reduces the loss of handover. Allocating more resources to the higher priority users will speed up the service of their requests and the resources released by them will be used to serve the low priority ones.

Г.7.3 Haka, V. Aleksieva and H. Valchanov, "Comparative Analysis of Traffic Prioritisation Algorithms in 6LOWPAN Networks," 2020 21st International Symposium on Electrical Apparatus & Technologies (SIELA), Bourgas, Bulgaria, 2020, pp. 1-4, doi: 10.1109/SIELA49118.2020.9167116.

This paper introduces a comprehensive comparative analysis between the suggested 6LoWPAN sensor network traffic prioritization algorithm by the authors and five standard sensor network algorithms. There are two main classes of traffic prioritization algorithms for sensor networks: Knowledge Free and Knowledge Based.
In the author's algorithm, according to the prioritization initially, the highest priority scheduled requests containing Emergent Dispatch Header. This header identifies the packet as emergency. In the case of multiple Emergent Dispatch Header packages or ordinary packages, the requests from mobile devices are served with higher priority. When many movable devices are available their requests are prioritized, using their movement speed. With higher priority are served the requests from faster moving devices. In the presence of multiple mobile devices moving at the same speed, the next criterion on which the requests are prioritized is the distance of the sensor to the coordinator. For this purpose, the principle of the Knowledge Based, Least Weighted Farthest Number Distance Product First mechanism is used. Higher priority has the packets sent by the sensors closest to the coordinator. When there are many sensors at equal distance to the coordinator has many sensors, the requisitions are prioritized using the sensor's type of application. With the highest priority served, the Healthcare applications and then Security and surveillance, Environmental monitoring, Animal tracking, Vehicle tracking, Agriculture and Smart Buildings.
The authors create a simulator, which is used for investigation influence of the proposed and standard sensor network, traffic prioritization algorithms on QoS.
Comprehensive comparative analysis of the traffic prioritization algorithm proposed by the authors is presented for 6LoWPAN and five others. For complex comparison of algorithms for traffic prioritization in 6LoWPAN a system of criteria proposed. Comparison of traffic prioritization algorithms mainly based on the results of Delay, Throughput, Packet Delivery Ratio and Packet Loss Ratio. This comparison made for a specific type of traffic, for certain end nodes.
The suggested by authors traffic prioritization algorithm in 6LoWPAN is better than others investigated, according to average arithmetic and average geometric complex estimations.

Г.7.4 Haka, V. Aleksieva and H. Valchanov, "Software Tool for Evaluation of Traffic Prioritisation Algorithms in 6LOWPAN Network," 2020 21st International Symposium on Electrical Apparatus & Technologies (SIELA), Bourgas, Bulgaria, 2020, pp. 1-4, doi: 10.1109/SIELA49118.2020.9167147.

This paper introduces enhancements to the simulation product for 6LoWPAN networks proposed by authors, which enables service quality research. The influence of different algorithms for prioritizing traffic on throughput, delay, packet delivery ratio and packet loss ratio considered. The included algorithms in the software tool are proposed by the authors algorithm and other classic algorithms for prioritization: First Come First Served (FCFS), Least Number of Sensors First (LNSF), Least Number of Hops First (LNHF), Least Number Distance Product First (LNDPF), Least Weighted Farthest Number Distance Product First (LWFNDPF). The software tool provides an interface for evaluation of proposed and classic algorithms on sensor networks.
In the proposed simulator the number of sensor nodes operating in a given region can vary up to 100, depending on the size of the area to be covered. Devices in this area may be fully functional or reduced functional. Fully functional devices can work both as coordinators and as end nodes, while those with reduced functionality only work as end devices.
A 6LoWPAN sensor network simulated with one fully functional device that serves the requests of the end devices. The purpose of the research is to determine the effectiveness of the algorithms embedded in the simulator for prioritizing traffic and in which situations, for which nodes they improve the QoS.
The results of the proposed prioritization algorithm show that the values for the studied parameters are better for the static devices closer to the coordinator. Prioritizing requests from nodes that are closer to the coordinator in sensor networks is important, because they are networks of multiple devices that transmit data constantly. This causes interference in the communication environment and errors, which initiates the resending of packets. As a result, communication load and delay increases, and degrade the QoS. With fewer devices, requests with highest priority served with more resources - from the nodes located up to 6 meters from the coordinator. This speeds up serving for these nodes, while freeing up resources to use for low-priority devices and offsetting delays. In case of insufficient resources, the requests of the lowest priority devices postponed for service in the next time interval.
The results for the mobile nodes according to the proposed prioritization algorithm show that the values for the studied parameters are better for the nodes moving at speeds above 3 m/s.

Г.7.5 D. Dinev, V. Aleksieva and H. Valchanov, "Study of Li-Fi Indoor Network Reliability", 2020 21st International Symposium on Electrical Apparatus & Technologies (SIELA), Bourgas, Bulgaria, 2020, pp. 1-4, doi: 10.1109/SIELA49118.2020.9167053.

An indoor test Li-Fi network implementation is proposed in this paper. The real network consists from three LiFi access points in the room for transmitting information at a distance of 2.5 m from the floor. Each of the LiFi devices is at a distance of 75 cm from the neighboring device. Maximum angle of illumination at which the devices transmit information 45o.
The goal is to implement a user equipment (UE) Handover between LiFi access points of the physically constructed network, taking into account correctly and incorrectly received data during this process. The handover realization in Li-Fi networks is very important for increasing the reliability of the network and transmitting all the data before leaving the network.
The following experiments were performed:
- to determine the health of the network;
- to perform a handover of users from one access point to another neighbor;
- reporting of correctly and incorrectly received data;

During the experiments, the light in the room was averaged 16.6 lx.
The following parameters were used for the first group of experiments:
- number of sent characters – 10 000;
- movement speed of the user equipment – 1m/s, 2m/s and 3m/s
- distance between the transmitter and receiver (L) – 0.5m, 0.8m, 1m, 1.2m and 1.5m;

From the results obtained through the experiments it can be seen that at a normal rate of 1m/s all the data sent by the transmitter were successfully received without any losses or erroneously received packets when passing from one access point to another. This is the case for each of the measured distances between the transmitter and the receiver. As the speed increases, an increase in the percentage of incorrectly received or not received data is observed.
The following parameters were used for the second group of experiments:
- number of sent characters – 100 000;
- movement speed of the user equipment – 1m/s, 2m/s and 3m/s
- distance between the transmitter and receiver (L) – 0.5m, 0.8m, 1m, 1.2m and 1.5m;

From the results obtained through the experiments it can be seen that at a normal rate of 1m/s almost all data sent by the transmitter are successfully obtained. The losses are due to the fact that for this speed the device has already left the network range and has not received the rest of the packets. As the speed increases, more and more incorrectly received or not received data are noticed.
The results show that as the speed of movement of the user device increases and the distance between the receiver and the transmitter, the percentage of erroneously received symbols increases. The speed at which there are no errors during the handover in this test Li-Fi network is 1m/s.

Г.7.6 Haka, V. Aleksieva and H. Valchanov, "Enhanced Simulation Framework for Visualisation of IEEE 802.15.4 Frame Structure on Beacon Enabled Mode of ZigBee Sensor Network," 2020 International Conference on Biomedical Innovations and Applications (BIA), Varna, Bulgaria, 2020, pp. 109-112, doi: 10.1109/BIA50171.2020.9244507.

This paper presents enhancements to the ZigBee network simulation product for IoT proposed by authors in previous research. The main improvements of the simulation software are: ability to calculate values for Received Signal Strength (RSS) and Received Signal Strength Indicator (RSSI); visualising the contents of the IEEE802.15.4 frame in beacon-enabled mode; study of classic algorithms for prioritising traffic in sensor networks; study of parameters affecting QoS such as Packet Delivery Ratio (PDR), Packet Loss Ratio (PLR), Delay and Throughput.
As an improvement to the simulation product for ZigBee network, one Knowledge Free and one Knowledge Based algorithm for prioritising network traffic implemented. Knowledge Free algorithms process requests in the order of their arrival. Such an algorithm for prioritising traffic is First Come First Served (FCFS). Knowledge Based algorithms use either application information, network information, or both to prioritise traffic. The implemented Least Number of Hops First (LNHF) algorithm bases on knowledge of the network information. According to this algorithm, requests from the devices closer to the coordinator are served with high priority.
The construction of a ZigBee network realizes using a graphical user interface, through which the coordinators are created and end sensor nodes added to them. Parameters such as: number of connected end nodes, channel bandwidth, region, frequency, beacon order and superframe order are set for each Personal Area Network (PAN) coordinator. To specify and link the created simulation with the restrictions for a certain region in the world, an option for selecting a certain channel and visualising the operating frequency added.
When the coordinator and end nodes correctly added with the appropriate configuration, the traffic generated by the end nodes in the network prioritised. When prioritising the traffic according to the selected algorithm, the contents of five IEEE 802.15.4 frames filled in.
The results for static nodes from the tests performed show that the LNHF algorithm improves QoS for end nodes, at a distance of up to 7m from the serving device. This will speed up the work, as the interference at these nodes is less, because the signal from the coordinator is better, respectively packet retransmissions will be less.
The results of the tests with mobile nodes for the considered prioritisation algorithms are similar. For the devices moving at medium speed, the allocated resources are few and the values considered deteriorate. This can worsen the QoS for these devices, as the handling of their requests will be delayed, and an additional delay will be caused by the initiation of a handover when the device is out of range of current coordinator.

Г.7.7 Haka, V. Aleksieva, H. Valchanov and D. Dinev, "Analysis of ZigBee Network Using Simulations and Experiments", 2020 International Conference Automatics and Informatics (ICAI), Varna, Bulgaria, 2020, pp. 1-4, doi: 10.1109/ICAI50593.2020.9311328.

This paper compares the results for the Received Signal Strength Indicator (RSSI) values from the end sensor nodes obtained by simulating a ZigBee network using the improvements to the simulation product presented and through a real ZigBee sensor network.
Graphical user interface of simulator allows adding the coordinators and end sensor nodes for construction of a ZigBee network. Once the ZigBee PAN coordinators have been added, the end sensor nodes can be connected to them. The values for RSS and RSSI calculated immediately after setting the sensor distance from the coordinator. The changes for all inserted parameters reflect in the data tables and can check from the "Nodes Table" tab. The simulator calculates automatically the values for RSS and RSSI, based on distance between the added end sensors and PAN coordinator. The calculated values represented by graphs according to node's distance to PAN or node's ID.
The tests for RSS and RSSI from the simulator were obtained after building a ZigBee network by one coordinator (ZigBee router) and 6 ZigBee sensor nodes connected in a star topology.
The physical construction of the ZigBee network is done with BeagleBone Black – BBB01-SC-505 board with Bone-Debian-7.8 operating system working as ZigBee Gateway, Texas Instruments (TI) transceiver - CC2531EMK and TI multi-standard sensor nodes – CC2650STK. The ZigBee Gateway configured using TI Z-Stack Linux Gateway. The CC2531EMK board configured to operate as a ZigBee transceiver and sensor nodes to operate in ZigBee network using CC-DEVPACK-DEBUG of TI. The data transfer and the receipt of the RSSI values from the end sensor nodes in the already built ZigBee network can be tracked, when a second CC2531EMK transceiver configured to work as a ZigBee sniffer.
The results for the obtained RSSI values from the constructed ZigBee network are inconsistent in the tests for 2, 4 and 6 sensor nodes. The results of 2 sensors show that with increasing distance from the coordinator the obtained RSSI values deteriorate. This trend is not observed in the tests with 4 and 6 sensor devices. In them, with increasing distance from the coordinator, the obtained RSSI values are identical or better for some of the nodes and worse for others. This is due to the presence of external noise influences and interference between the sensor nodes, which can increase with the number of devices in the network.
The obtained results show that for 2 end devices in the network the values for RSSI obtained through the simulator are almost identical to those for the tests with a real network. The results with 4 and 6 end devices obtained through the simulator are close to those of the real network. The deviation in the RSSI values of the simulator is about 10dB compared to the actual results

Г.7.8 D. Dinev, V. Aleksieva and H. Valchanov, "Simulation Framework For Studying Quality of Service Traffic Prioritization Algorithms in Li-Fi Network", 2020 International Conference Automatics and Informatics (ICAI), Varna, Bulgaria, 2020, pp. 1-4, doi: 10.1109/ICAI50593.2020.9311358.

This paper presents simulation software for studying QoS traffic prioritization algorithms in Li-Fi networks. In this framework are implemented algorithm proposed by authors from previous research, Wang traffic prioritization algorithm and two classic algorithms - First Come First Served (FCFS) and Least Number of Hops First (LNHF). The simulator calculates the packet delivery ratio (PDR), packet loss ratio (PLR), throughput and delay based on the allocation of resources through the implemented algorithms.
Proposed strategy for traffic prioritization distributes the resources in the terms of one timeslot, matches some criteria. The rearranging of the users connected to the Terminal according to the author's algorithm is the following: the users which are closer to transmitting point have higher priority and go higher in users table. If the distance from terminal to some of the users is equal, then the algorithm is looking for next criteria – is the client device mobile or static? The static devices have less priority. The mobile users have higher priority according to their speed. The higher the speed is the higher priority the device has. The type of requested service is the last criteria of the algorithm. Each one of them belongs to certain class which has different priority according to QoS parameters. There are four types of classes:

- Class 1 - contains services for Handover Calls, Link Recovery Calls and Voice Calls.
- Class 2 - contains Video Call services.
- Class 3 – contains services for Browsing, HDTV and Voice Messages.
- Class 4 - contains only Background Traffics services.

The new functionality includes availability for prioritizing connected users by implemented new algorithms, calculating their QoS parameters for PDR, PLD, Delay and Throughput, comparing the parameters by every algorithm and showing the transmitting matrix for each algorithm. The transmitting matrix for each algorithm can be shown after calculating and allocating the resources requested by connected devices. A new data table for storing the QoS parameter for each algorithm has been added. For easily comparing and studding the QoS parameters for every algorithm a graphic chart can be made for each of them.
The software realizes fully working simulation of Li-Fi network with terminal devices and connected to them users with their specifications. According to realized traffic priority and resource allocation algorithms the software can calculate the Packet Delivery Ratio, Packet Loss Ratio, Throughput and Delay which are important to providing better Quality of service.

According to those results a conclusion can be made that the algorithm which is proposed has better QoS indicator values than the others considered into this paper.

Г.7.9 Aydan Haka, Veneta Aleksieva, Hristo Valchanov, 6LoWPAN Network Analysis Using Simulations and Experiments, IOP Conference Series: Materials Science and Engineering, CIEES'20, vol. 1032 (2020), pp.1-4, doi:10.1088/1757-899X/1032/1/012015, ISSN: 1757-8981, https://iopscience.iop.org/article/10.1088/1757-899X/1032/1/012015

This paper presents the physical deployment of 6LoWPAN network and the study of throughput and end-to-end delay indicators, which compared with the results obtained through the 6LoWPAN simulation product presented in previous author's research.
The tests for throughput and end-to-end delay from the simulator were obtained after building a 6LoWPAN network by one coordinator and 6 6LoWPAN sensor nodes connected in a star topology. The coordinator configured to work on channel 25. Up to 6 end 6LoWPAN sensor nodes can be connected to the coordinator. All end nodes are static, perform the same type of application and located at the same distance from the coordinator (from 1m to 5m). The tests for reporting the values for throughput and end-to-end delay were made with 2, 4 and 6 sensor nodes connected to the 6LoWPAN coordinator. Once the coordinator and end node information added, a simulation performed to send a certain number of packets. After adding the packets to the send queue, the calculated values for end-to-end delay and throughput displayed.
The results of the conducted experimental studies are in large numbers, so they are summarised and presented in a table. Since the simulation product considers tests in ideal conditions, at different distances the values obtained are identical. The difference in the experiments performed is manifested in relation to the different number of sent packets.
The physical building of the 6LoWPAN network is done with BeagleBone Black – BBB01-SC-505 board with Bone-Debian-9.9 operating system working as 6LoWPAN Gateway, TI transceiver - CC2531EMK and TI multi-standard sensor nodes – CC2650STK. The data transfer and the receipt number of bits from the end sensor nodes in the already built 6LoWPAN network can be tracked, when a second CC2531EMK transceiver configured to work as a 6LoWPAN sniffer. This can be done on a Linux machine using Sensniff program for 6LoWPAN.
The experiments are made with 2,4 and 6 sensors with simulator and with real network in the same conditions. For example, the deviation in the simulated results with 6 sensors from the real ones for end-to-end delay is average 99% for 5, 10, 15 and 20 sent packets, and for throughput is 98% for 5 packets, 94% for 10 packets, 86% for 15 packets and 79% at 20 packets.
The results of real network tests are variable because the communication between the sensors and the coordinator is influenced by environmental factors such as electromagnetic interference, radio interference, packet transmission errors, other sources operating on the same frequency, interference between sensors, etc.
The obtained trend in the simulation results and real conditions are approaching, which gives reason to allege that the simulation product is suitable for purposes of education.

Г.7.10 Aydan Haka, Veneta Aleksieva, Hristo Valchanov, Deployment and Analysis of Bluetooth Low Energy Network, IOP Conference Series: Materials Science and Engineering, CIEES'20, vol. 1032 (2020), pp.1-4, doi:10.1088/1757-899X/1032/1/012016, ISSN: 1757-8981, https://iopscience.iop.org/article/10.1088/1757-899X/1032/1/012016

This paper presents the deployment of a physical Bluetooth Low Energy (BLE) sensor network for IoT and a study of the RSSI values derived from the end sensor units in the network.
The physical building of the BLE network done with RaspberryPi 4 Model B board with Raspbian operating system working as BLE master device, with built-in BLE transceiver and Texas Instruments multi-standard sensor nodes – CC2650STK.
The star topology by connecting the end sensor nodes and the master one was realised to examination the alteration in RSSI values. Different experiments with 1, 2, 3, 4, 5 and 6 static nodes performed, where for every one the nodes are located at distances from 1m to 10m from the master device. The examination of alterations in the received RSSI values for static sensor nodes located at different distances from the master device and for mobile nodes moving at different speeds done.
For 1 node the results show that as the distance of the sensor from the master device increases, the received RSSI values deteriorate. However the value at 10 meters is significantly better than the previous ones. Although only one device is transmitting on the communication environment which is not loaded, the decline in the previous values may be due to external sources of interference. The trend that at closer distance to the service device the obtained RSSI values are better is confirmed from the other tests with 3, 4, 5 and 6 sensors. The measured values for RSSI decline more and more when the distance from the master device and the end nodes number in the network increase.
Similar experiments were performed with mobile nodes. For the second node it is seen that the RSSI values are considerably lower. This is bred by the load on the communication environment and the emerged interferences. The trend when the sensors moving at a lower speed, the received RSSI values are better is confirmed from the other tests with 3, 4, 5 and 6 sensors.
Experimental results for the RSSI with static sensor nodes show that with increasing distance between the end nodes and the master device, the received values aggravates with considerable changes. Experimental results for the RSSI with mobile sensor nodes show that with increasing the speed of end nodes, the received values aggravate, but the change in the results is smoother.
For both static and mobile nodes sustained the tendency of aggravation of RSSI values with enlarging number of end sensor nodes in the network.

Г.7.11 A. Haka, V. Aleksieva and H. Valchanov, "Simulation Environment for Research of Algorithms for Traffic Prioritisation in ZigBee Network," 2021 17th Conference on Electrical Machines, Drives and Power Systems (ELMA), 2021, pp. 1-4, doi: 10.1109/ELMA52514.2021.9503088.

This paper presents a simulation environment that allows study the influence of the implemented algorithms for prioritising traffic on parameters related to Quality of Service (QoS) in ZigBee network.
Proposed traffic prioritisation algorithm for ZigBee is a modification of the previous proposed by authors algorithm and is intended to work in star topology. The product improvements are the ability to study the influence of different algorithms for prioritising traffic on parameters closely related to QoS, as well as visualisation of the built network topology. The simulation product has a modular architecture, and the operation of the individual modules is controlled by its core.
The algorithm checks several criteria for prioritising traffic in a ZigBee network. It is first checked for packages that are marked as urgent. In the presence of such packages, they are served with the highest priority. When there are multiple emergency packets or they are missing, the traffic is prioritised according to whether it is required by a mobile or static device. Requests from mobile devices are served with higher priority. When there are packets from more than one mobile device, requests are prioritised according to the speed at which the devices move. Higher priority requests are served from devices that move faster. Another criteria for prioritisation with the same others is the distance of the sensor from the network coordinator. Requests from sensors closer to the coordinator are served with higher priority. When the sensors are at the same distance from the coordinator, their requests are prioritised according to the value of cost. The requests with higher cost value are served with higher priority. Finally, the requests are prioritised according to the sensor application.
The performed experiments aim to study the influence of the implemented algorithms for traffic prioritisation in ZigBee network on the parameters PDR, PLR, Delay and Throughput, which are important to ensure good QoS. The presented experimental results show that with increasing number of nodes the service of the proposed algorithm for prioritising traffic in ZegBee network becomes even. However, higher values are provided for the studied parameters for the nodes closest to the coordinator. This will improve QoS and speed up service for these devices. This will free up the occupied resource faster and allow the lowest priority requests from the most remote devices to be served faster.
In contrast, the service of the classical algorithms is significantly even, which loads the entire communication in the network and can lead to deterioration of QoS. In addition, providing more resources from the proposed algorithm for serving requests from higher priority nodes, unlike the classic ones, will extend their battery life, as power consumption is only in active periods, and their number can be minimised by speed up serving.

Г.7.12 Aydan Haka, Diyan Dinev, Veneta Aleksieva, Hristo Valchanov, Comparative analysis of ZigBee, 6LoWPAN and BLE technologies for the Internet of Things, AIP, CIEES'21,25-27.11.2021, Rouse, Bulgaria, pp. 1-4, ISBN 978-073544375-4, DOI 10.1063/5.0099684

This paper presents the realisation of an IoT sensor network with Texas Instruments CC2650STK sensors, which can be configure and operate based on ZigBee, 6LoWPAN and BLE technologies. Experimental studies of the parameters End_to_End Delay, Throughput and PLR for the three technologies have been performed. Based on the results of the experiments, a comparison of the same between the considered technologies is presented. The aim is, as a result of the research, to formulate recommendations for the most appropriate technology for building a sensor network for IoT with the used sensor nodes.
The experimental studies for the considered technologies are realised with different number of simultaneously connected in the network static sensor nodes (2, 4 and 6). The experiments include calculating the values of the parameters End_to_End Delay, Throughput and PLR, which affect the QoS, at distances between the serving device and the sensor nodes of 1m, 2m, 3m, 4m and 5m, when sending 5, 10, 15 and 20 packets. In order to ensure comparability between the obtained results for the studied technologies, a star topology was used in all experiments.
According to the obtained results, the values for End_to_End Delay increase with the number of end nodes in the considered technologies, as more time is required to serve the requests of all devices. As the number of packets sent increases, so do the values obtained for End_to_End Delay, as there are more requests for serving on the network. With ZigBee in most experiments, the minimum and maximum value for End_to_End Delay is better than 6LoWPAN and BLE. In addition, in most experiments, the values obtained for ZigBee are constant and do not change drastic with increasing distance between the end nodes and the serving device.
From the obtained results for PLR it can be seen that the values increase in direct proportion to the number of nodes in the network for the considered technologies.
The following recommendations can be formulated from the experiments and the results obtained:
• In applications where it is important that the values for End_to_End Delay are relatively low and constant; it is better for the CC2650STK sensor nodes to be configured to work with ZigBee technology;
• In applications where constant throughput values are required, it is better for the CC2650STK sensor nodes to be configured to work with ZigBee technology;
• When required to provide higher throughput with a larger number of nodes in the network, it is better for the CC2650STK sensor nodes to be configured to work with BLE technology;
• In applications where less packet loss is required, it is better for the CC2650STK sensor nodes to be configured to work with ZigBee or BLE technology, as the PLR values obtained are extremely close, with lower values obtained for ZigBee.

Г.7.13 A.Haka, Y.Yordanov, V.Aleksieva, H.Valchanov, Simulation Environment for Bluetooth Low Energy Network , ICAI’21,30.09.-02.10.2021, Bulgaria pp. 287-290, doi: 10.1109/ICAI52893.2021.9639521 ISBN:978-1-6654-2661-9

Nowadays, with the expansion and improvement of communication technologies, the services offered are increasing, such as broadband Internet of Things (IoT) technologies, and one of the most common IoT technologies is Bluetooth Low Energy (BLE).
This paper presents a simulation product for the study of the communication and messaging between Master and Slave in the BLE network, which can also be used in education. It can be used both to study the basic functionalities of the technology and during onsite or online learning.
The developed simulator in the Department of Computer Science and Engineering at the Technical University - Varna has a modular architecture. When loading the application, the main functionality of the core is started, which is adding the Master device and realising its program logic for processing the incoming packets and their corresponding protocol data unit (PDU) type, as well as waiting for adding a Slave device and monitoring its status (Standby, Advertising or Connected). The execution of the main functionality is controlled by the "AppController" class.
To obtain statistical information about the time in which the end devices in the network were in a certain state, the core turns to the Statistics module, which is managed by the class "DeviceStatisticsUtil". The processed information through the various modules is visualised through the built graphical user interface (GUI).
After adding Slave devices, each of them can be allowed to visualise the distance to the Master, removed from the network or change its status from Standby to Advertising. When the state of Slave is Advertising, it starts sending advertising packets on the channels intended for this (37, 38 and 39). With this, the Slave sends broadcast packets on the communication medium so that it can be detected by the Master in the range, and possibly connected to it. When switching to Advertising mode, the tracking of the packets transmitted on the communication medium also starts.
In order to compare the exchange of messages when establishing a connection, sending data and terminating the connection between Master and Slave devices in the BLE simulator and a real environment, a real BLE network is configured. To ensure comparability between the results of the real and simulated BLE network, an experimental topology of one Master and one Slave was realized.
During the simulation, some of the details of the communication were omitted in order to simplify the process in consideration and facilitate its presentation during learning. The simulator represents the main messages in the implementation of the process, which allows it to be used during the learning both on site and online.
The results show that the simulator can be used to present the highlights of the communication between Master and Slave.

Г.7.14 D.Dinev, V.Aleksieva, H.Valchanov, K.Genov, Simulation Software For Finding Best Route in LoRaWan Network, ICAI'21,30.09.-02.10.2021, Bulgaria pp. 291-294, doi: 10.1109/ICAI52893.2021.9639718 ISBN:978-1-6654-2661-9

LoRaWan is a long-range, low-power, low-bitrate, wireless telecommunications system, promoted as an infrastructure solution for the Internet of Things: end-devices use LoRaWan across a single wireless hop to communicate to gateway(s), connected to the Internet and which act as transparent bridges and relay messages between these end-devices and a central network server.
This paper presents simulation software for finding best route in LoRaWan networks.
Depth First Search is an algorithm for crawling or searching in data structures such as "tree" and "graph". To implement the algorithm, a vertex or node of the structure is selected, which is denoted as a root and the crawl starts from it. All subsequent peaks are visited sequentially in depth until reaching one, without heirs, after which a search is performed with backtracking until reaching a new end point or after the complete crawl - to the root. The original version of the algorithm was created in the 19th century by Charles Pierre Tremo to solve maze problems.
The simulator uses a modified version of the algorithm, which searches all paths only to one end device defined as a destination, to find all possible routes from a particular Personal area network to another network.
There can be several routes in a multi-hop networks with the same parameters. Initially, using Hassle Free Route the route is selected according to the parameter for the shortest path. Authentic value is included to prioritize the routes. The authentic value is stored in the routing tables of the devices as a negative, positive number or 0, where:

- negative number - there is a large loss of packets along the route;
- positive number - the route is smooth;
- 0 - default value; the route has not been evaluated.

A higher positive value indicates that the route is more successful than the others. It indicates the number of successful broadcasts on this route. For each successful transmission, the authentic value is increased by 1, and for each unsuccessful transmission, it is reduced by 1. When a fragment contains an emergent dispatch header, it is forwarded to the route with the highest authentic value. These fragments are prioritized and sent in the most preferred way.
The average time to send a 51 byte LoRaWan fragment is Ttrans = 6 ms, which includes the transmission time and the back-off timer.
The simulation framework has 5 main blocks- GUI, Core, Creating topology, Topology modification, Finding best path between end devices.
To make the test about finding the "Best Route" between end devices a LoRaWan network with 5 terminal devices and 4 end devices is made. Tests for finding best route between end devices are made and proved that the proposed simulator is fully functional and suitable for LoRaWan networks researches.

Г.7.15 D.Dinev, V.Aleksieva, H.Valchanov, Comparative Analisis of Li Fi Simulators for Purposes of the Education, ICAI'21,30.09.-02.10.2021, Bulgaria pp. 125-128, doi: 10.1109/ICAI52893.2021.9639691 ISBN:978-1-6654-2661-9

The Internet of Things Li-Fi technology provides high speed, bidirectional and secure wireless access. This requires a research into the quality of service of this technology. This can be done by using simulation software that will minimize the cost and time for building such networks. This paper presents information about comparative analysis between author's proposed simulator and some of the most well-known simulators (OptSim, Veins VLC, NS-2,NS-3, MATLAB) to explore the quality of service on the Li-Fi Internet of Things networks. A system of criteria for performing a comparative analysis of simulators for the Li-Fi network is proposed. This approach to Li-Fi network research for IoT can also be introduced in education.
Existing simulators of Li-Fi networks have a number of disadvantages related to their operation and functionality. Here they are presented.
The proposed criteria for comparison of the LiFi simulators are:

- Modeling of different algorithms for traffic prioritization in LiFi
- Modeling of different methods for resource allocation
- Simulation of mobility
- Maintaining a GUI
- Visual presentation of the studied network
- Analysis of the obtained results
- Easy installation
- Scalability
- User/developer manual
- Programming language
- Memory usage
- License

According to the presented results of the study, due to the wide range of considered criteria, the most suitable for the study of Li-Fi networks are MATLAB, OptSim and NS-3. However, a separate study of the criteria shows that the proposed simulator by the author provides better values for the indicators: "Easy installation", "Memory in use" and "License for use", which are extremely important for educational purposes.
The criteria for comparison do not specify the criteria "Visualization of the transmission matrix", i.e. unlike the simulator suggested by the authors, none of the other simulators provides this capability. It provides comparable, with other simulators, results against many other criteria. This proves that authors' simulator is very suitable for training and educational purposes.

Г.7.16 A.Haka, V.Aleksieva, H.Valchanov, ZigBee Simulation Framework for Studying the Formation of a Hierarchical Tree Topology , ICAI'21,30.09.-02.10.2021, Bulgaria pp. 257-260, doi: 10.1109/ICAI52893.2021.9639563 ISBN:978-1-6654-2661-9

The ZigBee is one of the advanced technologies for managing of the IoT sensor networks, because it provides a high Quality of Service (QoS) and low power consumption. One possible solution to achieve the better QoS in these networks is to use an efficient traffic routing algorithm. This paper presents an improved simulation framework in which an algorithm for forming a ZigBee hierarchical topology based on priorities, allowing hierarchical routing, is implemented. The simulation framework provides an opportunity to analyze the results of the algorithm through a visual interpretation of the network topology.
ZigBee uses a mixed routing mechanism combined with hierarchical tree routing protocol (HRP) and ZigBee ad hoc on-demand distance vector (Z-AODV). HRP is an active routing method whose route information is established when the network is deployed and remains unchanged, except when the network structure changes.
In the simulation framework, the authors' algorithm for the formation of an energy-balanced ZigBee network based on priorities with a tree topology has been implemented. The ZigBee topology consists of a single coordinator (the tree root), multiple routers (branches) and end devices (leaves). In this algorithm, the pricing method is used to achieve the goal. In the algorithm, it is assumed that the routers serve only to build the topology, and do not function as end devices. Each router and end device has a willingness to pay value - the priority for end devices and an energy level for routers. The coordinator and routers have a charging rate value - a price that must be paid by the end nodes to connect to them. Therefore, the higher the value for willingness to pay, the higher the priority has the end device.With routers, the case is similar, the higher the value of willingness to pay, the more energy there is.
The developed simulation framework has a modular architecture. Simulating a ZigBee network requires working through two main windows. One of them for adding parameters for the coordinator and the other for routers and end nodes in the network.
The visualization of the topologies from the performed experiments for the implemented algorithm for forming a hierarchical topology in ZigBee network shows that with increasing number of routers the depth of the hierarchy in the constructed tree increases. In terms of energy balance, the algorithm for forming the hierarchy ensures that the routers with more energy are arranged at a lower level (closer to the coordinator). This provides better energy efficiency for routers, as more devices will be connected to those with more energy and fewer devices to those with less energy.
Experiments show that the implemented algorithm allows the construction of a balanced in terms of energy efficiency hierarchical tree topology.

Г.7.17 Yuri Dimitrov, Veneta Aleksieva, Hristo Valchanov, Comparative Analysis of Prototypes for Two Touch Finger Interfaces of Smartwatch, IOP Conference Series: Materials Science and Engineering, CIEES'20, vol. 1032 (2020), pp.1-4, doi:10.1088/1757-899X/1032/1/012019, ISSN: 1757-8981, https://iopscience.iop.org/article/10.1088/1757-899X/1032/1/012019

The paper presents a comparison study of the proposed prototype for two fingers touch interface for smartwatch with two other prototypes.
In order to be observed the touch areas on the bezel, a dedicated 3D model is designed and printed. It is as close as possible to real smartwatch according to its size and form.
The first step is to choose the dimensions of the 3D model. The diameters of the real smartwatches vary between 34.5mm and 58mm, but 73% of them are between 42mm and 46mm. The height of the real smartwatches depends on kind of functions, but vary between 10,9mm and 16mm. Almost 80% of them are between 14mm and 15mm. Based on this statistics, the chosen dimensions of the 3D model are - diameter: 44mm, height: 15mm, bezel angle: 450, bezel width: 4mm; color: white; material: PLA.
The second step is to evaluate in which two separate and distinguished sectors on the device bezel is possible to be registered in order the device interface to be activated and further interface actions to be performed. The detailed results from this evaluation are presented from authors in other research.
The third step is to activate some functions with this touch sensible bezel prototype and to compare its functionality with a prototype with buttons. The authors made this comparison in other research and the main conclusion is: the touch prototype outperformed the prototype with buttons in speed of operations, especially when the set of the interface commands is longer.
The final step is to evaluate the prototype in comparison with the similar prototypes. In order to perform a comparative analysis of the author's prototype with others, the same criteria for evaluation are used. In Oakley's prototype, physical contact with the device edge was integral to the vast majority of inputs – only 17% are with two fingers. Participants also favored their dominant hands and use of their thumb and index fingers. The authors gave the results for eight ordinal directions, but in this comparison only results for matching direction is used. In Yeo's prototype the authors use multiple fingers to move the whole prototype, which is with a typical smartwatch size. They showed that their prototype is competitive with commercial smartwatches of this size while input events are generated responsively (55-61ms) and accurately.
The experimental data for authors' prototype are presented in comparison with Oakley's prototype and Yeo's prototype in the table. According to the results obtained for the complex evaluation, the prototype for two fingers touch interfaces for smartwatch proposed by the authors is better than two others. The main advantage of the proposed prototype is its standard size and less touch time in long sequence of touches.

Г.7.18 Y. Dimitrov, V. Aleksieva and H. Valchanov, "Method for Body Pose Recognition based on Two-Finger Touch Bezel on Wearable Device", 2021 17th Conference on Electrical Machines, Drives and Power Systems (ELMA), 2021, pp. 1-5, doi: 10.1109/ELMA52514.2021.9503001.

The purpose of the presented study is to propose a method for recognizing the pose of the user (lying, sitting, standing) when waking a wearable device from sleep mode, and the device interface to be activated in order to visualize information and execute some commands due to deliberate touching of the bezel on a wearable device with two fingers by the user.
If the pose is successfully recognized when activating the wearable device interface, quick access to features and applications in the context of the pose (those most likely to be performed by the user) will be offered. This will reduce the active mode time of the device, which will extend the period between two charges of its battery. Body pose recognition can be combined with other factors such as the time of a day - for example, in a lying position in the evening to offer quick access to some functions and applications, and in the morning, again in the same position to offer the user quick access to other functions/applications. Another factor may be a previous pose and/or activity - for example, in a standing pose immediately after getting up to offer quick access to some functions/applications, and in the same pose, but after a long run, to offer others. Thus, pose recognition when activating the interface of a wearable device will reduce the time to work with it, which will lead to a longer period between two charges of its battery. Deliberately touching the device activation bezel with two fingers cannot be recognized by any other action and an unsolicited quotation of the device may occur.
The recognition of the pose of the user's body is based on the relative difference in the position of the fingers on the bezel when activating the interface by him in the different positions of his body when using a wearable device. For this reason, it is not necessary to measure the angles in the same position for different users, as well as to determine specific areas of the bezel to determine the position of the body. It is sufficient for each device / user to establish (after starting to use the device) the different areas of contact when activating the interface and on the base of these differences to predict in what position the user's body is most likely at the time of activating the interface.
The experimental group consists of 10 people, all with a leading right hand, all participating voluntarily in the experiment. There are made 300 attempts - 100 for each body position.
Based on the results of experimental studies, it can be assumed that the proposed method for determining the pose of the user's body on a wearable device based on the location of the fingers of his leading hand on a touch-sensitive bezel on a wearable device, is effective and applicable.
Въз основа на резултатите от експерименталните проучвания може да се предположи, че предложеният метод за определяне на позата на тялото на потребителя на носимо устройство въз основа на местоположението на пръстите на водещата му ръка върху сензорна рамка на носимо устройство, е ефективен и приложим.

Г.7.19 A. Haka, V. Aleksieva and H. Valchanov, "A Comparison Study of Decisions for Computer Network Laboratory in Distant Learning Education", 2021 17th Conference on Electrical Machines, Drives and Power Systems (ELMA), 2021, pp. 1-4, doi: 10.1109/ELMA52514.2021.9503059.

This paper presents a comparative analysis of the investigated solutions for distance learning in computer network subjects.
Compulsory social isolation during a pandemic posed new challenges for the system of education. The indispensability quickly moves to a remote form of education necessitated the use of a different educational approach. During a full lockdown, three approaches for training in classes related to computer networks were used and studied - simulation products, real computer network with remote access and virtual computer network with remote access:
To use simulation products as Packet tracer, GNS3 etc.
In the Department of Computer Science and Engineering at the Technical University-Varna is developed from authors a real computer network laboratory with remote access. The remote access is implemented through web management system, developed by authors. Citrix XenServer has been chosen as a virtualization platform, which has high performance, easy maintenance and is free to use. The main idea of laboratory design is to create a snapshot of the virtual machine (for the respective operating system) for each of the computers, using Xen's snapshot capability.
To achieve high flexibility and to avoid some disadvantages of a previous solution, an experimental virtual infrastructure has been implemented. It is based on two Sun Fire Z20 server machines connected to a 1G Ethernet network and using VMware ESXi. The choice of VMware Infrastructure 3 is dictated by its capabilities for multiprocessor support, dynamic balancing and resource allocation between virtual machines, as well as migrating virtual machines between individual servers without interrupting their operation. Based on the virtual infrastructure, a number of virtual machines with respective operating systems have been launched. The virtual infrastructure can be accessed with the VMware vSphere Client software.
The goal of the study is to evaluate which solution is more appropriated for distance learning in the student courses related to computer networks. A system of criteria for evaluating the studied solutions is developed, according to the challenges in online learning.
The comparison is based on a proposed by the authors system of criteria, consistent with the challenges of distance learning. In order to ensure objectivity in the comparison, a complex assessment of the considered approaches is performed, based on complex arithmetic estimation. According to the results from an average arithmetic estimation, the most appropriated solution for distance learning is determined to be using a virtual network infrastructure.

Г.7.20 Haka, A., Yordanov, Y., Aleksieva, V., Valchanov, H. Study of Received Signal Strength Indicator values of Bluetooth Low Energy in Test Environment and Simulation, ICAI 2022, pp. 282–286, ISBN 978-166547625-6, DOI 10.1109/ICAI55857.2022.9960009

This paper presents a study of RSSI values under BLE technology in real network and simulation. The research aims to compare the results obtained in a real environment and simulation to formulate recommendations for the situations in which equipment from a specific manufacturer and the simulator can be used. The influence of the number of end nodes and the distance between them and the central one on the strength of the received signal in a real environment as well as in simulation is considered to determine the reliability of the simulated values and their applicability in research and training.
For experimental purposes, to study the RSSI values of BLE technology in a real environment, two BLE networks are connected with end nodes from two different manufacturers – TI and Arduino. The components of one BLE network are: Raspberry Pi 4 Model B with built-in BLE transceiver and sensor nodes CC2650STK from TI. The role of the master BLE device is performed by the Raspberry Pi 4 Model B board, on which the Raspbian operating system is preloaded. The CC2650STK end sensor nodes are pre-programmed to work with the BLE standard. The components of the other BLE network are: an Arduino nano 33 IoT main unit and Arduino nano 33 BLE sense sensor nodes from the Arduino company. The main BLE device is implemented with an Arduino nano 33 IoT board, which is configured to run the program described in pseudocode Code 1. Several Arduino nano 33 BLE sense boards are used as end sensor nodes.
The experiments obtained on a BLE network with end devices from the manufacturer TI show that the reported RSSI values mainly vary in the range from -40dBm to -70dBm, with the exception of the experiments with 4 and 5 simultaneously connected end nodes. In these experiments, the reported RSSI values are increased for the nodes that are located farthest from the Master device. The reported values range from approximately -72dBm to -85dBm.
The RSSI values in the experiments in a BLE network with Arduino devices vary mainly in the range from -80dBm to -95dBm, with the exception of the experiments with 5 and 6 simultaneously connected end nodes. In these experiments, the reported RSSI values are increased for the nodes that are located after 3 meters from the Master device. The reported values range from approximately -80dBm to -105dBm.
A comparison was made between the experiments carried out under the same conditions. Based on the presented results, recommendations are formulated for the cases in which the considered end nodes and the simulator can be used.

Г.7.21 Haka, A., Dinev, D., Aleksieva, V., Valchanov, H. Internet of Things Sensor Data Storing Systems for Educational Purposes, CIEES’22, 2022, Electronic ISBN:978-1-6654-9149-5, DOI: 10.1109/CIEES55704.2022.9990805

This paper presents two proposed and developed data storage systems of LoRa and ZigBee sensor networks that have wide application in the field of IoT. The systems were created in the Department of "Computer Science and Technology" of the Technical University - Varna for use in training. By developing a web interface, they provide remote monitoring of data stored in a non-relational database. The systems allow students to familiarize themselves with the configuration and operation of two of the most widely used IoT and home automation standards. They also allow students to study the workings of the MQTT protocol. In addition, they enable students to consolidate and acquire new knowledge and skills in the field of database management and programming.
For the purposes of training in the KNT department, at the Technical University - Varna, a LoRa system for storing sensor data for IoT with a Web-based interface for work with the possibility of remote access has been developed. It is not necessary to generate special identifiers to configure the system. The system consists of three main components Dragino LG01-S - Single Channel LoRa IoT Gateway for managing the network and receiving data from sensor nodes, an MQTT server for receiving and forwarding data from the LoRa Gateway and a MongoDB database for storing the received information. The LoRa Gateway needs to be configured to forward the received messages from the sensors to an MQTT server. The MQTT messages from the corresponding channel are then passed to the MongoDB database, where they are stored in the corresponding collection. Passing information from an MQTT server to MongoDB is provided with Python scripts. The developed Web interface provides appropriate visualization of the stored information and output of statistical samples based on the individual characteristics and time segments of work, as well as creation of a model of operation of the devices in the studied environment. Advantages and disadvantages of existing ZigBee data storage solutions as well as the developed one are presented. According to the presented information, the developed solution includes some of the advantages of the existing ones and overcomes most of their noted disadvantages. The developed solution has shortcomings in the ability to provide a working environment for multiple users, limited storage for storing information and sharing information. The developed solution allows working with various IoT technologies, and the presented shortcomings can be overcome by improving the system. This shows that the developed system is suitable for learning purposes in a university environment.

Г.7.22 Haka, A., Dinev, D., Aleksieva, V., Valchanov, H. A Study of ZigBee Networks in Experimental Environment and Simulation, CIEES'22, 2022, Electronic ISBN:978-1-6654-9149-5, DOI: 10.1109/CIEES55704.2022.9990742

This paper presents an investigation of End_to_End Delay, Throughput and Packet Loss Ratio (PLR) parameters that affect QoS in one of the most commonly used IoT technologies – ZigBee. The conducted research presents results for the considered parameters based on simulation and a real ZigBee network with end nodes of the manufacturers Texas Instruments (TI) and Sonoff. Based on the calculated values, a comparison was made between the results, aimed at formulating recommendations for the selection of devices for building a ZegBee network, according to the specific QoS requirements in the network, as well as determining the application possibilities of the considered simulation product.

The experimental studies on the considered ZigBee devices were carried out with different numbers of static end nodes simultaneously connected to the network (2, 4 and 6). In each experiment, the end nodes are located at different distances from the master device (1m, 2m, 3m, 4m, and 5m), and at each of the distances, 5, 10, 15, and 20 packets are sent from each node. The connection topology for the conducted experiments is star.

The ZigBee network with end nodes from the manufacturer TI is built with a BeagleBone Black (BBB) - BBB01-SC-505 board installed with Z-Stack Linux Gateway software to act as a ZigBee coordinator. Connected to the BBB board is a TI CC2531EMK transceiver configured to transmit and receive ZigBee signals. The end devices are TI CC2650STK configured to operate as ZigBee nodes.

The ZigBee network with end nodes of the manufacturer Sonoff is built with ZigBee2MQTT software installed on a Windows PC, which implements a ZigBee coordinator. A TI CC2531EMK ZigBee transceiver is connected to the Windows PC to communicate with the end nodes, Mosquitto server and Nodejs are also installed to read the data from end nodes. The final devices are the Sonoff Door Sensor and the Sonoff Temperature and Humidity Sensor.

The ZigBee network through the simulator is built entirely virtually with coordinator and end node parameters based on published standards.

Based on the conducted experiments and the summarized results, the following recommendations can be formulated:

• For ZigBee applications where significantly low and constant End_to_End Delay values are required, it is better to use TI devices in the network;

• For ZigBee applications where higher Throughput values are required, it is better to use TI devices in the network;

• For ZigBee applications where it is required to maintain low PLR values, both TI and Sonoff devices can be used in the network;

To study the direction and trend of change of the investigated QoS parameters in different situations, the considered simulation environment can be used, which is also suitable for use in training both in-person and in remote form.

Г.8. Публикации в нереферирани списания с научно рецензиране

Г.8.1 Вълчанов Х. Виртуализирана мрежова лаборатория. Национална конференция по е-обучение във висшите училища, Русе, 2014, 159-164.
Valchanov H. Virtualized networking laboratory. National conference on E-learning. Rousse, 2014, pp.150-164.

Building and maintaining laboratories for training in computer networks and Internet technologies is a practically complex task. Users should be able to build different network topologies as well as have full administrative control over the devices. At the same time, before carrying out practical activities and research, students must build the relevant experimental setup. This takes time, and on the other hand, wrong connections are possible, which can lead to equipment damage.
The essence of the proposed approach is the use of available equipment in an existing computer network laboratory, with each of the computers configured to perform a certain role according to a corresponding network scenario. For this purpose, a characteristic feature of virtualization platforms is used - the so-called snapshot, which is an up-to-date copy of the current state of the virtual machine. The current copy contains an IP configuration and a set of started services that define the role of the corresponding machine in the virtual infrastructure. By means of the up-to-date copies of the virtual machines, a quick switch between different infrastructures is implemented, and within a short time interval, the requested functioning virtual infrastructure is built.
The virtualization platform used is KVM (Kernel-based Virtual Machine). The choice of KVM is dictated by the following considerations. KVM is tightly integrated with the virtual hardware emulator QEMU (Quick EMUlator), which provides capabilities to create up-to-date copies of the state of machines. Management is implemented using a web interface based on PHP and AJAX. Parallel AJAX requests to the web server are used to speed up the reconfiguration process.
Research is presented on the actions required to build and deploy a particular network topology. The results show that the duration of the process of commissioning a functioning network infrastructure for a specific task is about 46 minutes. This represents almost half of the allotted laboratory time. As a result, the learning material cannot be absorbed by the students in the necessary volume - this reflects in a decrease in the quality of the learning process.
The implementation of the virtualized network laboratory makes it possible to eliminate these problems. The construction of a specific network topology is done within 1 minute Students receive a built and functioning infrastructure, which allows them to focus their efforts directly on solving the specific practical task. This reflects both in the success rate in solving tasks and in increasing the quality of the students' learning process.

Г.8.2 Вълчанов Х. Прехвърляне на мултимедиен трафик през WAN мрежи. Proc. of Int. Conf. Automatics and Informatics'14, Sofia, 2014, pp.I-167 – I-170. ISSN 1313-1850
Valchanov H. Transfer of multimedia traffic trhough WAN networkds. Proc. of SAI 2014. pp.I-167 – I-170. ISSN 1313-1850

Multi-streaming at the transport level is the ability of transport protocols to support different data streams, providing each stream with an independent sequence of data delivery. The Stream Control Transmission Protocol (SCTP) is a standard reliable transport protocol providing multi-stream data transmission.
The presented report proposes an approach that consists of using existing software solutions (HTTP servers and clients) operating on the basis of the TCP protocol, and transmitting the data through WAN networks using the SCTP protocol. The motivation for the proposed solution is based on the fact that servers and clients are located in local networks characterized by high throughput. These networks are typically connected using low-speed WAN technologies, having an order of magnitude less bandwidth. With the proposed approach, the full capabilities of SCTP can be used for single-stream exchange.
The report presents some features of the TCP/SCTP proxy server implementation. The internal structure of the proxy server includes four separate modules:
• The communication control module maintains information to determine the correct socket on which the received packet is delivered.
• The TCP session manager is based on the Linux system library. It creates and terminates sessions and controls data exchange with packet loss detection and network congestion reporting.
• The SCTP session manager is based on the sctplib library. It arranges the packets within a stream - for each individual stream, the module monitors the sequence of receiving the data. Data loss in one stream does not affect other streams. At the same time, it provides for the creation and termination of associations - it includes mechanisms for creating, normally closing and terminating an association.
A test infrastructure was built, including WAN connections, on which groups of experiments were conducted. Two groups of experiments were performed - file exchange and web page loading. The tests were done at different speeds of the serial connections between the routers - 64000 bps, 115200 bps and 128000 bps. Web page loading experiments show better response times when using the SCTP protocol. These results are achieved thanks to SCTP's common data flow control mechanism. When downloading a data file, the SCTP stream management mechanism is not efficient because only one stream is used. In contrast, the TCP protocol only needs a single session to transfer the file.
A goal of future work is the development of a multi-threaded proxy server architecture.

Г.8.3 Valchanov H., M. Angelov. Improving Performance of Multimedia Web Transfer over WAN Connections. Proc.of the ICEST 2014, Nis, Serbia, v.1, 27-30. ISBN 978-86-6125-108-5.

The paper presents an approach to improve the performance of data transmission between networks based on WAN technologies. The architecture and internal details of the SCTP web proxy are given. The main idea of the proposed approach is that the presented TCP / SCTP proxy server will work as an interface between TCP and SCTP protocols, allowing web browsers and servers to take advantage of SCTP capabilities without having to change their code. Since the purpose of the present study is to explore the capabilities of the approach, the structure of the proxy server is simplified, providing only the basic functionality needed to conduct the study.

The functionality of the proxy server is implemented on the basis of a dual-stack model. Dual stack uses two transport protocols – TCP and SCTP, thus allowing easy integration of the proxy server into TCP/IP infrastructures. When a TCP request is received from a client (web browser), the proxy acts as a TCP server. When it forwards this request to another SCTP proxy, it acts as an SCTP client. Similarly, when receiving a response from a TCP server (web server), the proxy acts as a TCP client, and when returning the response to SCTP, the proxy acts as an SCTP server. The Proxy Core component implements the core functionality of a proxy server.

The performance analysis of the proposed approach is done on a network infrastructure representing slow WAN links. The test environment includes Cisco routers 2901, VLAN Cisco Catalyst 2960 switches, computers HP Desktop 500B CPU Intel Core Duo E5800 3.2 GHz with 2G RAM. The web server platform is based on Slackware Linux 2.6 and gcc 4.4.3. Windows 7 operating system and Google Chrome 28.0.1500 browser are used as the client.

Two test groups were conducted – file transfer and web page loading. Experiments were conducted at different speeds of the serial connections between the routers - 64000 bps, 115200 bps and 128000 bps.

The obtained results show that the proposed approach is fully functional and applicable for the transmission of multimedia data over low-bandwidth networks.

The goal of future work is to develop a multithreaded proxy server architecture. This will allow him to serve multiple clients at the same time. Another future work is to add data caching and traffic filtering capabilities.

Г.8.4 Вълчанов Х. Подход за мултимедиен Web трансфер през нискоскоростни глобални мрежи. Proc. of UNITECH'14, Gabrovo, 2014, pp.II-213 – II-218. ISSN 1313-230X
Valchanov H. An approach for multimedia Web transfer trough low speed global networks. Proc. of UNITECH'14, Gabrovo, 2014, pp.II-213 – II-218. ISSN 1313-230X

The use of the multi-streaming protocol SCTP as the transport protocol for HTTP. can solve a number of problems in the currently used multimedia document transmission model in WAN infrastructures. Due to the fact that multimedia documents consist of objects of different types and sizes, multi-streaming allows them to be sent in partial order rather than in strict sequence. As a result, the visualization of the pages when loading in the browser is improved. At the same time, transport takes place within an association, thus all flows use a common mechanism to manage data exchange. This, in turn, significantly reduces system costs at the transport level.
In the presented paper, a different approach is proposed, which consists of using existing software solutions (HTTP servers and clients) operating on the basis of the TCP protocol, and transmitting the data through WAN networks using the SCTP protocol.
The report presents a TRCP/SCTP proxy server with a modular architecture.
The purpose of the conducted experiments is to investigate the behavior of the two transport protocols in network infrastructures built on the basis of WAN technologies. These types of technologies provide throughput that is orders of magnitude less than the technologies used in local area networks. This is an important factor that globally affects the response time of web servers.
An experimental network infrastructure was built, with serial connections based on the High-Level Data Link Control (HDLC) protocol established between the routers. By configuring them with different speeds, a transport WAN environment with different throughput is simulated. The experiments were conducted in two directions:
• direct communication between web client and server, based on the TCP protocol;
• indirect communication between web client and server, realized through TCP/SCTP proxy server.
The obtained results show that in web page loading experiments, better response times are observed when using the SCTP protocol. Loading a web page is done within a single SCTP session (in contrast to TCP, where a new session is created for each resource).
When downloading a data file, the SCTP stream management mechanism is not efficient because only one stream is used. In addition, it should be noted that both transport protocols ensure reliable data delivery without loss of information.

Г.8.5. V.Aleksieva, H.Valchanov, T. Dlugosz, R. Wrobel. Real efficiency of SoHo routers with alternative software. Telecommunication review+Telecommunication news Tele-Radio-Electronics, Poland, v.1, pp.10-12, 2014, ISSN 1230-3496.

The article presents an analysis of the performance of three popular models of SoHo (Small Office Home Office) routers. The authors present an overview of the development of SoHo networks in Bulgaria and Poland. The performance of these three models is tested based on bandwidth for UDP and TCP protocol in three topologies.
The results are presented in tables and graphs, which allows their comparison. Data transfer rate is one parameter that was studied. It is not only the price and additional features that influence the choice of SoHo routers. The most important selection factors are the topology and the type of data transferred. As a result of the study, the most productive is the TP-Link WR1043ND router.

Г.8.6 H.Valchanov, S.Andreev. Muti-threaded user and kernel-space library. Proc. of the ICEST2015, Sofia, 2015, pp.208-211

The development of technologies and the large range of possibilities offered by modern hardware allow the use of specialized high-performance approaches for the implementation of various software systems and algorithms. One of the most used and effective approaches is to create multi-threaded software running in parallel on multiple processors. This paper presents the specifics of a multi-threaded library implementation under Linux that allows operation in both user-space and kernel-space.
In user mode, each thread is represented by a special structure (Thread Control Block - TCB), containing the necessary information for its management. This information is used by the dispatcher for scheduling and context switching. The dispatcher takes care of event management, which is very important for the proper functioning of the library. The dispatcher is called whenever a thread grants the CPU to another thread or when it blocks automatically on an event.
The implementation of threads in this mode is based on the clone() system call. For the Linux kernel, a thread is stored in the same structure that is used for an individual process. Although the underlying information is stored in operating system structures, it is still necessary to maintain thread data in the user context as well. Such information, for example, is about the initialization function and its argument. Maintaining this duplicate information allows for simpler implementation of some library functions.
Experimental studies have been conducted to test the efficacy of both implementations by performing computational and I/O intensive tasks. A comparison is made with the pthreads library.
For benchmarks, as a result, pthreads and the kernel-space implementation perform best on the test. The results are close, but the advantage is for the pthreads library. This test shows the big drawback of user-space libraries – they can only work on one processor.
In the I/O tests, the pthreads library and the kernel-space implementation show the same performance with a slight advantage of the kernel-space implementation.
Overall, the results show that identical and in specific cases high performance is achieved for the presented library. The developed library is relatively small, resulting in faster compilation. The code is written in a simple way that allows the library to be used for learning multithreaded programming.

Г.8.7 Х.Вълчанов. Хибридна многонишкова библиотека. Сб. на международна конференция "Автоматика и информатика'2015, София, 2015. 157-160. ISSN 1313-1850
H. Valchanov. A hybrid multithread library. Proc. of SAI 2015, pp. 157-160. ISSN 1313-1850

Multithreading is a widely used modern approach to increase computing performance by introducing parallelism in program execution. This paper presents some aspects of the implementation of a hybrid multi-threaded library under Linux, which enables operation in both user-space and kernel-space. The library provides an implemented user interface that is maximally identical for implementations in both modes.
An implementation in user-space mode is presented in which a non-preemptive dispatching scheme is used. The different thread states in this mode are shown. Non-blocking read/write operations, timing thread sleep, synchronization primitive operations, and thread-to-thread binding functionality are all based on the Linux event system.
In the kernel-space mode implementation, thread scheduling and dispatching uses system facilities provided at the operating system level. In the current implementation of the library, two types of synchronization primitives are developed - spinlocks and mutexes. With spinlocks, synchronization is based on active waiting, which takes up processor cycles. From the point of view of efficiency, this approach is not desirable, but the implementation is extremely simplified. The mutexes synchronization primitives are implemented using a mixed approach, which aims to achieve greater efficiency. A short active wait is initially attempted. If the mutex is still occupied after that, the kernel proceeds to block the execution of the thread. For this purpose, the Linux-provided Fast User-Space Mutex (futex) basic access locking facility is used.
Testing of the multi-threaded library is done against two main types of system load:
• With processes limited by CPU time. These are processes that perform a large amount of calculations.
• With processes limited by input-output operations. These are processes that perform intensive system calls to the operating system.
The first test involves matrix multiplication, as an example of a problem containing many calculations. The pthreads library and kernel-space implementations perform best.
The third test aims to evaluate the efficiency of synchronization primitives when accessing a shared resource. In a loop of 1000000 iterations, the mutex is locked and unlocked. 5 separate threads are created, each of which performs the described action. The results show that a mutex implementation in the pthreads library is slower and takes 615154μs on average. The version in the kernel-space library is about 1.5 times faster, thanks to a simpler and more efficient thread organization structure.

Г.8.8 Вълчанов Х., Д.Тодоров, Система за индексирано търсене в локална Windows мрежа. Сб. Научна конференция 2015, РУ, 57-61, ISSN 1311-3321
Valchanov H., D. Todorov. System for indexed search in local Windows network. Proc. sc. conf., Rousse, 2015, pp. 57-61, ISSN 1311-3321

The paper presents the architecture of a distributed system for indexing and searching on a local Windows network. The system indexes in addition to file names and their contents. The indexing process can be performed in both RAM and disk memory. The system allows complex Boolean queries to be set when searching. Access to system functionality is organized based on user groups.
The architecture of the indexed search system on a local Windows network has a distributed organization. It consists of multiple independent services running on individual machines on the network. The Service consists of the following main components: an indexing subsystem, a search subsystem and a communication subsystem. Each machine on the network stores only its indexes. In this way, there is no need to use a server machine to store all the indexes of the participants in the local network. As a result, the volume of messages over the network is reduced.
The implemented strategy allows each component of the local network to manage its indexes by itself. Each of the services can accept local search requests as well as network requests. The exchange of information between individual components over the network is carried out by the communication subsystem.
The experimental studies were done on a local network with connected Windows machines. Two sets of tests were conducted - for indexing a specified directory with a volume of 2.12 Gb and search queries on the local machine.
The presented results show that in indexing The system load is very high in Microsoft indexing and in-memory indexing. With in-memory indexing, all indexes are stored in RAM, which leads to high overhead, while Microsoft does not know how they build the indexes.
In query processing, the results show that processing a query with a mask in OP (with a tree structure) is twice as fast as in disk memory (by repeatedly loading a vector buffer), with the largest time being the Microsoft search.

Г.8.9 Todorov D., H. Valchanov. Multicast Indexing and Search System in Local Network. Proc. of UNITEH’15, Gabrovo, 2015, pp.II-269 - II-274. ISSN 1313-230X

The paper presents the system for indexing and searching with a distributed architecture. It consists of multiple search engines (TMs) that are installed as services on all local network machines participating in the information search process. Each TM functions independently of the others by indexing information on the local machine. Search requests can be issued locally (by the user on the local machine) or sent to all cores over the network. The exchange of information between TMs is implemented on the basis of multicast. Each TM consists of three main components: for indexing, for searching and for communication.

Indexing is based on the so-called inverted index. It is a list of discovered terms (words), and for each word a list of file indexes containing that term is built. Characteristically, each word is unique using a key when processing search queries. For each word, the position and frequency of occurrence in each file is maintained. This allows searching by phrase as well as sorting the results obtained.

The communication implementation in the presented system is based on Winsock2 multicast. The nature of the data exchanged (words, search phrases, search results in the form of file paths and the files themselves) requires that they be delivered reliably. Winsock2's embedded multicast protocol for reliable delivery is Pragmatic General Multicast (PGM).

Research has been conducted to investigate the functionality of the proposed system in a real local area network and to compare its performance with that of the built-in Windows indexing and searching system.

The experiments were performed on a local network including four clients and a separate authenticator machine. The computers are running Windows 7 Enterprise. Two types of tests were performed: indexing and query searching:

• Indexing. Three types of indexing are tested: in-memory indexing, disk-based indexing, and Windows indexing.

• Fulfillment of requests. Four types of requests are tested:

o term search - term;

o search with a mask - term*;

o phrase search – term1 term2;

o logical search of the type term1 AND term2 OR term2

OP indexing experiments consistently show better times than in-memory indexing, and by an order of magnitude against Microsoft. When processing a query with a mask in OP (with a tree structure), it is twice as fast as with disk memory (by repeatedly loading a vector buffer), and with the largest time is the Microsoft search. For single word searches and logical expressions, OP search again gives the best results, the vector buffer algorithm is slower, while Microsoft's performance is twice as slow as disk search.

Г.8.10 Николов В., Х. Вълчанов. Система за анализ и диагностика на цифрови изображения на кръвни проби. Компютърни науки и технологии, 2015 81-88, ISSN 1312-3335.

Nikolov V., H. Valchanov. System for analysis and diagnozismof digital images for blood samples. // Computer Science and Technologies, 2015 pp.81-88, ISSN 1312-3335

The article presents a system for rapid diagnosis of diseases affecting the state of blood cells, such as changes in size, shape, color, presence of inclusions in them, etc. The presented solution is based on the analysis of the images of blood cells, by determining their contours, through which a description is formed and a neural network is trained for recognition.

Since blood sample images have a large number of visual objects to be automatically analyzed, it is necessary to extract only certain features from the segments in the image. For this purpose, signs are compiled, each of which represents an approximating straight line of the contour of a blood cell. A characteristic is represented by a vector of five elements (x1, y1, x2, y2, sin, cos), and for their determination the following steps are performed:

• digital representation of the image;
• determining the contours of visual objects (red blood cells) and their thinning;
• traversing the contours and determining control points for distinguishing signs;
• forming a description of the objects in the image by describing the signs.

After forming the description of the contours of the images, the training of the neural network is started. It is of the multilayer perceptron type and is trained by the widely used back-propagation algorithm.

The system structure consists of an administrative module and a number of client modules. The administrative module contains disease information that is provided to the client modules using web services. The information is built through pre-structured and disease-grouped images of blood cells. Classification is carried out by experts, medical persons who provide examples of use.

Client modules can initiate a query for the most current disease definitions and perform classification on a submitted image. As a result of the request, the main module provides a trained neural network in the form of an XML tree, which is built in memory at the client. The XML information contains the architecture of the neural network (number of layers, number of neurons) and the weights of the connections themselves. The functions of the client module only allow diagnostics by determining the category of the image presented at the input.

The presented system uses a connectionist approach for the classification of images presented in the form of matrices of numbers, using the information of previously formed classes of images. A goal of future work is the integration of the presented system into a comprehensive infrastructure for early notification of epidemics.

Г.8.11 Вълчанов Х. Комуникационна среда за търсене в локална мрежа. Proc. of TechSys’16. Plovdiv, 2016. ISSN 1310 – 8271, II-199 – II-202.
Valchanov H. Communication framework for searching in local network. Proc. of TechSys’16. Plovdiv, 2016. ISSN 1310 – 8271, II-199 – II-202

The paper presents the communication subsystem architecture of a distributed indexing and searching environment on a local Windows network. The indexing and searching system has a distributed architecture. It consists of multiple search engines that are installed as services on all local network machines participating in the information search process. The system allows execution of several types of queries: search by word, by part of a word (mask search), phrase search and complex logical expressions.
The communication subsystem ensures the exchange of information between TMs executed on separate machines. This exchange involves the transmission of search requests to all machines and the sending of the search result. Communication in implemented on Winsock2 multicast protocol PGM.
The communication subsystem consists of several basic components:
• Communication kernel – kernel providing the basic functionality.
• User message queue - a synchronized queue of messages that are intended to be transmitted or processed.
• Search result – a buffer for storing search results.
• Hash – buffer for the hash used in the phases of the authentication process.
• RX, TX port – PGM sockets for receiving and sending messages over the network.
The multicast address on which the system communicates is fixed and common to all instances of the application. In this form, every user would participate in searches. In a real-world environment, this is woefully inadequate. In order to divide users into subsets and limit searches to the limits of the given subset, the mechanism of groups is introduced. Each user can be a member of one or more groups. Data read protection is implemented based on the multicast encryption concept. Unlike the classic scheme, instead of having one key for the multicast group, one key is entered for each group of the system, called a group key, provided by a special centralized component - an authenticator.
Experimental research has been conducted on a local network with Windows machines. Two types of tests were performed: indexing and query searching. The indexing process can be performed in both RAM and disk memory. Experimental comparisons and evaluations of the two methods of information indexing, as well as searching with Windows 7's built-in indexing and searching system, have been made. The results show that better performance is achieved than the existing Windows implementation.
Development of a hybrid indexing algorithm as well as automatic generation of authentication keys are envisaged as directions for future work.

Г..8.12 V.Aleksieva, H.Valchanov, M.Magdziak-Toklowicz, R.Wrobel, R.Wlostowski, Transmission of vibrations from the engine to the car body, Journal of KONES Powertrain and Transport,vol.23,No.4 2016, pp.17-23, ISSN:1231-4005

Vibrations have become an important factor of vehicles. Vibration tests help identify, and then tune the automotive vehicle to improve the structural strength. Vibration testing is often carried out using Laser Doppler Vibrometry (LDV), a device that is used for contactless measurement of vibration on the surface. The laser beam is directed from the device to the surface of interest, and the amplitude and frequency of vibration are extracted from the Doppler shift frequency of the reflected laser beam due to the movement surface. High values of vibration transmitted from the engine, and the way significantly affect the body of the vehicle and the driver are investigated.
Article presents results of research carried out on vehicles powered by three different engines and rpm. Tests were carried out on an engine dynamometer in the same environmental conditions. Two of engines were with spark ignition, including one with a supercharged engine and compression ignition engine.
The measurements were made using the Laser Doppler Vibrometry using Fast Fourier Transform. The spectrum obtained is used for further analysis to determine the acceleration level at various frequencies. Obtained from Fast Fourier Transform readings used for drawing graphs of frequency V acceleration.
As the rotational speed of the crankshaft increases (and the length of the period decreases), additional fluctuations occur in all types of vehicles (they were clearly seen even for the lowest velocity of the supercharged engine), but the vibration signal is of stationary character.
The diagrams show unambiguously that the amplitude of the relative vibration velocity notwithstanding the measurement target is the greatest for the vehicle with compression ignition engine and the lowest for the vehicle with spark ignition engine (non-supercharged). Simultaneously, the fluctuations and mean values of the signals indicate that the vehicle with diesel engine that is most ergonomic, whereas the vehicle with supercharged spark ignition engine is least ergonomic.

Г.8.13 Алексиева Ю., Х. Вълчанов. Симулатор на Ботнет DoS атаки в мрежова среда. Сб. на международна конференция "Автоматика и информатика”, 2016, София, 2016. 163-166. ISSN 1313-1850
Aleksieva Y., H. Valchanov. Network simulator for botnet DoS attacks. Proc. of SAI 2016, pp. 163-166. ISSN 1313-1850

One widespread variant of botnet attacks is the IRC botnet. There are a number of techniques to detect botnet attacks, but they lack the functionality to prevent malicious botnet activity. The development of quality botnet threat detection and removal equipment needs a good simulation environment that is safe and fully meets the functionality of botnet DoS attacks. The report presents the specifics of implementing a network simulator for DoS attacks, allowing you to create different attacks with different parameters.
The architecture of the presented simulation environment has a distributed nature. It consists of multiple bots running on separate machines and communicating through IRC channels. Each bot is built on a modular basis. The use of modules enables easy subsequent expansion of the simulator with new components implementing additional classes of attacks.
The presented simulation environment aims to simulate the action of malicious network attacks. For this reason, the method of turning machines into bots is not the subject of research in the presented report. It is assumed that the botnet network has already been created - the computers have been infected. In order to simulate the operation of an IRC botnet, each machine must have a bot pre-installed. Due to the potential for this software to be misused, the functionality to start it automatically when a machine is turned on has not been added.
The need for combinativity regarding the execution of different types of attacks requires that each type of attack can be executed autonomously with maximum parameterization options. Hostnames/addresses, port numbers, number of packets, size of packet content, etc. are specified as parameters. Each attack can be configured with a specific duration. After each execution of a command or attack, the bots communicate their status via a personal message to the user. Each bot sends a message with a report of how many packets it sent, how long the attack lasted, and what the result of the executed command was.
The testing of the simulation environment was done with respect to two main types of network infrastructure - planar and routed.
The tests were carried out in two directions: to test communication between bots and to implement network attacks. The bots are started on the respective machines and when they start, they join the test channel through which the attacking bot will issue the commands. The conducted tests show the workability of the developed system.

Г.8.14 Ю.Алексиева, Х.Вълчанов. Симулатор на ботнет DoS атаки в мрежова среда. Автоматика и информатика, N:2, 2016, 43-52. ISSN 0861-7562
Aleksieva Y., H. Valchanov. Network simulator for botnet DoS attacks. Proc. of SAI 2016. //Automatics and Informatics, N:2, 2016, pp.43-52. ISSN 0861-7562

The article presents some aspects of the implementation of a simulation environment for generating IRC DoS network attacks. The architecture of the system is distributed, consisting of multiple bots that run on separate nodes and communicate with each other via IRC channels. Each bot consists of the following components:
• Core – realizes the basic functionality of the bot by coordinating the work of its individual components.
• Command interpreter – implements the internal system for receiving and processing commands from remote users.
• Attack generators – implement the three basic types of attacks - TCP, UDP and ICMP.
• Communication subsystem – implements communication to other bots in the network using the mechanism of sockets.
• IDE – graphical interface of the bot, allowing interaction with a user on the local machine.
The system can operate with at least one bot connected to the network up to an unlimited number of bots. The bot is run locally on each machine and controlled via the IRC channel it connects to. This IRC channel is the C&C center of the botnet. Botnet DoS attacks are managed remotely by issuing commands in the designated IRC channel. All bots receive the same command and each one executes it independently of the others in order to increase the effectiveness of the attack. After executing the attack, the bots use the IRC protocol and connect to the user of the channel who set the command, sending him a message with a report of the activity performed.
The testing of the simulation environment is done with respect to two main types of network infrastructure - planar and routed. The tests were carried out in two directions: to test communication between bots and to implement network attacks. The bots are started on the respective machines and when they start, they join the test channel through which the attacking bot will issue the commands. The second set of tests examines the functionality and efficacy of generated network attacks. The objective is to determine the effectiveness of the simulation environment for saturating the communication channel with packets. The attacks were carried out against the three basic protocols TCP, UDP and ICMP. Information about traffic results for each attack is obtained from WireShark aggregate reports.
As directions for future work, it is envisaged to expand with generators of other network attacks, as well as to add capabilities to implement DDoS attacks. Another direction is the use of threads to increase the efficiency of the simulator.

Г.8.15 Димитър Н. Тодоров, Христо Г. Вълчанов, Разпределена система за търсене в локална мрежа. Компютърни науки и технологии, ТУ-Варна, 2016, бр.1, с.32-38, ISSN 1312-3335.
D. Todorov, H. Valchanov. Distributed system for searching in local network. // Computer Science and Technologies, 2016, N.1, pp.32-38, ISSN 1312-3335

Sharing resources on a local network is one way to efficiently use information. The Windows search system does not provide a local network search capability. The article presents the architecture and features of a distributed search system in a local area network.
The architecture of the system has a distributed nature. It consists of multiple search engines, each running on a separate machine on the local network. Modules function independently by communicating with each other, exchanging information about the search process. Each module consists of two main systems - an indexing system and a search system.
The indexing system ensures the construction of the index files. It performs the processing of the indexing request, the extraction of text from documents, the implementation of the word simplification algorithm and the construction of the indexes. Indexes are built in two ways: in RAM and in disk memory. The process of building indexes in RAM starts by removing all punctuation characters from the next document and returns clean text as a result. Separation of the terms (words) is realized by Flex Lexical Analyzer generated code. The indexing subsystem terminates by writing the created index files to the machine's disk memory, frees the used RAM, and the service goes into standby mode. The steps for indexing in disk memory are analogous to indexing in RAM, but with some differences. In this case, there is a hard-fixed buffer in which the formed lists are written. When the buffer is full, it is stored in disk memory, then cleared and the indexing process continues until all files in the desired directory and its subdirectories have been indexed.
The system processes the received search requests, which can be both locally submitted and received over the network from other machines. The already filtered query is checked for logical operators, masked terms, phrase, or is only a single term search. Each query term is checked against the list of constructed indexes. Depending on how the indexes are built, different functionality is applied. When indexing in OP in memory, the tree is loaded first. Each character of the specified term is searched in the tree structure. The last node of each recorded word contains a pointer to the term's list of file indexes. When indexing the disk, the index card is loaded. First, the initial letter of the term is checked. If a match is found, the starting and ending positions are taken, a buffer with the corresponding indices is loaded, and the term is searched for in it. The buffer can be refilled depending on the size of the indexes. The system allows execution of several types of queries: search by word, by part of a word (mask search), phrase search, complex logical expressions.

Г.8.16 Райчинов К., Х. Вълчанов. Архитектура на система за предпазване от мрежови атаки. Proc. of UNITECH'16, Gabrovo, 2016, pp.II-316-II-321. ISSN 1313-230X
Rajchinov K., H. Valchanov. Intrusion preventing system for network attacks. Proc. of UNITECH'16, Gabrovo, 2016, pp.II-316-II-321. ISSN 1313-230X

This paper presents the architecture and basic functionality of an IDPS attack detection and protection system, which offers an affordable, flexible and efficient solution for building similar systems in various fields.
The architecture of the presented system has a modular character. It consists of multiple modules, each having a certain functionality:
• Operating system on which the software implementation runs.
• Software performing the detection of attacks.
• Event translator.
• Console for presenting network traffic analysis results.
The operating system used is the Debian Linux based system - Raspbian, which is one of the officially supported systems according to the Raspberry Pi Foundation and provides capabilities that meet the requirements of the current development. The operating system runs Snort, which is a cornerstone of network IDPS and proven software. In this implementation, the attack detection system interacts with the network traffic through a transparent bridge through which the traffic passes.
The testing of the presented system was performed in a real network infrastructure. Denial-of-service (DoS) attacks are simulated. A large amount of requests (on the order of hundreds of thousands) are generated from fake sources in flooding mode, in a maximally short period
The system is based on open source, allowing extreme flexibility in case of need for modifications and further developments, but also providing an opportunity to check the security of individual programs and guaranteeing high-level code quality.
A goal of future work is to develop the architecture by using multiple Raspberry Pi sensors distributed in different key nodes of the network to forward their output data to a centralized base, thus optimizing the use of the attack detection system.

Г.8.17 Martin Todorov, Hristo Valchanov. System for generating of DoS network attacks. In Proc. of University of Rousse- 2016, volume 55, book 3.3, pp.7-11, 2016. ISSN 1311-3321

The paper features a system for generating network DoS attacks that provides the ability to implement several of the most common and used attacks in a way that is convenient for learning the principles behind them. The system is extremely convenient for training network administrators, who can test their network and devices for software and configuration gaps, as well as properly structured topology and security.
The architecture of the attack generation system has a modular nature. The advantage compared to the presented similar means is the achievement of flexible functionality. On the one hand, it is possible to easily modify the actions when generating attacks, and on the other hand, the functionality can be expanded by adding new modules implementing new attacks.
• The core implements the basic functionality by coordinating the work of the other components.
• The attack modules implement the attacks on the three main protocols – UDP, TCP and ICMP. The separation of three basic modules is determined by the specifics of packet formation for these basic protocols. Each of the modules includes separate libraries that are loaded into the system's kernel at startup.
• The UDP attack module implements a UDP flood attack, sending multiple UDP packets to a specific port on the target machine in order to block a service and generate large traffic. The TCP module implements a SYN flood attack in order to borrow resources from the destination. The ICMP module generates two types of attacks on this protocol – Ping of Death, which overflows the receiving machine's buffers, and Smurf attack, which aims to generate massive traffic on a particular network segment. The module for RAW attacks implements ARP Spoofing in which a MAC address is replaced in the address table.
Experiments were carried out in order to study the functionality of the presented system in a real network environment. The first set of tests is for a UDP attack. A machine running a DNS server is attacked. The second group is a SYN flood attack where Apache HTTP server is used as a service. The third group is ARP Spoofing attack. It sends a fake MAC address of the router in the segment as a response.
The conducted experiments show that the proposed system has full functionality in terms of basic network attacks.
A goal of future work is to develop additional components for other attacks requiring more specialized means of implementation, as well as a graphical interface for more convenient manipulation of attack parameters.

Г.8.18 Raychinov K., H. Valchanov. Intrusion Detection and Preventing System. In Proc. of TechSys’17, 2017, Plovdiv, pp. II 177-II 180. ISSN Online: 2535-0048

The paper presents the architecture and functionality of an attack detection and protection system. The main idea of the developed system is its easy integration into any network infrastructure. For this purpose, a simple and effective solution is proposed using a Raspberry Pi single-board computer. A characteristic feature of this system is its integrity - the concentration of all software components on powerful hardware, at the same time with a small size and low cost of ownership. The proposed system takes into account the convenience and functionality provided by the graphical console for analyzing detected threats. This is how a high efficiency architecture is realized.
Traffic through the device is controlled by Snort. For greater Snort efficiency, system data is generated in a binary file in unified2 format. This file is processed by a translator that analyzes recorded events and specific traffic and forwards them to a database. The translator used is Barnayrd2 running in parallel with Snort and redirecting new events from the binary to the database in real time. MySQL is used as the database management system. Key to any signature-based IDPS are lists of rules that define events as having a possible threat level or not. These signatures are stored by Snort in several types of local files.
The system testing was done by simulating a DoS attack. The attack is carried out by sending only SYN packets, targeting the router in the network. Over 65,000 reports of unusual network behavior are generated. Their number continues to grow because the translator continues to read from the binary and populate the database with new messages, even though the simulated attack is over. The reason for the anomaly detected by the system is data in SYN packets intercepted by Snort's Stream preprocessor.
The results of the conducted experiments show that the developed system is fully functional and applicable in real conditions.

Г.8.19 Rajchinov K., H.Valchanov. Embedded Network Intrusion Detection and Preventing System. In Proc. of Conf. Automatics and Informatics’17, Sofia, 2017, pp.249-252. ISSN 1313-1850

Network Intrusion Detection and Preventing System (NIDPS) is software that automates the process of detecting attacks and preventing possible incidents. The paper presents a NIDPS system for integration into any network infrastructure based on a Raspberry Pi single-board computer.
In order to avoid physical accessibility to the Razpbery Pi (keyboard or terminal console), an engineering solution has been implemented with the addition of a third WiFi network interface. The standard B model of Raspberry Pi rev.2 has only one Ethernet interface, but to implement the system you need at least two, and in the best case, three network communication interfaces. For this purpose, additional external modules of two USBs have been added - one USB-Ethernet adapter, with Ethernet and one USB dongle for a wireless WiFi connection. In this way, the intended functionality is achieved through two interfaces for network traffic to pass through to be checked for problems and one interface to access the system for the purpose of managing it. The two Ethernet interfaces have no assigned IP addresses and are connected in a bridge operating at the channel layer of the OSI model, which is transparent to the upper layers.
The network attack detection system is located in a part of the network through which traffic passes between the internal and external networks of the organization. The sensor of the passing traffic inspection software - Snort, must be behind the devices communicating over an encrypted connection (for example, VPN routers), so that the monitored traffic is not encrypted. At the same time, it must be located behind a firewall to prevent possible attacks.
Attack detection is based on the presence of anomalies. After detecting the events, it is necessary to classify them, from the point of view of the analysis. In concrete experiments, they are defined as a denial of service attack due to the volume of requests from numerous addresses without a completed TCP dialogue. This classification is done through the management console - Snorby. After the mass classification process, the new events are marked as a Denial of Service attack.

Г.8.20 Raychinov K., H. Valchanov. Intrusion Detection and Preventing System. //Journal of the technical University of Sofia Plovdiv Branch, Bulgaria, Fundamental science and Applications,7, 2017, vol.23, pp. 97-100, ISSN 1310 - 8271

The article presents the architecture and functionality of a system for detecting and protecting against attacks. The main idea of the developed system is its easy integration into any network infrastructure. For this purpose, a simple and effective solution is proposed using a Raspberry Pi single-board computer. A characteristic feature of this system is its integrity - the concentration of all software components on powerful hardware, at the same time with a small size and low cost of ownership. The proposed system takes into account the convenience and functionality provided by the graphical console for analyzing detected threats. This is how a high efficiency architecture is realized.
Traffic through the device is controlled by Snort. For greater Snort efficiency, system data is generated in a binary file in unified2 format. This file is processed by a translator that analyzes recorded events and specific traffic and forwards them to a database. The translator used is Barnayrd2 running in parallel with Snort and redirecting new events from the binary to the database in real time. MySQL is used as the database management system. Key to any signature-based IDPS are lists of rules that define events as having a possible threat level or not. These signatures are stored by Snort in several types of local files.
The system testing was done by simulating a DoS attack. The attack is carried out by sending only SYN packets, targeting the router in the network. Over 65,000 reports of unusual network behavior are generated. Their number continues to grow because the translator continues to read from the binary and populate the database with new messages, even though the simulated attack is over. The reason for the anomaly detected by the system is data in SYN packets intercepted by Snort's Stream preprocessor.
The results of the conducted experiments show that the developed system is fully functional and applicable in real conditions.

Г.8.21 Y.Aleksieva, H.Valchanov. Network Simulator for Botnet DoS Attacks. Information Technologies and Control, N:1, 2017, 33-40, ISSN 1312-2622

The article presents some aspects of the implementation of a simulation environment for generating IRC DoS network attacks. The architecture of the system is distributed, consisting of multiple bots that run on separate nodes and communicate with each other via IRC channels. Each bot consists of the following components:

• Core – realizes the basic functionality of the bot by coordinating the work of its individual components.

• Command interpreter – implements the internal system for receiving and processing commands from remote users.

• Attack generators – implement the three basic types of attacks - TCP, UDP and ICMP.

• Communication subsystem – implements communication to other bots in the network using the mechanism of sockets.

• IDE – graphical interface of the bot, allowing interaction with a user on the local machine.

The system can operate with at least one bot connected to the network up to an unlimited number of bots. The bot is run locally on each machine and controlled via the IRC channel it connects to. This IRC channel is the C&C center of the botnet. Botnet DoS attacks are managed remotely by issuing commands in the designated IRC channel. All bots receive the same command and each one executes it independently of the others in order to increase the effectiveness of the attack. After executing the attack, the bots use the IRC protocol and connect to the user of the channel who set the command, sending him a message with a report of the activity performed.

The testing of the simulation environment is done with respect to two main types of network infrastructure - planar and routed. The tests were carried out in two directions: to test communication between bots and to implement network attacks. The bots are started on the respective machines and when they start, they join the test channel through which the attacking bot will issue the commands. The second set of tests examines the functionality and efficacy of generated network attacks. The objective is to determine the effectiveness of the simulation environment for saturating the communication channel with packets. The attacks were carried out against the three basic protocols TCP, UDP and ICMP. Information about traffic results for each attack is obtained from WireShark aggregate reports.

As directions for future work, it is envisaged to expand with generators of other network attacks, as well as to add capabilities to implement DDoS attacks. Another direction is the use of threads to increase the efficiency of the simulator.

Г.8.22 Valchanov H., D.Trifonov. Performance analysis of Virtualization and Containerization Platforms for Big Data Processing. Proc. of UNITECH'17, Gabrovo, 2017, pp. 203-208, ISSN 1313-230X

The paper presents a comparative analysis of the performance of systems for processing large volumes of data, built on the basis of virtual machines and on containers.
The difference between virtualization and containerization is mainly in the location of the virtualization layer and the way system resources are used. Containerization, also called "container-based virtualization," "paravirtualization," or "application virtualization," is a virtualization method for deploying and running distributed applications at the operating system level without the need to run an entire virtual machine for each application. Instead, multiple isolated systems, called containers, run on a single host and access the operating system kernel.
The VMware ESXi 6.5 hypervisor was chosen as the platform for virtualization, and Docker for containerization. The choice of these two platforms is dictated by their wide use, high performance and capabilities.
The tests are selected to cover the requirements for the various hardware devices: processor, memory and storage devices. The first test tests the performance of the system when processing data in a semi-structured form. Such processing is constantly required in Big data systems, because the input data quite often comes from heterogeneous sources and is in different formats. The results show a ~4.5% advantage for containerization and Docker. The second test is complex in terms of processor processing and input/output operations – Hive dataset import. This test uses the MapReduce programming model. A large file is read and imported into a MongoDB non-relational database table. It should be noted that this test does not give a clear result for the advantage of virtualization or containerization. It runs faster than the first test, because the text is stored in a different format in the database, and data processing is reduced only to extracting from the text and writing it to the database. The phases of sorting and combining the sorted results are missing.
The results obtained give reason to claim that containerization gives slightly better results in most cases.

Г.8.23 Valchanov H., D.Trifonov. Performance analysis of Virtualization and Containerization Platforms for Big Data Processing. Proc. of UNITECH’17, Gabrovo, 2017, Selected papers, pp. II283-II288, ISSN 2603-378X.

The paper presents a comparative analysis of the performance of systems for processing large volumes of data, built on the basis of virtual machines and on containers.
The difference between virtualization and containerization is mainly in the location of the virtualization layer and the way system resources are used. Containerization, also called "container-based virtualization," "paravirtualization," or "application virtualization," is a virtualization method for deploying and running distributed applications at the operating system level without the need to run an entire virtual machine for each application. Instead, multiple isolated systems, called containers, run on a single host and access the operating system kernel.
The VMware ESXi 6.5 hypervisor was chosen as the platform for virtualization, and Docker for containerization. The choice of these two platforms is dictated by their wide use, high performance and capabilities.
The tests are selected to cover the requirements for the various hardware devices: processor, memory and storage devices. The first test tests the performance of the system when processing data in a semi-structured form. Such processing is constantly required in Big data systems, because the input data quite often comes from heterogeneous sources and is in different formats. The results show a ~4.5% advantage for containerization and Docker. The second test is complex in terms of processor processing and input/output operations – Hive dataset import. This test uses the MapReduce programming model. A large file is read and imported into a MongoDB non-relational database table. It should be noted that this test does not give a clear result for the advantage of virtualization or containerization. It runs faster than the first test, because the text is stored in a different format in the database, and data processing is reduced only to extracting from the text and writing it to the database. The phases of sorting and combining the sorted results are missing.
The results obtained give reason to claim that containerization gives slightly better results in most cases.

Г.8.24 H.Valchanov. Power Management of a Virtual Infrastructure. // Computer Science and Tecnologies, v.1, 2018 77-82, ISSN 1312-3335

The report presents the architecture of a power management system and virtual machines in an infrastructure built on the basis of a VMware ESXi cluster. Presented the virtual infrastructure that needs to be managed. This infrastructure is used in the educational process of the students of "Computer Systems and Technologies" and "Software and Internet Technologies" majors at TU-Varna. The main component is a cluster of HP BL460c Gen8 high-performance servers with a VMware virtualization platform based on ESXi 6.0. Over 50 virtual machines with a specific purpose for the educational process are running on them. Virtual machines are under Windows and Linux OS. A disk array and two server machines are installed in the communication cabinet. Power is provided by an Eaton 9SX5000 uninterruptible power supply.
The architecture of the power management system consists of several modules. The modules are implemented as services and scripts that run on a separate server running Windows Server 2012 R2. The SNMP trap receiver module runs as a background service. Its purpose is to monitor the network for the presence of an SNMP message with a certain code. This message is generated by UPS. The UPS is configured to go into shutdown mode at a certain remaining battery capacity (for research purposes, this is 65%).
If all the conditions are present, it is necessary to proceed to the normal shutdown of the infrastructure. A special script is launched that performs the necessary sequence of actions. The script is implemented using PowerShell under Windows and the VMware package - VMwareCLI. Initially, it goes through the authentication phase performed by the Authentication module. Its purpose is to ensure the correct identification and login to the various systems. After a successful connection, it goes to the actual shutdown sequence. It is implemented by the Shutdown module, which is executed as a script.
With each action, information is recorded in a log file, which can be used later by the administrator for analysis.
The functionality of the presented system does not depend on the type of uninterruptible power supply used. The system is modular, allowing for easy future expansion. The system has been implemented and for 1 year has proven its functionality.
As guidelines for future work, development of a graphical user interface is envisaged, as well as its integration as a service in the ESXi operating system.

Г.8.25 Todorov D., H.Valchanov. Routing and Traffic Load Balancing in SDN-NFV Networks. Proc of International Conference Applied Computer Technologies, 2018, pp.127-130. ISBN 978-608-66225-0-3

The paper presents an overview of methods for routing and load balancing traffic in SDN-NFV networks.
The wide use of virtualization technologies for building virtual networks and the need for means and methods for automating their management are presented.
Software-defined networking (SDN) and network function virtualization (NFV) architectures are discussed. The functioning of the SND controllers, the interaction with the controlled devices, as well as the application of the standardized interface between the controller and the devices, based on the OpenFlow protocol, are described.
The characteristic features of SDN routing techniques are considered, such as Virtual Routers as a Service (VRS), Routing as a Service (RaaS), RouteFlow Routing Control Platform through SDN (RFCP), SoftRouter, RouteFlow IP routing.
One of the main problems of computer networks is traffic load balancing. The report examines load balancing techniques in terms of two tiers: NFV and SDN.
Regarding NFV, solutions are presented by choosing routing paths, distributing traffic between NFV systems, and using the ORBIT algorithm.
Regarding load balancing in SDN, solutions such as the least loaded real-time server (RLS), using a controller to analyze responses from OpenFlow switches, applying the Dynamic Load Balancing algorithm, using heuristic methods are presented.

Г.8.26 Trifonov D., H. Valchanov. Virtualization and Containerization Systems for BigData. In Proc. of TechSys'18, 2018, Plovdiv, pp.II157 – II160, ISSN Online: 2535-0048

The paper focuses on the application of virtualization and containerization systems for large volumes of data. A performance study of a large data processing system deployed on virtual machines and containers is presented.
The data processing system is Hadoop. Hadoop is an open source software developed in Java. Hadoop has a number of tools for executing code and scripts in different programming languages. It consists of two main parts - a storage part and a data processing part. The storage component is the Hadoop Distributed File System (HDFS). The data processing part is MapReduce. It is a programming model for parallel processing of multiple tasks. One of the main aspects of MapReduce programming is that MapReduce divides tasks in such a way that they allow their parallel execution on a distributed system of computing nodes. Contrary to traditional relational database management systems that cannot scale to handle large amounts of data, programming in the Hadoop MapReduce environment allows users to run applications on a huge number of machines, which includes processing thousands of terabytes of data .
An experimental performance study of Hadoop on VMware ESXi hypervisor and Docker platform is presented. Three types of tests were conducted. The first test examines performance when processing unstructured data. The test is a Java application (WordCount) executed directly from Hadoop by calling MapReduce. Processing data is one of the latest Wikipedia (EN) archives, using only English content. It is provided as a bz2 archive which has an xml file of approximately 62 GB. The results show a ~4.5% advantage for containerization and Docker. This is to be expected given that the maximum amount of memory is used during the test and each virtual machine uses approximately 1.3 GB of memory that is otherwise used by Hadoop. Accessing disk storage for virtualization is potentially a bit slower, which also increases Docker's lead slightly.
The second test is complex in terms of CPU performance and I/O operations - importing a Hive dataset. This test again uses MapReduce. It reads the contents of the enwiki-20170701-pages-articles-multistream.xml text file and imports it into a relationless MongoDB table. The third test evaluates the storage system. It runs in two parts: TestDFSIO-write writes 100GB of data to the Hadoop-HDFS file system, and TestDFSIO-read reads it back. A slight advantage is again reported for containerization due to the fact that disk access through the virtual disk controller is slightly slower than through the interface provided by containerization.

Г.8.27 Y.Aleksieva, H.Valchanov. BOTNET DETECTION SYSTEM BASED ON GENETIC ALGORITHMS. In Proc. of Conf. Automatics and Informatics'18, Sofia, 2018, pp.129-139. ISSN 1313-1850.

The paper presents some aspects of the implementation of a host-based system for detecting botnet attacks. The system uses a genetic algorithm variation based behavior anomaly detection technique. The architecture of the presented system has a modular character:

• Core – realizes the basic functionality of the system by coordinating the work of individual components.

• Command interpreter – implements the internal system for receiving and processing the commands set by the user.

• Communication subsystem – realizes the communication between the user and the administrator / between the system and the administrator.

• GUI – a graphical interface of the system, allowing interaction with the user of the local machine.

A proposed algorithm for detecting Botnet attacks performs detection of spoofing attack by a specific variation of a genetic algorithm, and the presented variation is based on the genetic operator selection, evaluating all individuals in successive generations based on an analytically determined fitness function. The algorithm creates chromosomes from the resulting packets, examines them carefully, and thus detects changes in phenotype and mutation. In the case of the genetic algorithm used, the system treats the received packets as a phenotype. The algorithm extracts from the resulting package (phenotype) its genes and combines these genes into a verification chromosome. When detecting a spoofing attack, the algorithm examines the headers of each packet and extracts attributes (genes), such as IP address, MAC, TTL, Protocol number, etc.

The fitness function of the algorithm is an analytical calculation between two chromosomes obtained from the same address. The two chromosomes are pairwise compared bit by bit, mathematically calculating the fitness level of the resulting packet. When the system receives a packet for the first time from an IP address, its genes form a chromosome and it is saved with 100% fitness level (fitness level). The chromosomes of the next packets obtained from the same source are compared with the chromosome of this packet in the database, applying the fitness function, and the fitness level of the packet is similarly calculated. If the fitness level of the newly received packet is less than that of the previously received packet, then there is therefore an external interference and an alarm is generated.

The testing of the presented system was performed in a real network infrastructure. The testing results show that the algorithm works correctly, but the specific network needs to be carefully analyzed to choose an appropriate fitness level. In an attacked environment, the host-based system detects the attack immediately and signals adequately.

A goal of future work is to extend the functionality of the system, providing more possible anomaly detection techniques, such as adding data integrity analysis.

Г.8.28 Aleksieva Y., H.Valchanov. Anomaly Based Botnet Setection System. Proc. of UNITECH'18, Gabrovo, 2018, vol.2, pp. II123-II127, ISSN 1313-230X

The report presents some aspects of the implementation of a host-based system for detecting botnet attacks. The system uses a genetic algorithm variation based behavior anomaly detection technique.
The attack detection algorithm detects a spoofing attack by a specific variation of a genetic algorithm, the presented variation being based on the genetic operator selection, evaluating all individuals in successive generations based on an analytically determined fitness function. The algorithm creates chromosomes from the resulting packets, examines them carefully, and thus detects changes in phenotype and mutation. In the case of the genetic algorithm used, the system treats the received packets as a phenotype. The algorithm extracts from the resulting package (phenotype) its genes and combines these genes into a verification chromosome. When detecting a spoofing attack, the algorithm examines the headers of each packet and extracts attributes (genes), such as IP address, MAC, TTL, Protocol number, etc.
The fitness function of the algorithm is an analytical calculation between two chromosomes obtained from the same address. The two chromosomes are pairwise compared bit by bit, mathematically calculating the fitness level of the resulting packet. When the system receives a packet for the first time from an IP address, its genes form a chromosome and it is saved with 100% fitness level (fitness level). The chromosomes of the next packets obtained from the same source are compared with the chromosome of this packet in the database, applying the fitness function, and the fitness level of the packet is similarly calculated. If the fitness level of the newly received packet is less than that of the previously received packet, then there is therefore an external interference and an alarm is generated.
Once the chromosome has been generated, its fitness level must be calculated. Let, for example, the chromosome of a packet received from a source for the first time, which is generated by the algorithm, is 1010101010. Then, if another packet is received from a source with the same IP address, its chromosome is expected to be almost the same, if not - at least close to the previous one. This is because the source and destination MAC addresses must be the same, and the TTL and HopCount must be approximately the same. Also, the packet ID is expected to be greater than the previous packet ID. If all these are present, the second packet received is expected to have almost the same chromosome. For example, let the chromosome of the second packet received is 1100101001. The comparison shows 4 differences in the chromosomes. In this case, the second package has a fitness level of 60%, which means that it has been significantly modified and the system will generate an alarm.
The testing of the presented system was performed in a real network infrastructure. The test results show that the algorithm works correctly, but the specific network needs to be carefully analyzed to choose an appropriate fitness level.

Г.8.29 Trifonov D., H.Valchanov. VIRTUALIZATION AND CONTAINERIZATION SYSTEMS FOR BIG DATA. //Journal of the technical University of Sofia Plovdiv Branch, Bulgaria, Fundamental science and Applications,7, 2018, vol.24, pp. 129-132. ISSN 1310 – 8271.

The article focuses on the application of virtualization and containerization systems for large volumes of data. A performance study of a large data processing system deployed on virtual machines and containers is presented.
The data processing system is Hadoop. Hadoop is an open source software developed in Java. Hadoop has a number of tools for executing code and scripts in different programming languages. It consists of two main parts - a storage part and a data processing part. The storage component is the Hadoop Distributed File System (HDFS). The data processing part is MapReduce. It is a programming model for parallel processing of multiple tasks. One of the main aspects of MapReduce programming is that MapReduce divides tasks in such a way that they allow their parallel execution on a distributed system of computing nodes. Contrary to traditional relational database management systems that cannot scale to handle large amounts of data, programming in the Hadoop MapReduce environment allows users to run applications on a huge number of machines, which includes processing thousands of terabytes of data .
An experimental performance study of Hadoop on VMware ESXi hypervisor and Docker platform is presented. Three types of tests were conducted. The first test examines performance when processing unstructured data. The test is a Java application (WordCount) executed directly from Hadoop by calling MapReduce. Processing data is one of the latest Wikipedia (EN) archives, using only English content. It is provided as a bz2 archive which has an xml file of approximately 62 GB. The results show a ~4.5% advantage for containerization and Docker. This is to be expected given that the maximum amount of memory is used during the test and each virtual machine uses approximately 1.3 GB of memory that is otherwise used by Hadoop. Accessing disk storage for virtualization is potentially a bit slower, which also increases Docker's lead slightly.
The second test is complex in terms of CPU performance and I/O operations - importing a Hive dataset. This test again uses MapReduce. It reads the contents of the enwiki-20170701-pages-articles-multistream.xml text file and imports it into a relationless MongoDB table. The third test evaluates the storage system. It runs in two parts: TestDFSIO-write writes 100GB of data to the Hadoop-HDFS file system, and TestDFSIO-read reads it back. A slight advantage is again reported for containerization due to the fact that disk access through the virtual disk controller is slightly slower than through the interface provided by containerization.

Г.8.30 Hristo Valchanov, Veneta Aleksieva, Jan Edikyan, Study of wireless networks security, UNITECH'19, 15-16 November 2019, GABROVO, vol.II, pp. 7-11, ISSN 1313-230X

Wireless network allows easy to build small enterprise and home networks based on IEEE 802.11 standard. However, wireless networks are easily susceptible to attacks against their security. This requires an analysis of the problems and creating recommendations to improve their security. This paper presents a methodology and study of wireless network security in Varna city. The information was collected using the war-driving technique. The obtained results are analyzed and compared with those from previous studies.

The data collection system is based on a single-board computer Raspberry Pi 3 Model B, with ARM Cortex-A53 processor, 1.2GHz, built-in Wi-Fi and Bluetooth functionality. For the purpose of the implementation, it is necessary to record the position of each wireless access point. The chosen GPS module, due to support for the NMEA 0183 standard, long-lasting battery and large memory, is the Holux M-1200E. The CanaKit Wi-Fi Module is used to scan wireless networks. A Canyon CNS-TPBP5DG portable battery with a capacity of 5000mAh is used to provide a long-term power to the single-board computer.

The scan of the wireless networks is done using open source software Kismet. The software is compiled and installed under the Raspbian OS operating system. The data received from Kismet is saved in netxml format. The collected information is converted via Python script to csv format. This is necessary so that the data can be presented in tabular form for easier processing and analysis through Microsoft Excel. The selected area for analysis includes the central part of Varna, as it is home to most of the offices and most of the residents. Also, the region coincides with a similar study conducted in 2008, in order to compare the results obtained.

The results show a significant increase in the security of Wi-Fi networks in the city, but there is still room for improvement.

The reasons for improving security can be considered in two ways. First, manufacturers offer devices that have WPA2 configured by default. Second, larger organizations have IT departments that take care of security. Based on the results of the detected SSIDs, mixed mode WPA / WPA2 and WPS, it can be concluded that most of the analyzed Wi Fi networks belong to ordinary users who do not have sufficient security knowledge.

The main recommendations can be presented in the following areas:

1. Use only WPA2 encryption method.
2. Disable WPS for all devices.
3. Choose a complex password.
4. Update the device software to the latest version.
5. Inform users about Wi-Fi security issues.

Г.8.31 Veneta Aleksieva, Hristo Valchanov, Yuri Dimitrov, Study of smart watch interfaces, UNITECH'19,15-16 November 2019, GABROVO, vol.II, pp. 12-16, ISSN 1313-230X

Smart watches are wearable devices' small sizes. Their display size and limited space for input controls require specific attention to the device interfaces development processes. The research in this paper aims to compare two different approaches in the interface design - a Novel interface based on two-finger touch interface activation and management on touch sensitive device bezel (the surface that is around the display) to a Standard "wristwatch" style input interface based on side push buttons.

The purpose of this study is to compare the process of human-smartwatch interaction, when the same user performs the same task using two prototypes of smartwatches developed for this purpose. For the purposes of the study, two experimental prototypes with different input interfaces were made - "Novel" with a touch-sensitive bezel interface, which was developed for a previous study, and "Standard" with four buttons on the side of the device, which is designed and developed for the current research.

In order for the research results to be completely comparable, the two models are made on the basis of the same 3D model of a wristwatch. The experimental models are controlled by a computer Arduino Mega 2560. Software has been developed to control each model based on Arduino. Both software products recognize four main interoperability command interfaces - Up, Down, Select, and Back.

The test group consists of 10 volunteers using their right hand to work with the prototypes. The average age of the participants in the experiments was 37.6 years. All experiments were performed under equal other conditions - in the same room, without artificial lighting.

The study of the developed models of smartwatch interfaces was conducted in three stages - the first stage compares the time and accuracy of a simple task (choice of only one function), the second stage compares the time and accuracy of a complex task (choice of function in several steps), and in the third stage the volunteers give a subjective assessment of the comfort of working with both interfaces.

Experimental data are presented.

The conclusions of the study can be summarized in the following:

• The new experimental model surpasses the standard in speed of work. When the set of commands is longer, the benefit of using the new interface model is greater.

• The error rates when working with the new model are higher than when using the standard model. The reason may be the fact that the traditional interfaces with side buttons on an electronic watch are familiar to most people, but the interface of the new model with a touch-sensitive ring is something new for them.

• Users' assessment of the comfort of working with both interfaces is higher for working with a standard model.

Г.8.32 В. Алексиева, Х.Вълчанов, А. Хулиян, Приложение на интелигентни договори базирани на Ethereum блокчейн за целите на застрахователни услуги, // Информатика и иновативни технологии, сс.7-14 бр.1(1),2019, ISSN 2682-9517
Aleksieva V., H.Valchanov,A.Huliyan, Application of smart contracts based on Ethereum blockchain for the purposes of insurance services, Informatics and inovative technologies, pp.7-14, No1(1),2019, ISSN 2682-9517)

This article presents an experimental implementation of smart contract for insurance service on the Ethereum blockchain. The authors present a classic model of insurance service and point out its shortcomings. On this basis, they offer a model for insurance services based on blockchain technologies. An experimental implementation on Ethereum blockchain is presented.
The claim processing process can be improved using smart contracts and blockchain technology. The information about the occurred damage can be sent by the insured or directly by sensors installed in the insured object (smart asset), to an automated application for processing a claim. For the relevant insurance policies provided by the smart contract, the customer will receive real-time feedback. The claim is processed automatically by a smart contract based on a set business logic, using information provided by the insured. DLT automatically uses additional sources (statistics, reports) to assess the claim and calculate the damage. Depending on the insurance policy, the smart contract can automatically calculate personal liability. In certain situations, a smart contract may activate an additional assessment of the claim. If the claim is approved, the payment to the insured is initiated through a smart contract.
The advantages of the new approach, based on smart contracts on blockchain technology, can be considered in several aspects. The submission of the claim is simplified and automated. Thanks to the direct exchange of damage information between insurers, DLT eliminates the need of brokers to participate and reduces the time for processing the claim. The built-in business logic in the smart contract in the blockchain eliminates the need for experts to review every claim (except in specific situations). The insurer has access to the history of the origin of the damage, which helps to identify potential attempted fraud. The information used is integrated, thanks to DLT's ability to aggregate data from multiple trusted sources. The process of paying the damage is automated by the smart contract on the blockchain, without the need to use an intermediary.
There are presented the advantages and the disadvantages of using private and public blockchain, as well as combined solutions with 2 blockchains (for automation of back-office operations to use a private blockchain, and for management of automatic payments with existing cryptocurrencies or when necessary to provide trust to use a public blockchain).
The presented solution is on the public blockchain Ethereum.

Г.8.33 V. Aleksieva, A. Huliyan, H. Valchanov, An approach of Crypto-token for Smart Contract based on Ethereum Blockchain, Journal of the Technical University – Sofia, Plovdiv branch, Bulgaria, “Fundamental Sciences and Applications”, Vol 25 No 1 (2019), pp.1-7, ISSN 2603-459X, https://journals.tu-plovdiv.bg/index.php/journal

The proposed paper presents a solution for the creation of a decentralized token for the implementation of a smart contract based on Ethereum block-chain. A web based interface has been created for Initial Coin Offering (ICO). In the experimental environment the research was carried out for various scenarios. The results are presented. This smart contract and web-based interfaces are presented in Г8.17 and Г8.18. In this paper are presented experimental tests and results for its functionality.
First part of the tests is related to the proper work of smart contracts- balance of account, transfer of tokens etc. There are a handful of tools for automated smart contract (written in Solidity) security vulnerability testing based on code-level analysis. In Reza’s approach is given a synopsis of the four most related tools that is possible to use in experiments, namely Oyente, Mythril, Securify, and SmartCheck. However, the evaluate level of rigor, ranging from syntactic, heuristic, analytic to fully formal, refers to underlying security testing technique of the given tool and up to this moment the researchers trusted on the implemented in the Solidity test tools. Truffle (and Solidity) has a built-in smart-contract-testing mechanism that is written in JavaScript, which here is used.
For direct transfer testing, 250 000 tokens are transferred from the administrator's address to the recipient's address. Once the transfer has taken place, the event is captured and checked for "Transfer" type. If this test is successful, the balance of the recipient address is checked for the presence of transferred tokens.
The delegated transfer check is similar to the direct transfer check. First, 100 tokens are transferred from the administrator's address to the address from which a delegated transfer will be allowed - address 1. It is allowed 10 tokens to be spent from address 3, which sends them to address 4. After performing these actions, it is expected that address 1 to have 90 tokens, address 2 - 0, and address 3 to be 10 tokens. The result of their implementation with wrong and correct parameters are shown.
In the real Ethereum block, as much as power to include block time is fixed, there comes a dynamic change of difficulty depending on how much power is included in the network. The tests were carried out in the local network with flat topology. The client connects to the Metamask server. The parameters of computers are Apple Mac Book Pro Late 2011 Specs, Core i5 (I5-2435M) 2.4GHz 2/4 Cores/Threads, 4GB DDR3 1333Mhz RAM. In the Metamask when sending an ether to buy a token, there is used protocol TLSv1.2. In the paper is presented network communication between client and Metamask server during successful transaction of tokens.

Г.8.34 D.Dinev, V.Aleksieva, H.Valchanov, K.Genov, LoRaWan Network Mobility Software Simulation Tool, // Computer Science and Tecjnologies, Bulgaria N.1,2021, pp.31-38, ISSN 1312-3335

This paper proposes a software simulation for analysis of the efficiency of allocating bandwidth in passive optical networks. An algorithm is presented for resource allocation in two stages for maximum utilization of bandwidth using orthogonal frequency multiplexing in the passive optical networks (OFDM-PON) - first it is allocated time interval (timeslot) for each subscriber unit (Optical Network Unit -ONU) and second sub-channels are arranged, where each of them consists of a group of subcarriers. Implementing the proposed approach based on dynamic allocation of subcarriers channels provides efficient allocation of bandwidth and reduces delays in transmission of requests of individual users.

The PON network consists of a centralized Optical Line Termination (OLT) on the ISP side and multiple Optical Network Unit (ONU) devices on the user side. ONUs share resources in a common optical stream that connects them to OLTs. The PON system must implement an appropriate MAC mechanism to ensure efficient transmission, efficient use of network resources, arbitrage access to the shared environment and avoid data collisions.

In the present development, an algorithm for resource allocation in two stages for maximum utilization of the connection capacity by using OFDM-PON is presented. OFDM-PON uses a synchronous frame structure to provide differential service to requests. When allocating resources, the proposed algorithm first allocates the time slot for each ONU and then arranges the subchannels (subcarrier group). This algorithm must meet two constraints:

• calculations for the allocation of resources are made for a single frame;

• One ONU uses only one subchannel to send to OLT data for multiple services within the frame duration.

The proposed algorithm is applied only in the upstream direction and is implemented in two phases:

1) Time slot allocation - assigns a temporary subchannel for each ONU

2) Redistribution of subchannels - the temporary subchannel j for ONU [i, j] will be replaced by a confirmed subchannel in which there is enough resource to deploy ONU [i + 1, j], thus minimizing the number of delayed for next frame resource blocks, and the bandwidth is compressed, i.e. no free trays remain.

To analyze the bandwidth allocation algorithm proposed in the present study, a traffic model for several OLTs is created, but visualization of the transmission matrix is performed only for an optional OLT.

A database has been created for storing the data from the individual experiments for the individual OLTs and for the users connected to them.